ABSTRACT

Title of thesis: ALGORITHMS AND MODELS FOR 3-D SHAPE MEASUREMENT USING DIGITAL FRINGE PROJECTIONS

Tao Peng, Doctor of Philosophy, 2006

Thesis directed by: Associate Professor Satyandra K. Gupta,
Department of Mechanical Engineering and
Institute for Systems Research

Many industrial applications require accurate and rapid measurement of the 3-D shapes of physical objects. Representative applications of 3-D shape measurement include mechanical reverse engineering, 3-D digital replication, and part inspection. Traditional 3-D measurement techniques, such as coordinate measurement machines (CMM) and laser scanning, provide high accuracy but are generally slow and expensive. In recent years, shape measurement based on digital fringe projection (SMDFP) has been developed for non-contact shape measurements. Systems based on SMDFP are promising due to low cost, fast speed, and flexibility. However, the existing models and algorithms for SMDFP systems need to be significantly improved to fully exploit the potentials of this technique.

This dissertation presents a new mathematical model for SMDFP that provides an accurate modeling of the optical geometry of SMDFP systems. Based on this model, three related algorithms for shape measurements were developed, namely the

algorithm for construction of absolute phase map, algorithm for construction of point cloud, and algorithm for estimation of sensor parameters. With the new model and algorithms, the measurement speed of existing SMDFP systems is improved and the calibration procedure is made easier. At the same time, high measurement accuracy is ensured. This dissertation also provides a framework for using adaptive projection patterns in SMDFP technique. A new algorithm was developed for automatic generation of projection patterns with variable fringe pitches to achieve improved measurement performance. This capability is particularly important for ensuring the accuracy and speed when measuring surfaces with a large range of normal directions. Finally, this dissertation presents a comprehensive uncertainty model for describing the relations between various error sources and the resulting uncertainties in shape measurements. Based on this model, measurement uncertainties can be estimated from the image data acquired in a measurement.

The research results reported in this dissertation can be used to improve the performance and features of existing SMDFP systems in the following aspect: measurement accuracy, speed, ease of calibration, and estimation of measurement uncertainties. These improvements could make SMDFP technique more attractive to industrial 3-D shape measurement applications and to stimulate the wide spread use of this technique.

ALGORITHMS AND MODELS FOR 3-D SHAPE MEASUREMENT USING DIGITAL FRINGE PROJECTIONS

by

Tao Peng

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2006

Advisory Committee:

Associate Professor Satyandra K. Gupta, Chair
Associate Professor David I. Bigio
Associate Professor Hugh A. Bruck
Professor Bongtae Han
Professor Amitabh Varshney

Dedicated to my dear parents and grandparents

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Satyandra K. Gupta, for his advising on my research work over the past four years. His teachings were inspiring. His encouragement and patience has led me to get through many difficult moments during the completion of this dissertation. Without his support, this dissertation would not be possible. I would also like to thank the other members in my dissertation review committee, Professors David Bigio, Hugh Bruck, Bongtae Han and Amitabh Varshney for their valuable suggestions.

Many thanks to my fellow researchers: Arvind Ananthanarayanan, Ashis Banerjee, Abhijit Deshmukh, and Cheuk Yiu (Horace) Ip, for proofreading my dissertation and papers and giving me valuable comments. And to the members of my research groups, Jeb Brough, Antonio Cardone, Regina Gouker, Mukul Karnik, Alok Priyadarshi, Max Schwartz, Brent Spranklin and Changxin Xu: Thank you for all the help; It has been a great pleasure to have worked with this fantastic group.

During my study in University of Maryland, my family, especially my wife Ning Qiu, have given me great support. It was because of their encouragement that I made my decision to pursue a PhD degree, and it was because of their support that I could concentrate on my research and study. I could not thank them enough for their love and faith in me.

Finally, I would like to thank Maryland Industrial Partnerships (MIPS) program and Automated Precision Inc., for funding a portion of my research.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides an introduction to the Shape Measurement technique based on Digital Fringe Projections (SMDFP) and an outline of the research work to be presented in this dissertation. Section 1.1 gives a brief description of the SMDFP technique, its applications in industrial 3-D shape measurement, and the advantages and disadvantages when comparing it to other techniques. Section 1.2 describes the motivation behind the research undertaken in this dissertation, as well as the objectives and associated challenges. An outline for the remainder of this dissertation is given in Section 1.3.

1.1 Background

1.1.1 3-D Shape Measurement for Industrial Applications

Many industrial applications require accurate and rapid measurement of the 3-D shapes of objects. Representative applications of 3-D shape measurement include mechanical reverse engineering, 3-D digital replication, part inspection, and quality control. For the first two applications, the 3-D shapes of objects need to be measured

so that 3-D mesh representations or CAD models of objects can be created to facilitate further engineering processes. Detailed examples include: measurements of the clay models and prototyped models in the body design of new automobiles; measurement of human bodies to make proprietary and artificial dental/orthopedic components for patients in medical practice; and visualization and reproduction of ancient statues in museums. For applications of part inspection and quality control, the shapes of objects need to be measured to compare against nominal data (e.g. CAD models) so that possible defects can be detected. A representative example is automatic on-line inspection in manufacturing industry, in which surfaces of the parts, as well as various 3-D features on the parts, need to be measured and analyzed. In this process, the measurements need to be adequately accurate. Errors in measurements can lead to erroneous inspection that results in acceptable parts being rejected and defective parts being accepted. On the other hand, the measurements need to be conducted in a short amount of time in order to follow the speed of production. Hence, both measurement accuracy and speed are equally important.

A predominant number of shape measurement equipments used in industry are contact-based. Representatives of such devices include Coordinate Measuring Machine (CMM), coordinate measuring arm and laser tracking system. These devices require mechanical parts, such as touch probes, to be placed in contact with the objects being measured, and possibly move along the surface of the object, in order to measure the surfaces. The requirement of physical contact leads to the following disadvantages:

- A high resolution measurement of a large surface with complex shape can be time-consuming since it takes a long time for the probe to walk along the surfaces.
- In cases that the surface of the object being measured is soft, fragile or easy to

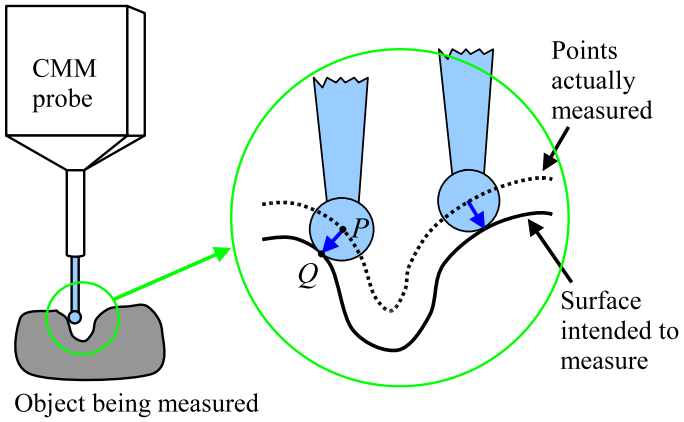


Figure 1.1: Point measurement using contact-based techniques is indirect measurement

be scratched, contact-based techniques will not work or are not desired.

- Contact-based techniques are indirect measurement. That is, the 3-D point coordinates obtained in measurement are not of points on the object's surface but positions of a characteristic point on the measuring device. Taking CMMs as an example, this characteristic point is the center of the touch probe's spherical tip. As can be seen from Fig. 1.1, when the tip of the CMM's touch probe gets in contact with the surface and a measurement point is taken, the point whose coordinates been measured is actually the center of the spherical tip, i.e. point P. The point on the surface which is intended to measure, point Q, is not known. As a result, post-processing of the measurement data is required to remove the normal offset ($\overrightarrow{QP}$) in the point coordinates obtained.

In order to solve the problems associated with contact-based techniques, a number of non-contact shape measurement methods have been developed and are now increasingly being used in industry [1, 2]. Techniques that have been used in non-contact shape measurement include laser scanning, photogrammetry, and structured light. All of these techniques use digital imaging and triangulation method to resolve

the shapes of objects. Compared to contact-based techniques, non-contact methods are much faster. They are able to measure thousands or even millions of points on surfaces in a few seconds. There is no need to touch the surface during the measurement and offsetting of point coordinates is not required. Moreover, some of the non-contact techniques can provide good measurement accuracy.

1.1.2 Shape Measurement Based on Digital Fringe Projection

As mentioned above, structured light is one of the non-contact techniques that have been used in 3-D shape measurement. A typical shape measurement system based on structured light consists of one projection unit and one or more cameras. During the measurement, light patterns with known structures are projected sequentially on the object being measured. Meantime, images of the object under the light projections are captured by the camera(s). By utilizing triangulation method and knowledge on the light patterns, the 3-D shape of the object can be resolved from the images captured.

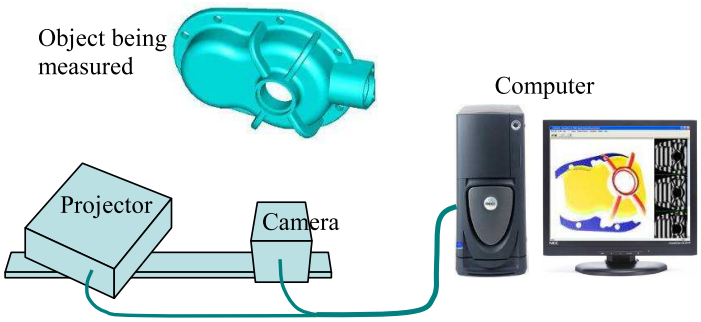
A commonly used type of structured light pattern for shape measurement is fringe pattern, particularly fringe patterns with sinusoidal intensity distributions. The corresponding shape measurement technique by using fringe patterns is often referred as Shape Measurement based on Fringe Projection (SMFP). In recent years, efforts have been made to use computer projectors as the projection unit for SMFP systems. Compared to traditional projection units such as those based on glass grating [1] and laser interferometry [3], computer projectors offer promising performance and features, as well as challenges, to the shape measurement technique. The shape measurement technique with the use of computer projectors is referred as Shape Measurement based on Digital Fringe Projection (SMDFP).

Figure 1.2(a) shows the hardware setup of a typical SMDFP system with one projector and one camera. The projector and the camera are placed away from each other with an angle between their optical axes. They are usually fixed in a housing and the unit as a whole is usually called the measurement sensor. Both the projector and the camera are controlled by the computer. The object to be measured is placed at a distance away from the sensor such that it is in the camera's field view while at the same time covered by the light projection from the projector. Figure 1.2(b) shows an example of the sinusoidal fringe patterns that are used for projection. The corresponding image of the object under the projection of the fringe pattern is shown in Fig. 1.2(c). The final output of the shape measurement is a dense point cloud that represents the surface of the object, as shown in Fig. 1.2(d).

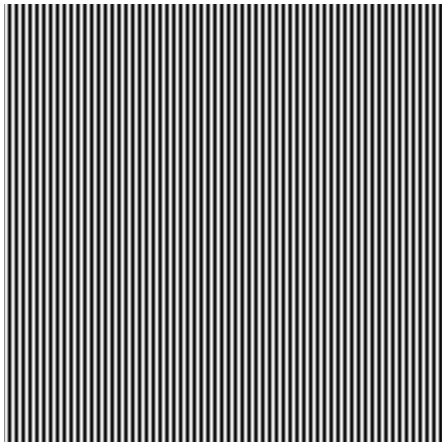
1.1.3 Advantages of SMDFP

Compared to other non-contact shape measurement techniques such as laser scanning and photogrammetry, SMDFP has several advantages on measurement speed and system cost, which makes it a preferable choice for many industrial applications that require an accurate, fast and inexpensive shape measurement tool. In the following paragraphs, a comparison between SMDFP and the two other techniques will be given and the advantages of SMDFP will be described.

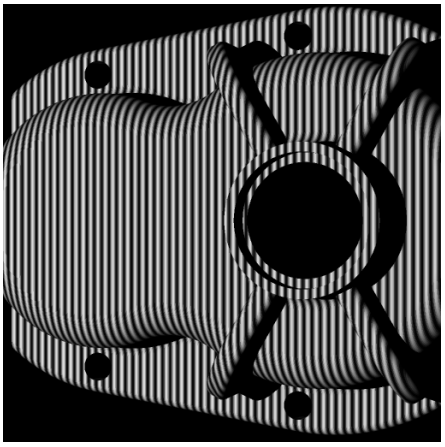
Laser scanning technique measures the surface of an object by scanning a laser beam (or projecting a laser sheet) on the object and observing the position of the laser spots/contour from a different angle (by using a camera). Figure 1.3 shows a schematic diagram of the measurement setup when using a laser scanner. Laser scanning technique offers good measurement accuracy and a reasonably high data rate, and the scanner can be made into a compact size and light weight [6, 7]. However, a laser scanner is a line-scan device, which means it is only able to measure points



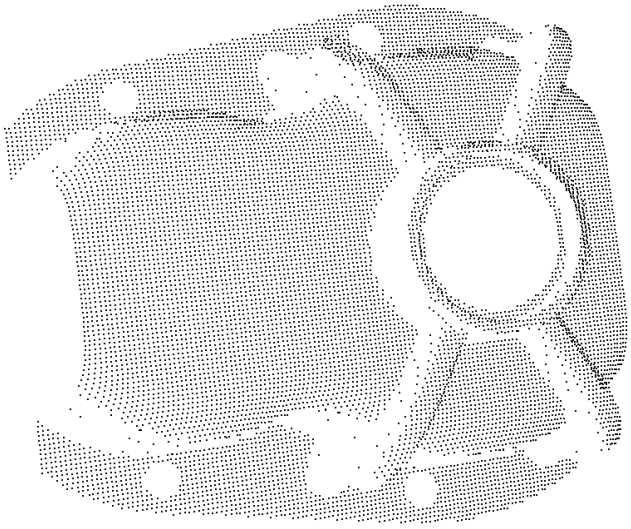
(a) Schematic diagram of a SMDFP system with one projector and one camera



(b) One of the fringe patterns projected on the object



(c) Image of the object when shone by the fringe pattern shown in (b)



(d) The constructed point cloud that represents the surface of the object been measured

Figure 1.2: 3-D shape measurement based on digital fringe projection

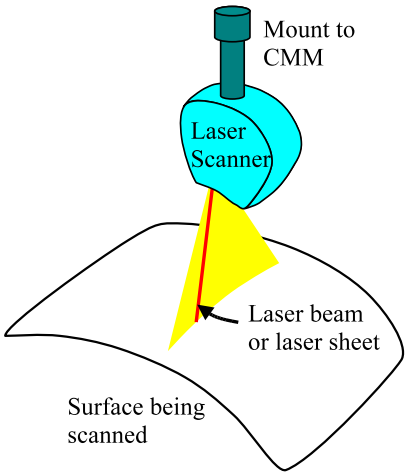


Figure 1.3: Schematic diagram of laser scanning technique

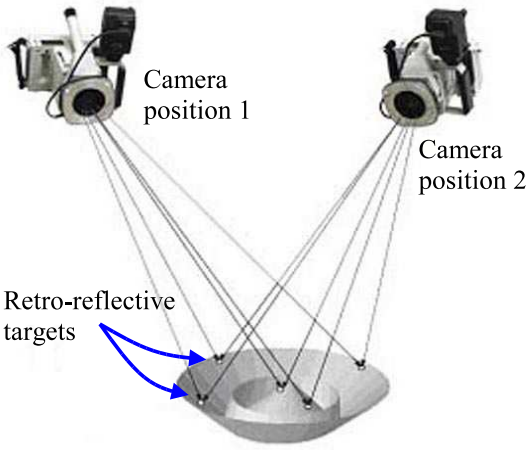


Figure 1.4: Schematic diagram of shape measurement based on photogrammetry (Picture courtesy of Geodetic Systems, Inc.)

that are in the scanning plane of the laser beam or in the plane of the laser sheet. It relies on other device, such as a CMM, to move it around in order to measure an area. As a contrast, SMDFP system is doing true "area scan", i.e. it is able to measure an area on a 3-D surface without the need to be moved around by an external device. As a result, SMDFP technique can perform measurement much faster than laser scanning. While a laser scanner can measure thousands or tens of thousands points per second, a SMDFP system can measure over a million points in a couple of seconds or less, depending on hardware configuration. Also, SMDFP systems are significantly cheaper than laser scanners, due to the use of digital cameras and computer projectors as main components.

Photogrammetry [8, 9, 10] is a technique based on multiple-view geometry, in which images of an object taken from two or more different perspectives are combined to form a 3-D view of the object (see Fig. 1.4). A challenging problem in photogrammetry is the registration of homogeneous points in multiple images. Despite the significant research advances in this field, an universal and accurate algorithm

that works for natural images of objects is not available. A solution to the registration problem is to place special marks, such as retro-reflective targets, on the surface of the object to be measured. These marks can be easily and accurately located by image processing algorithms, and by placing them in a fairly sparse manner and adding special patterns on them an automatic and robust registration algorithm can be realized. However, by applying special marks on the surface of the object, the number of points obtained from one measurement cannot be very high. Compared to SMDFP technique, the data rate of photogrammetry is quite low. Therefore, photogrammetry is usually not used for measurement of objects with complex shapes, but often as a tool for global coordinate registration.

From the above comparison we can see that, SMDFP technique has significantly higher data rate, and a lower system cost, than other non-contact shape measurement methods. In latter chapters, we will also show that it has potential to achieve sufficiently high accuracy that could satisfy the requirements of many shape measurement applications. These factors make SMDFP technique a very promising technology in the field of 3-D shape measurement involving objects with complex shapes.

1.1.4 DMD-based Digital Projection for SMDFP Systems

The projection unit in a SMDFP system projects light patterns on objects during measurements. The accuracy and repeatability of the light patterns are crucial to the accuracy of shape measurement. Two of the most popularly used methods for projection of fringe patterns are glass grating and laser interferometry. For their use in shape measurement devices, these two methods have been tailored and improved on certain features, and the resulting techniques are Miniature Projection Technique (MPT) [1] and Accordion Fringe Interferometry (AFI) [3, 11, 12] respectively.

In recent years, computer projectors based on Digital Micro-mirror Devices (DMD)

Figure 1.5: A DMD chip made by Texas Instruments (1280×720 pixels)
(Picture courtesy of Texas Instruments)

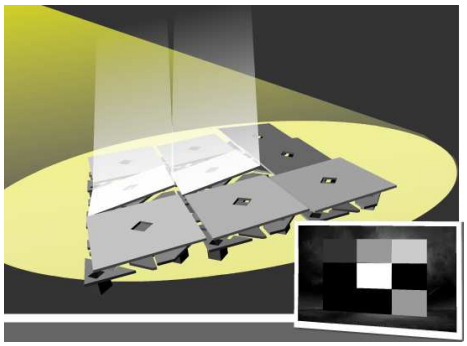


Figure 1.6: DMD uses the flickering of micro-mirrors to achieve a large range of gray-levels in a fully digitized manner
(Picture courtesy of Texas Instruments)

have been increasingly used as the projection units in SMDFP systems. DMD is a semiconductor technology that was invented and successfully commercialized by Texas Instruments in the early 90's [13, 14]. A DMD chip is a micro-mechanical silicon chip, which measures less than 5/8 inch on each side but contains more than 700,000 tiny, movable aluminum mirrors, as well as a wealth of logic, memory and control circuitry (see Fig. 1.5 for a picture of a DMD chip made by Texas Instruments). Each individual mirror on the chip can be controlled by computer signals to move with great precision and at very fast speed. By shooting a light beam on a DMD chip and sending programmed computer signals to it, high quality light patterns can be create with excellent resolution, brightness, contrast and gray-level fidelity (see Fig. 1.6).

Compared to projection devices based on glass grating or laser interferometry, DMD-based computer projectors have the following advantages:

- **Great flexibility of light patterns**: The light pattern generated by a DMD projector can be described by a high-resolution (e.g. 1024×768 pixels) gray-scale bitmap. The gray-scale of each pixel can be controlled individually at 256 different levels at least. Advance programming of DMD chip can extend the range

of gray-levels to almost infinite. Due to the large number of pixels it has and the individual controllability over them, DMD projector can generate almost arbitrary light patterns. When used in fringe projection with phase-shifting method, it is able to generate fringe patterns with different fringe widths, fringe orientations and phase-shift values. Sophisticated projection patterns, such as adaptive patterns (see Chapter 5), cannot be realized by using glass grating or laser interferometry. However, they can be easily generated by DMD projectors.

- **Accurate control over light patterns**: DMD chips are able to control the projected light intensities in a very accurate and repeatable manner [16]. On the other hand, since the control signals from computer to DMD projector can be transmitted in digital format, there will be no distortion introduced in between. As a result, the light patterns generated by DMD projector can be controlled precisely from the computer.

- **Low cost**: DMD projectors are commercially available and they are much less expensive than projection devices based on glass grating or laser interferometry. Nowadays, the typical market price of a DMD projector ranges from $900 to $2,000.

Due to its excellent performance and attractive attributes, DMD-based computer projector has become a promising projection unit for SMDFP systems.

1.2 Motivation and Research Issues

SMDFP has shown considerable promise in the field of non-contact 3-D measurements due to its many desirable features. However, as an emerging technique, SMDFP is still in development stage and its potentials have not been fully exploited. To make it more attractive to industrial 3-D measurement applications, improvements are needed in

the following aspects: measurement accuracy, measurement speed, ease of calibration and the capability of handling complex shapes.

The mathematical model used for a SMDFP system and the related algorithms for point cloud construction are crucial to the system's measurement accuracy. Existing models for SMDFP can be classified into two categories, geometry based models and calibration matrix based models [18] (see Section 2.1 for more details). Between the two, geometry based models have been more popularly used and studied, mainly because it has the potential to achieve higher measurement accuracy. A very promising model based on geometric approach was proposed by Legarda-Sáenz et. al. recently [19]. However, this model involves a large number of parameters (30 in total) and an accurate acquisition of all the parameters requires a sophisticated calibration process. To make the calibration process easier while maintaining a high measurement accuracy, the mathematical model needs to be modified and new algorithms need to be developed.

The measurement speed of a SMDFP system is mainly determined by the amount of time required for projecting patterns and acquiring images in a measurement cycle. From the hardware point of view, a fast measurement speed requires a high frame refreshing rate (for the projector), a short exposure time (for the camera) and a good synchronization between the projector and the camera. From the algorithm point of view, a fast measurement speed means minimizing the number of projection patterns used in a measurement. Generally speaking, fringe projection methods that yield better performance, in terms of accuracy and the capability of handling surface discontinuity, require more projection patterns. However, the redundancy that exists in multiple fringe patterns has been traditionally ignored, which could actually be used for reducing the number of patterns required for a measurement.

Most existing SMDFP systems use fixed-pitch fringe patterns, i.e. patterns containing straight fringes that are uniformly spaced. When measuring surfaces with a

large range of normal directions, the use of fixed-pitch fringe patterns requires extra number of patterns to achieve good measurement accuracy and coverage. This is because, for such surfaces, the fringe pitch (in the projection pattern) that gives satisfactory measurement performance may be significantly different for different areas on the surface. Therefore, multiple patterns with different fringe pitches need to be used for a single measurement. However, the increase in number of patterns per measurement is not desired for applications that demand high measurement speed, e.g. 100% on-line part inspection. Hence, new methods need to be developed for measuring surfaces with a large range of normal directions.

From the metrology point of view, it is a desired feature for SMDFP systems to give the uncertainty of measurement along with a measurement result. The accuracy of a shape measurement device can be approximately evaluated by measuring some master gauge with accurately known geometry. However, the uncertainties of a specific measurement depend on a number of measurement-related factors, e.g. environmental lighting, optical properties of the surface, and the shape of the object. Therefore, the accuracy evaluation obtained from the measurement of master gauge may not reflect the real accuracy of a measurement made on other objects. A good solution to this would be, to build an uncertainty model for SMDFP systems, which could give a good estimation of the measurement uncertainties by analyzing the image data acquired from the measurement and the knowledge on other fixed uncertainty sources in the system. However, a comprehensive uncertainty model for SMDFP systems does not currently exist.

The research work conducted in this dissertation focuses on the following issues:

- **Development of system model and algorithms for shape measurements**:

  The accurate modeling of the optical geometry of SMDFP systems involves a

large number of sensor parameters. To acquire these parameters accurately, a sophisticated calibration procedure is required [19]. A possible way to simplify the calibration is to change the representation of the sensor parameters (some of them if not all) such that they can be acquired easily and reliably. The new representation of the parameters should also ensure the accurate modeling of the system. In this dissertation, a reference phase map is used as an implicit representation of a majority portion of the projector parameters. Based on this approach, a new algorithm for point cloud construction needs to be developed. The algorithm for estimation of sensor parameters also needs to be modified.

The improvement in measurement speed requires new ideas to reduce the number of projection patterns needed in a measurement cycle. Through a careful study of the phase-shifting method with multiple fringe frequencies, we discovered that images acquired by using this method contain redundant information which can be utilized to reduce the number of patterns. A new algorithm for construction of absolute phase maps needs to be developed based on this approach. While using fewer projection patterns, this algorithm should also maintain a good phase measurement performance, in terms of the capability of handling surface discontinuity and robustness to projection and image noises.

Since the use of adaptive projection patterns is considered in this dissertation, the mathematical model and algorithms for shape measurement should also be able to work with fringe patterns with variable fringe pitches.

- **Development of a framework for using adaptive projection patterns**: The idea of adaptive projection pattern is proposed for the measurements of surfaces with a large range of normal directions. The basic idea of adaptive projection pattern is to vary the fringe pitch in a pattern such that, when the pattern is projected on the surface being measured, all areas on the surface

receive optimal (or close) fringe pitches. Hence, a good measurement accuracy and coverage can be achieved by using one set of phase-shifted adaptive patterns, instead of using multiple sets of fixed-pitch fringe patterns. So far, very little research has been done on adaptive projection patterns. A complete framework needs to be developed on the use of adaptive projection patterns in SMDFP technique. The system model for SMDFP and the algorithms for shape measurement need to ensure the support for adaptive projection patterns. A new algorithm needs to be developed for automated generation of adaptive pattern(s) for measuring an object, based on knowledge of the approximate shape of the object. The suitable applications for using adaptive projection patterns also need to be studied.

- **Development of new model and algorithms for estimation of measurement uncertainties**:

  In order to estimate the uncertainties in a shape measurement, the sources that could contribute errors in the shape measurement result need to be identified. The relationship between individual error sources and the corresponding uncertainties created in the results needs to be found out and modeled appropriately. Algorithms for estimating the magnitudes of error sources, either measurement-dependent or fixed, need to be developed.

1.3 Dissertation Outline

The research work conducted in this dissertation is presented in the following manner:

Chapter 2 gives a review of the related work and the state of the art in SMFP/SMDFP technique. Chapter 3 explains the mathematical model and the related algorithms developed for conducting shape measurements using SMDFP. Chapter 4 describes the

simulator developed for SMDFP systems, which has been used as a tool for testing the models and algorithms proposed in this dissertation. Chapter 5 discusses the use of adaptive projection patterns in SMDFP, including the measurement procedure and the required algorithms. Chapter 6 describes a framework developed for estimation of measurement uncertainties. Chapter 7 summarizes the conclusions reached from this dissertation and provides suggestions for future extensions.

Chapter 2

Related Work

This chapter gives a review of the related work and the state of the art in SMFP/SMDFP technique. Two major research topics are discussed. One is the mathematical models for SMFP systems, which is presented in Section 2.1, and the other is the methods for acquisition of absolute phase maps of surfaces. The discussion of the latter is presented in two parts, namely the phase-shifting method (Section 2.2) and the phase unwrapping algorithms (Section 2.3). Since the research work undertaken in this dissertation also involves the modeling of cameras, projectors and lens distortions, a short review on camera models and camera calibration methods is also given (Section 2.4).

2.1 Mathematical Models for SMFP Systems

Existing mathematical models for SMFP systems can be put into two categories, geometric approach (also referred as optical geometry based models) and calibration matrix based approach [18]. The former group of models use mathematical equations to describe the optical geometry of SMFP systems, i.e. the projection of 2-D patterns to the space and the imaging of 3-D objects in the camera. The algorithm

for construction of point clouds that represent objects' shapes is derived from the mathematical equations established. For calibration matrix based models, the details on the system's optical geometry is ignored. Instead, the approach is to get a dense sampling of the 3-D measurement volume of the system, which is organized in the form of a huge coefficient matrix. The construction of point clouds (that represent the surfaces being measured) is achieved by interpolations of the sample points.

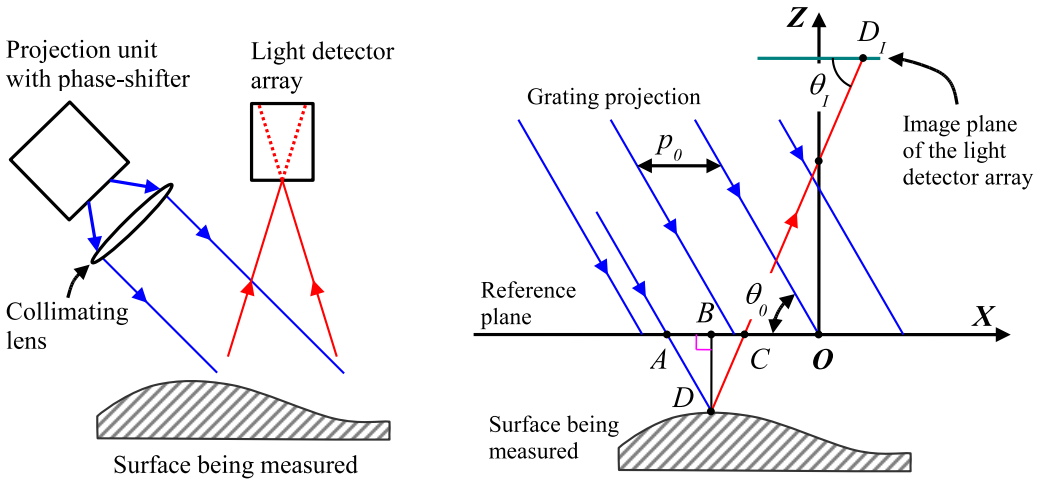
In the following sections, these two approaches are explained individually.

2.1.1 Models Based on Geometric Approach

2.1.1.1 Model Proposed by Srinivasan et. al.

Srinivasan et. al. [20] developed one of the earliest shape measurement systems based on fringe projection. The system consists of a projection unit (with phase-shifting capability) and a light detector array. A schematic diagram of the system's hardware setup is shown in Fig. 2.1(a). The projection unit generates sinusoidal fringe patterns by using laser interferometry. The projection of the fringe patterns are collimated, which means the fringe pitch of the pattern remains constant despite the distance to the projection unit. The light detector array is basically a digital camera, whose optical axis is aligned to be co-planar with the fringe projection's collimation direction with an angle of $(90^0 - \theta_0)$ between them. The coordinate frame for the system is defined as shown in Fig. 2.1(b). The collimation direction and the optical axis of the light detector array determines the X-Z plane. Z-axis is perpendicular to the image plane of the light detector array (i.e. parallel with the optical axis of the light detector array) and goes through the center of it. The position of the X-axis is defined close to the center of the system's measurement volume.

When placing a flat plane coinciding with the X-Y plane, the fringe pitch of the projected fringe pattern on the plane is a constant. Due to its purpose of obtaining



(a) Schematic diagram of hardware setup

(b) Mathematical model of the optical geometry

Figure 2.1: SMFP system and mathematical model proposed by Srinivasan et. al.

reference/calibration data, such plane is often called a reference plane. Let p_0 denote the fringe pitch on the plane, the distribution of light intensity on the plane can be written using the following equation:

$$I(x, y) = a(x, y) + b(x, y) \cos\left(2\pi x / p_0\right) \tag{2.1}$$

where $b(x, y)$ is the fringe contrast at point $(x, y, 0)$ and $a(x, y)$ is the background intensity. The term $(2\pi x/p_0)$ is the reference phase value at point $(x, y, 0)$ and often denoted by $\Phi_R(x, y)$, i.e.

$$\Phi_R(x, y) = 2\pi x/p_0 \tag{2.2}$$

The shape measurement system uses phase-shifting method (will be discussed in Section 2.2) to measure the phase values of points on an object's surface, and from the phase values to compute the 3-D coordinates of the points. Consider a point, D, on the surface the object being measured (see Fig. 2.1(b)). The phase value of point D, denoted by $\Phi(D)$, is the same as the phase value of point A, which is on the

reference plane and along the same projection ray as point D. On the other hand, point C is the point on the reference plane that is imaged to the same point (on the image plane of the light detector array), D_I, as point D. The distance between point A and point C can be expressed using their phase values as in the following equation:

$$|AC| = -\frac{p_0}{2\pi}\big(\Phi_R(A) - \Phi_R(C)\big) \tag{2.3}$$

where $\Phi_R(A)$ and $\Phi_R(C)$ are the phase values of point A and point C respectively. Since $\Phi_R(C)$ is the phase value associated with point D_I when the reference plane is in place, the term $(\Phi_R(A) - \Phi_R(C))$ actually represents the phase change at D_I when the object in place is the real surface to be measured and the reference plane respectively. Hence, this term is also called the "phase difference" at point D_I and denoted by $\Delta\Phi(D_I)$, i.e.

$$\Delta\Phi(D_I) = \Phi_R(A) - \Phi_R(C) \tag{2.4}$$

By utilizing the geometric relationships of points A, B, C, and D, the z-coordinate of point D can be written as the following:

$$z_D = -|BD| = \frac{-|AC|\tan\theta_0}{1 + \tan\theta_0/\tan\theta_I} \tag{2.5}$$

where θ_I is the angle between line $\overline{DD_I}$ and the image plane of the light detector array, which can be calculated from the position of D_I and parameters of the light detector array.

Combining Eqns. 2.3, 2.4 and 2.5, the z-coordinate of point D can be expressed

as a function of the phase difference $\Delta\Phi(D_I)$ and the angle θ_I as the following:

$$z_D = \frac{p_0 \tan\theta_0 \, \Delta\Phi(D_I)}{2\pi(1 + \tan\theta_0 / \tan\theta_I)} \tag{2.6}$$

Once z_D is known, the x- and y-coordinates of point D can be calculated from the position of point D_I.

A remarkable characteristic of the system developed by Srinivasan et. al. is the use of collimated projection, which achieves constant fringe pitch on the reference plane and hence simplifies the mathematical model of the system. However, collimated projection can only cover a surface area of approximately the same size as the projection unit's collimation lens, hence the measurement volume of the system is limited.

2.1.1.2 Model Proposed by Toyooka and Iwaasa

Toyooka and Iwaasa [22] developed a SMFP system which consists of a slide projector and a camera. Since a slide projector complies with the rules of perspective projection, the optical geometry model developed by Srinivasan et. al. (see Section 2.1.1.1), which requires collimated projection, is no longer applicable. Toyooka and Iwaasa developed a new model for their system, in which a simplified pinhole camera model is used to describe the optical geometry of the projector and the camera.

An illustration of the mathematical model proposed by Toyooka and Iwaasa is shown in Fig. 2.2. The model requires that the optical axes of the projector and the camera are in the same plane and the projection center of the projector (point P) and the projection center of the camera (point I) have the same distance to the image plane of the camera. The system coordinate frame is defined as the following: The Z-axis passes through the projection center of the camera and is perpendicular to the camera's image plane; The optical axes of the projector and the camera determines

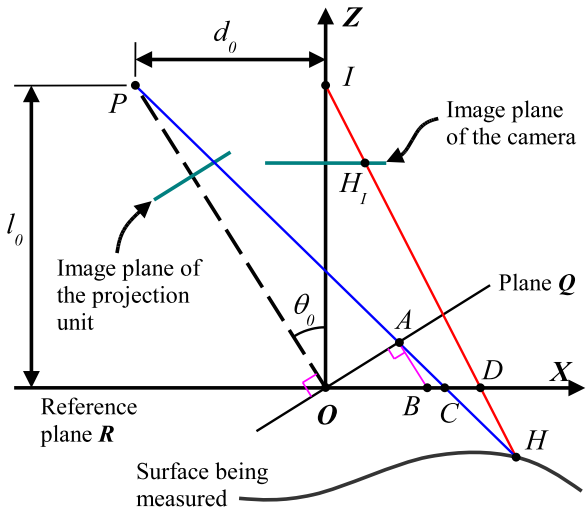


Figure 2.2: Mathematical model for SMFP system proposed by Toyooka and Iwaasa

the X-Z plane and the intersection of them defines the origin O. By this definition, point P and point I have the same distance, l_0, to the X-axis.

The slide projector generates sinusoidal fringe patterns with fixed fringe pitch. It is also required that the direction of the fringes is perpendicular to the X-Z plane. Therefore, if a reference plane is placed perpendicular to the optical axis of the projector, e.g. plane $\boldsymbol{Q}$ as shown in Fig. 2.2, the distribution of light intensity on the plane can be written as the following:

$$I'(s,y) = a'(s,y) + b'(s,y)\cos\left(2\pi s/p_0'\right) \tag{2.7}$$

where s is the distance to the plane defined by line $\overline{PO}$ and the Y-axis, and p_0' is the fringe pitch on the reference plane. In Toyooka's model, the reference plane, plane $\boldsymbol{R}$, is placed coinciding with the X-Y plane, which means not perpendicular to the projector's optical axis. Due to the projector's perspective projection, the light pattern appearing on plane $\boldsymbol{R}$ is not of fixed fringe pitch. Let θ_0 denote the angle between the optical axes of the projector and the camera, and let d_0 denote the

distance between their projection centers (i.e. points P and I). The light intensity distribution on reference plane $\boldsymbol{R}$ can be expressed using the following equation:

$$I(x,y) = a(x,y) + b(x,y)\cos\left(\frac{2\pi x}{p_0} - \frac{2\pi x}{p_0}\frac{\sin\theta_0\cos\theta_0}{l_0/x + \sin\theta_0\cos\theta_0}\right) \tag{2.8}$$

where p_0 is the local fringe pitch at point O.

Consider a point, H, on the surface being measured. Point H_I is its image on the camera's image plane and point D is its corresponding point on the reference plane ($\boldsymbol{R}$) which is imaged at H_I as well. Point C is a point on the reference plane that is on the same projection ray as H, i.e. their phase values are the same. From the geometric relationship of these points, the following equation of the z-coordinate of point H can be obtained:

$$z_H = l_0\left|\overrightarrow{DC}\right| \Big/ \left(d_0 + \left|\overrightarrow{DC}\right|\right) \tag{2.9}$$

where $\left|\overrightarrow{DC}\right|$ is the signed distance between points D and C, i.e. $\left|\overrightarrow{DC}\right| = x_C - x_D$, since the y-coordinates of D and C are always the same. The x-coordinates of points D and C can be calculated from their phase values by using the following equation:

$$\Phi(x) = \frac{2\pi x}{p_0}\left(1 - \frac{\sin\theta_0\cos\theta_0}{l_0/x + \sin\theta_0\cos\theta_0}\right) \tag{2.10}$$

The mathematical model proposed by Toyooka and Iwaasa has a number of requirements on the alignment of SMFP system, e.g. the optical axes of the projector and the camera need to be in the same plane (X-Z plane), the projection centers of the projector and the camera need to be of the same distance to the camera's image plane, the direction of fringes needs to be perpendicular to the X-Z plane, and the reference plane needs to be placed in parallel with the camera's image plane. All these requirements simplifies the mathematical model of the system while at the same time

limits the system's flexibility and increases calibration efforts. Some of the system parameters, such as the distance between the projection centers of the projector and the camera (d_0) and the distance from the projection centers to the reference plane (l_0), cannot be obtained by direct measurement. However, the method for accurately estimating these parameters were not given.

2.1.1.3 Model Proposed by Hu et. al.

Hu et. al. [24, 25] proposed a model for SMFP systems which can be considered as an extension of the model proposed by Toyooka and Iwaasa (see Section 2.1.1.2). Compared to Toyooka's model, Hu's model removed some of the constraints on system configuration, such as the requirement of the projection centers of the projector and the camera to be of the same distance to the camera's image plane. Nevertheless, it requires the following conditions to be met: 1) The optical axes of the projector and the camera need to be in the same plane; 2) The vertical direction of the projector's image plane and the vertical direction of the camera's image plane need to be parallel to each other and perpendicular to plane defined by the optical axes of the projector and the camera; And 3) the principal point of the projector needs to be centered at the projector's image plane. The alignment process to achieve some of the goals is given in their paper [24]. Similar to the model proposed by Toyooka, Hu's model also uses a simplified pinhole camera model, but with a few more parameters, to describe the optical geometry of the projector and the camera.

Two coordinate frames are defined in Hu's model (see Fig. 2.3). The main frame, XYZ, is defined on the projector's optics as the following: Axis Z coincides with the optical axis of the projector; Axis Y is parallel to the vertical direction of the projector's image plane; And the intersection of the optical axes of the projector and the camera is the origin, O. The other coordinate frame, $X^{(C)}Y^{(C)}Z^{(C)}$, is defined on the camera's optics for convenient description of the point mapping from the 3-D

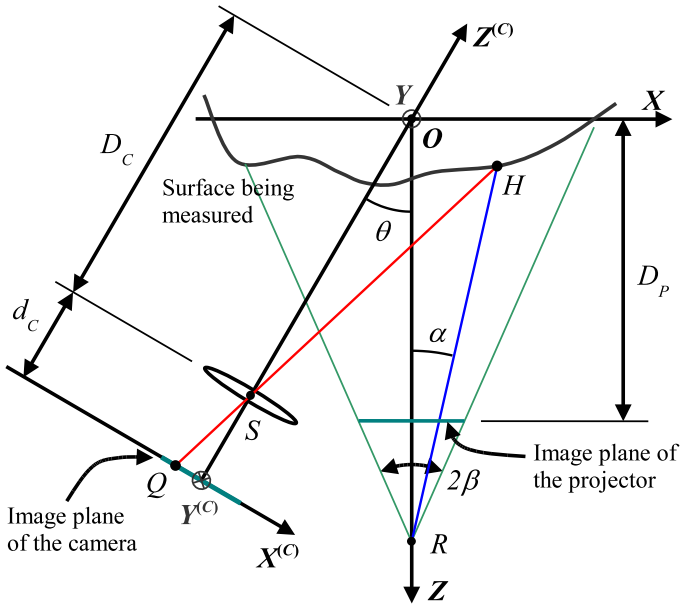


Figure 2.3: Mathematical model for SMFP system proposed by Hu et. al.

space to the 2-D image plane of the camera. Hence, it is referred as the camera coordinate frame. The axis $Z^{(C)}$ coincides with the optical axis of the camera and passes through the main frame's origin, O. Axis $Y^{(C)}$ is parallel to the vertical direction of the camera's image plane, which means it is also parallel to the Y-axis of the main frame. Axis $X^{(C)}$ lies on the image plane of the camera, and the camera's principle point defines the origin of the frame.

Let S denote the projection center of the camera. Let D_C and d_c denote the distances from S to point O and the origin of frame $X^{(C)}Y^{(C)}Z^{(C)}$ respectively. The coordinate transformation between frame $X^{(C)}Y^{(C)}Z^{(C)}$ and frame XYZ can be de-

scribed by the following equation:

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & -(D_C + d_C)\sin\theta \\ 0 & -1 & 0 & 0 \\ \sin\theta & 0 & -\cos\theta & (D_C + d_C)\cos\theta \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^{(C)} \\ y^{(C)} \\ z^{(C)} \\ 1 \end{bmatrix} \tag{2.11}$$

where $[x, y, z, 1]^T$ are the homogeneous coordinates in frame XYZ, $\left[x^{(C)}, y^{(C)}, z^{(C)}, 1\right]^T$ are the homogeneous coordinates in frame $X^{(C)}Y^{(C)}Z^{(C)}$, and θ is the angle between the optical axes of the projector and the camera.

A 3-step phase-shifting method with an additional centerline pattern is used to obtain the phase values of points on the surface being measured. By using vertical sinusoidal fringe patterns with selected phase-shift values, the absolute phase values of points on the X-Y plane can be described by the following function:

$$\Phi(x, y, 0) = 2\pi x / p_0 \tag{2.12}$$

where p_0 is the fringe pitch on X-Y plane. Furthermore, the absolute phase value of an arbitrary point in the projector's field-of-view can be described using the following equation:

$$\Phi(x, y, z) = \frac{2\pi x z_R}{p_0(z_R - z)} \tag{2.13}$$

where point R is the projector's projection center and z_R is its z-coordinate. The value of z_R can be calculated from the projector parameters (both x_R and y_R are equal to zero as known from the coordinate frame definition).

Consider a point H on the surface being measured (see Fig. 2.3). Let Q denote the image of point H on the camera's image plane and let $\Phi(H)$ denote the absolute phase value of point H obtained by using phase-shifting method. The coordinates of

point Q in the camera coordinate frame can be easily calculated from its position in the image and the camera's intrinsic parameters. Its coordinates in the XYZ frame can then be acquired by using Eqn. 2.11. Let α denote the angle between line $\overline{RH}$ and the Z-axis. α can be calculated from $\Phi(H)$ as the following:

$$\alpha = \arctan\left(\frac{p_0}{2\pi z_R}\Phi(H)\right) \tag{2.14}$$

By using α and the position of point Q (x_Q, y_Q, z_Q), as well as the positions of points S and R, the z-coordinate of point H can be computed from the following equation:

$$z_H = \frac{(z_S - z_Q)\big[(x_R - x_Q)\cos\alpha + z_R \sin\alpha\big] + z_Q(x_S - x_Q)\cos\alpha}{(x_S - x_Q)\cos\alpha + (z_S - z_Q)\sin\alpha} \tag{2.15}$$

Once z_H is known, the x- and y-coordinates of H can be computed by using the following equations:

$$\begin{cases} x_H = \dfrac{(z_H - z_Q)(x_S - x_Q)}{z_S - z_Q} + x_Q \\ y_H = \dfrac{(z_H - z_Q)(y_S - y_Q)}{z_S - z_Q} + y_Q \end{cases} \tag{2.16}$$

Besides the mathematical model for SMFP system and the algorithm for construction of the point cloud that represents the measured surface, Hu et. al. also developed a two-step approach to estimate the parameters in the mathematical model, e.g. θ, D_C, d_C and D_P. The first step of the process is to acquire approximate values of the parameters. A few calibration patterns, such as a horizontal line or a bright circular spot at the center of the pattern, are projected on a flat plate which is mounted on a straight rail. Under the projection of the calibration patterns, images of the plate are taken with the plate being translated to a number of different positions. By analyzing the positions of the spots and the lengths of the lines in the images, approximate values of the parameters can be obtained. The second step of the process is to fine-tune the values of the parameters by minimizing the residual errors of a few

calibration measurements. The calibration measurements are conducted on a gauge part with a number of geometric features whose positions and dimensions are known very accurately (from measurement made by other means such as CMM). By varying the parameters around the initial values obtained from the first step, the point cloud computed changes and hence is the residual error of the measurement. The set of parameter values that gives the minimum residual error is then taken as the final values for the parameters.

Compared to Toyooka's model, the model proposed by Hu et. al. has less constraints on system configuration and provides better measurement accuracy with the use of the parameter estimation process developed. However, the remaining requirements on system alignment still demand considerable efforts in calibration. Moreover, Hu's model does not consider the lens distortions of the projector and the camera, which are generally not negligible for high accuracy measurements [25].

2.1.1.4 Model Proposed by Legarda-Sáenz et. al.

Legarda-Sáenz et. al. [19] proposed a sophisticated mathematical model for SMFP systems with one projector and one camera. The pinhole camera model with modeling of lens distortion is used for both the camera and the projector. The complete model contains 30 parameters in total. It describes the optical geometry of the system with high accuracy and has minimal constraints on system alignment.

The construction of the point cloud (representing the surface of the object) is done by triangulation method, similar to the technique used in stereo vision. For each point M on the object's surface that is imaged to a point M_I in the camera's image plane, the pixel coordinates of M_I are used to determine two degrees of freedom of point M, by utilizing the camera model and the associated parameters that are known. The remaining one degree-of-freedom of point M (often called the depth of M) can be calculated from the absolute phase value of M, which was acquired by using

the phase-shifting method. However, when considerable degree of lens distortions is present in the projector, locating point M accurately by using the above algorithm becomes challenging and computationally intensive.

An algorithm for estimating the system parameters was also proposed by Legarda-Sáenz et. al. Due to the large number of parameters to be estimated and the poor conditioning of the problem, a two-step approach is used. At the first step, the camera parameters and the projector parameters are estimated separately by using well-known techniques in literature [26, 27]. At the second step, the parameter values obtained from step one are used as initial values for the global minimization of the estimation's residual errors. More details on the formation of the minimization problem is given in their paper [19].

Compared to the earlier models based on geometric approach, the model proposed by Legarda-Sáenz et. al. is the most accurate one, due to the use of the full pinhole camera model and the modeling of lens distortions, and it has the minimum constraints on system alignment. As a trade-off, a large number of system parameters (30 parameters in total) need to be acquired and an accurate acquisition of the parameters needs to go through a carefully designed calibration process.

2.1.1.5 Summary

The four models for SMFP systems explained above are all based on geometric approach, i.e. the construction of point clouds (for representing the surfaces of objects) is based on a mathematical model that describes the optical geometry of the system. Among these models, the one proposed by Legarda-Sáenz et. al. is the most accurate and has the minimum constraints on system alignment. However, a large number of system parameters (30 parameters in total) are involved in Legarda-Sáenz's model and an accurate acquisition of the parameters needs to go through a carefully designed calibration process. Moreover, the point cloud construction based on Legarda-Sáenz's

model becomes challenging and computationally intensive when considerable degree of lens distortions is present in the projector.

In this dissertation, we aimed at developing a system in which the calibration, i.e. the acquisition of system parameters, can be conducted more accurately and with less efforts. On the other hand, we also wanted to integrate the use of adaptive projection patterns into the system. To achieve these goals, we developed a new mathematical model for SMDFP systems as well as the associated algorithms for the construction of point clouds. These methods will be described in Chapter 3.

2.1.2 Models Based on Calibration Matrix

2.1.2.1 Overview

SMFP systems use phase-shifting method to acquire the absolute phase values of points on the surface being measured. With the absolute phase value as a third dimension, each 2-D point in the camera's image plane can uniquely identify a 3-D point in the measurement volume of SMFP system. In other words, each 3-D point in the measurement volume can be uniquely identified by its image coordinates and its absolute phase value.

The basic idea of calibration matrix based models can be described as the following: First of all, a 3-D matrix of discrete sampling points of the measurement volume is built, in which each element represents a 3-D point in the measurement volume with known (x, y, z) coordinates, absolute phase value and image coordinates; In a successive measurement, after the absolute phase values of points on the surface (being measured) are acquired, the corresponding (x, y, z) coordinates of these points can be calculated from interpolations of elements in the calibration matrix, based on their image coordinates and absolute phase values. That is, in contrast to models based on geometric approach, which computes the (x, y, z) coordinates of points by

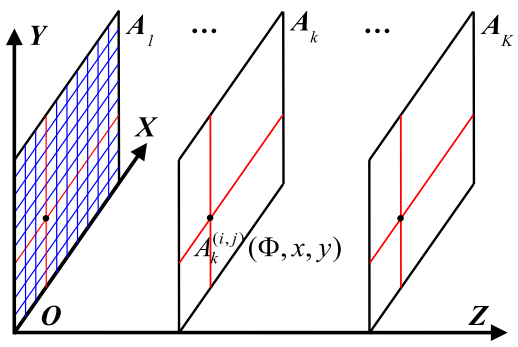


Figure 2.4: The calibration matrix defined in the model proposed by Sitnik et. al.

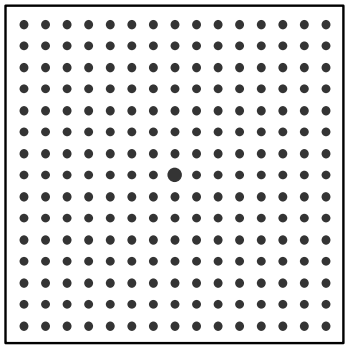


Figure 2.5: The calibration plate used in the construction of the calibration matrix

using equations established from optical geometry, calibration matrix based models use interpolations over existing sample points for the computation of the point cloud.

Calibration matrix based models have been successfully used on some commercial SMFP systems [18]. However, very few of them were published.

2.1.2.2 Model Proposed by Sitnik et. al.

Sitnik et. al. [29, 30] proposed a mathematical model based on the calibration matrix approach, which includes the definition of the matrix, a procedure and related algorithms for the construction of the matrix, and an algorithm for the computation of the point cloud that represents the surface been measured.

Sitnik's model defines the 3-D calibration matrix as a number (K) of 2-D matrices, denoted by $\boldsymbol{A}_k$ ($k = 1, \cdots, K$). Each 2-D matrix corresponds to a z-position in the coordinate frame defined (as shown in Fig. 2.4) and has a dimension of M by N, which is the same as the resolution of the camera. Each element in the matrices corresponds to a point in the measurement volume of the system, and has the (x, y, z) coordinates and the absolute phase value of that point. The column and row indices of an element are the image coordinates, (i, j), of its corresponding point.

The procedure for constructing the calibration matrix is as the following:

- A calibration plate is used in the construction of the 2-D matrices. It has an array of circular markers on it and the position of the markers are known precisely. A schematic diagram of the plate is shown in Fig. 2.5. The plate is mounted on a rail such that it can be translated along the rail and the direction of translation is perpendicular to the surface of the plate. The translation direction is also defined as the direction of the Z-axis, hence the plate is parallel to the X-Y plane for all the time. A reference point on the plate is taken as the point of $(x = 0, y = 0)$, and a reference position of the plate is taken as $z = 0$.

- The calibration plate is moved to a number (K) of different z-positions. At each position $z = z_k$ $(k = 1, \cdots, K)$, the corresponding 2-D matrix $\boldsymbol{A}_k$ is constructed. The absolute phase values for all elements in $\boldsymbol{A}_k$ are acquired by using phase-shifting method. An image of the calibration plate under normal lighting is also taken, from which the image coordinates of the center positions of the markers can be acquired. Since the positions of the markers are known, the (x, y, z) coordinates for matrix element $A_k(i, j)$ can then be calculated from interpolations of the markers' positions, since the points to be calculated and the markers are on the same plane. In Sitnik's model, a third order polynomial interpolation is used for the calculations.

At the measurement of an object, the absolute phase values of points on the object's surface are obtained by using phase-shifting method. The (x, y, z) coordinates of the points can then be computed from their phase values by using interpolations over the calibration matrix.

Consider a point on the surface, P, whose image coordinates are (i, j). In the calibration matrix, there are a total number of K points whose image coordinates are (i, j). The z-coordinates of these points and their absolute phase values, Φ, can be

fit into a polynomial function, $z = f_z^{(i,j)}(\Phi)$, in a least square sense. This function describes the relationship between the z-coordinates and the absolute phase values of all points in the measurement volume whose image coordinates equal to (i, j). Sitnik et. al. pointed out that a fifth order polynomial function can approximate this relationship to a good accuracy [29]. Once the approximation function $f_z^{(i,j)}(\Phi)$ is acquired, the z-coordinate of point P can be calculated from its absolute phase value, $\Phi(P)$. The computation of the x- and y-coordinates of P is conducted in a similar manner. The only difference is, the approximation functions, $f_x^{(i,j)}$ and $f_y^{(i,j)}$, are defined as functions of the z-coordinate, not the absolute phase value Φ. In Sitnik's approach, these two functions are selected to be linear functions.

2.1.2.3 Summary

Calibration matrix based models have a number of advantages, such as they have no constraints on system configuration and alignment, they are fairly tolerant to optical aberrations, and the implementation of the model and algorithms is relatively easy. On the other hand, calibration matrix based models also have disadvantages. Firstly, the structure of the 3-D calibration matrix has no modularization of system parameters. All parameters of the system are coupled and implicitly expressed in the calibration matrix. As a result, each time a system parameter has been changed, such as due to a change of an optical component, the complete calibration procedure has to be redone, which is time-consuming. Secondly, the computation of the point cloud is done by using polynomial approximation functions, which is not as accurate as optical geometry based models (see Section 2.1.1.4), especially when the configuration of the SMFP system involves wide-angle lenses and large perspective angles.

2.2 Phase-Shifting Method

2.2.1 Basic Idea

Phase-shifting method, also called phase-stepping method, is a technique that has been widely used in optical measurement fields, such as moiré interferometry [31, 32] and digital holography [33]. In SMFP technique, phase-shifting method is used to obtain the accurate phase values of points on the surface being measured, from which the 3-D positions of the points can be resolved. In the phase-shifting procedure, a sequence of sinusoidal fringe patterns (with selected phase-shift values) are projected on the surface being measured. Meanwhile, images of the surface under the projections are taken by the camera. From the image set acquired, a 2-D matrix of phase values can be calculated, with each element in the matrix corresponds to a pixel in the camera's image plane and hence a point[1] on the surface being measured. This matrix of phase values has the same dimension as the individual images acquired and is usually called the "phase map" of the surface.

There are many different phase-shifting algorithms available due to varied designs of phase-shift values for the sequence of fringe patterns. The basic idea of phase-shifting method, however, can be illustrated from the classic 4-step phase-shifting algorithm as the following:

- The 4-step phase-shifting algorithm takes a total number of four projection patterns, one "original" sinusoidal fringe pattern and three phase-shifted versions of the "original". For SMDFP technique, which uses digital projection, the projection patterns are constructed as grayscale bitmaps. The intensity

[1] Strictly speaking, each element in the matrix corresponds to a tiny area on the surface being measured, not a single point, just as a pixel in the image does.

distributions of the patterns can be described using the following equation:

$$I_n^{(P)}(x,y) = \frac{I_{max}^{(P)}}{2}\left[1+\sin\left(\frac{2\pi x}{p}+\frac{(n-1)\pi}{2}\right)\right], \quad n=1,\cdots,4 \tag{2.17}$$

where x and y are coordinates on the horizontal and the vertical axes of the bitmaps respectively, n represents the phase-shift step, $I_{max}^{(P)}$ is the maximum intensity in the bitmaps, p is the fringe pitch, and $I_n^{(P)}(x,y)$ is the intensity distribution of the n-th pattern. As can be seen that, the projection patterns defined by Eqn. 2.17 are vertical fringe patterns.

- Using the four fringe patterns as defined above for projections, the corresponding images of the surface being measured can be expressed using the following equation:

$$I_n(i,j) = A(i,j) + B(i,j)\,\sin\left(\Phi(i,j)+\frac{(n-1)\pi}{2}\right), \quad n=1,\cdots,4 \tag{2.18}$$

where (i,j) are the indices for pixels, and for a given pixel (i,j), $\Phi(i,j)$ is the pixel's absolute phase value, $A(i,j)$ and $B(i,j)$ are both constants for $n = 1,\cdots,4$, and $I_n(i,j)$ is the pixel's intensity in the n-th image.

- Using the four images obtained from the phase-shifting process, the "wrapped" phase map, $\phi(i,j)$, can be calculated from the following function:

$$\phi(i,j) = \mathbf{arctan}^*\left(\frac{I_1(i,j)-I_3(i,j)}{I_2(i,j)-I_4(i,j)}\right) \tag{2.19}$$

where the function $\mathbf{arctan}^*(\cdots)$ has *two* arguments and is defined as the fol-

lowing:

$$\textbf{arctan}^*\left(\frac{f}{g}\right) = \begin{cases} \arctan\left(\dfrac{f}{g}\right) , & \text{if } g \geq 0 \\ \arctan\left(\dfrac{f}{g}\right) + \pi , & \text{if } g < 0 \text{ and } f \geq 0 \\ \arctan\left(\dfrac{f}{g}\right) - \pi , & \text{if } g < 0 \text{ and } f < 0 \end{cases} \tag{2.20}$$

The *wrapped* phase map $\phi(i,j)$ computed from Eqn. 2.19 has a value range of $[-\pi, \pi]$ and is a 2π wrapping (hence the name) of the *absolute* phase map $\Phi(i,j)$, which has a much larger value range depending on the number of fringes in the projection patterns. The relationship between $\phi(i,j)$ and $\Phi(i,j)$ can be expressed using the following equation:

$$\phi(i,j) = \mathrm{mod}\big(\Phi(i,j)\ ,\ 2\pi\big) \tag{2.21}$$

For a N-step phase-shifting algorithm with uniform phase shifts, the projection patterns can be expressed using the following equation:

$$I_n^{(P)}(x,y) = \frac{I_{max}^{(P)}}{2}\left[1 + \sin\left(\frac{2\pi x}{p} + \frac{2\pi(n-1)}{N}\right)\right] , \quad n = 1, \cdots, N \tag{2.22}$$

Accordingly, the images of the surface under the projections can be written as

$$I_n(i,j) = A(i,j) + B(i,j)\sin\left(\Phi(i,j) + \frac{2\pi(n-1)}{N}\right) , \quad n = 1, \cdots, N \tag{2.23}$$

The equation for computation of the wrapped phase map is as the following:

$$\phi(i,j) = \textbf{arctan}^*\left(\frac{\sum_{n=1}^{N}\left[I_n(i,j)\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n(i,j)\sin\left(2\pi\frac{n-1}{N}\right)\right]}\right) \tag{2.24}$$

2.2.2 Design of Algorithms for Accurate Phase Measurement

Phase-shifting method assumes that the projection patterns generated are perfect sinusoidal fringe patterns with accurate phase shifts. However, this is generally not true in reality. Fringe patterns generated by grating or interferometry have non-sinusoidal waveforms and the phase shifts achieved are often not very precise. Similar problem could also happen to patterns generated by computer projectors, e.g. if the Gamma effect of the projector is not taken care of appropriately. As a result of the imperfect fringe patterns, the phase maps acquired from phase-shifting method are erroneous. In order to achieve high-accuracy phase measurement without requiring super accurate hardware and calibration, intensive research has been done in the design of phase-shifting algorithms to reduce or eliminate the influence of imperfections in fringe patterns.

Hibino et. al. [34] studied the influence of the non-sinusoidal waveform of fringe patterns and imprecise phase shifts to the accuracy of phase measurement. They pointed out that, in order to eliminate the effects of harmonic components in the waveform up to the jth order in the presence of a constance phase-shift error, at least $2j + 3$ fringe patterns are required and the phase-shift interval (between two consecutive patterns) must be less than $2\pi/(j+2)$. A general procedure for designing phase-shifting algorithms to achieve such goals was derived. Similar research has also been conducted by Surrel [35] and Joenathan [36] by using different analytical approaches.

Fringe patterns generated by computer projectors are pixelated and the intensities of pixels are quantized to a limited number of grayscales. As a result, the waveform of the fringe patterns is not a smooth sinusoidal curve but has a jagged profile, which could cause errors in the measured phase values. Coggrave and Huntley [37] studied this problem experimentally and found out that, by purposely defocusing the

projection on the surface being measured, the phase errors caused by the pixelation of fringe patterns can be reduced significantly.

Zhao and Surrel [38, 39] studied the influence of the quantization of image intensities to the accuracy of phase measurement. The relationship between the variance of phase errors and the variance of image intensity errors is formulated by using a characteristic polynomial method. It is shown that, the magnitude of the phase errors caused by image intensity quantization is also related with the modulated fringe intensity and the phase-shifting algorithm being used.

2.2.3 Summary

Phase-shifting method is a well developed technique for accurate phase measurement. Intensive research has been done on the error analysis of phase-shifting method, as well as the design of phase-shifting algorithms for achieving high accuracy. However, before the emergence of computer projectors, the development of phase-shifting algorithms was mainly focused on fringe patterns with *fixed* fringe pitches.

In this dissertation, the use of phase-shifting method is extended to fringe patterns with *variable* fringe pitches. Also, to satisfy the requirement on measurement speed in applications such as 100% on-line inspection, a new phase-shifting algorithm has been developed, which uses fewer projection patterns than existing algorithms but achieves similar measurement accuracy.

2.3 Phase Unwrapping Methods

In phase-shifting method, the sinusoidal fringe pattern with zero phase shift carries the absolute phase information that is defined implicitly on itself and its phase-shifted counterparts. Take the fringe patterns defined by Eqn. 2.22 as an example, $(2\pi x/p)$ is the phase distribution defined. Although the intensity distribution in a sinusoidal

fringe pattern repeats itself from fringe to fringe, the absolute phase values employed in one fringe do not overlap with the phase values in another fringe, due to the use of fringe order information. However, when using phase-shifting method for a phase measurement, the acquired phase map of the surface being measured is a *wrapped* phase map, which means the fringe order information has been lost. In order to recover the fringe orders and obtain the *absolute* phase map of the surface, a phase unwrapping process has to be performed.

The most popularly used phase unwrapping methods in SMFP technique can be put into two categories [40], namely spatial phase unwrapping methods and temporal phase unwrapping methods. In the following sections, these two classes of methods are explained individually.

2.3.1 Spatial Phase Unwrapping

2.3.1.1 Basic Idea

In a wrapped phase map, there exists many 2π phase jumps from one pixel to the adjacent pixel, which are caused by the 2π wrapping of phase values. Fig. 2.6 shows a grayscale rendering of a wrapped phase map, which is obtained from a measurement of the part shown in Fig. 1.2(a). In the figure, the grayscales represent phase values ranging from $-\pi$ to π, and the cyan color indicates areas that the phase information is either unavailable or not applicable. Places in the figure where the grayscales of pixels change abruptly from white to black indicate the occurrence of phase jumps.

The basic idea of spatial phase unwrapping is as the following: Firstly, the differences of phase values between adjacent pixels are examined throughout the phase map. The places where the phase differences are close to 2π are considered as possible phase jumps. Secondly, the phase values of pixels in the phase map are compensated by multiples of 2π so that the phase jumps are "flattened" and the phase distribu-

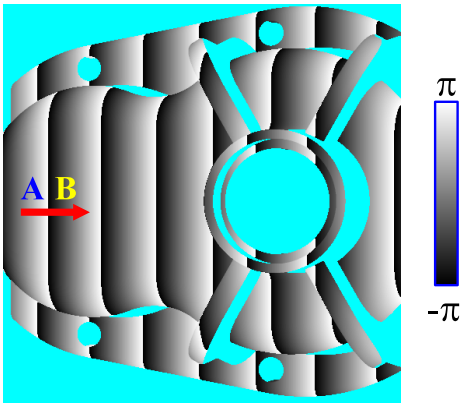


Figure 2.6: A grayscale rendering of a wrapped phase map

tion throughout the phase map is continuous in the valid regions. As a result of this step, the compensated phase map should agree with the absolute phase map only by a constant phase difference which is not known. Finally, by utilizing the reference pixels whose absolute phase values are known from other measurements, the absolute phase map can be obtained.

2.3.1.2 Problems with Measurement of Complex Surfaces

Based on the spatial phase unwrapping approach, a wrapped phase map can be partitioned into regions of three different types. Regions of the first type are those with no valid phase values, e.g. holes on the surface and shadowed areas that are caused by projection angle. Fig. 2.7 shows a topological diagram of a wrapped phase map, in which the regions with no valid phase values are denoted by $U_1, \ldots, U_k$. Separated by the U-type regions, the rest of the phase map, in which every pixel has a valid phase value, consists of a number of isolated regions. Let $S_1, \ldots, S_n$ denote these regions. The third type of regions is a subset of the S-type regions, which contains the reference pixels whose absolute phase values are known. These regions are denoted by $R_1, \ldots, R_m$.

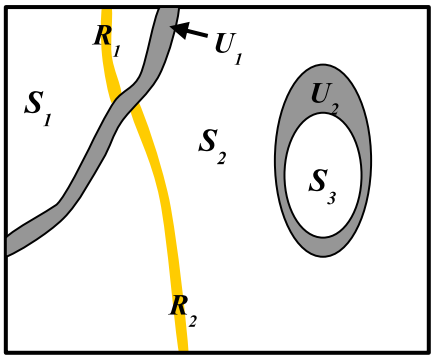


Figure 2.7: Topological diagram of a wrapped phase map

In order to successfully construct an absolute phase map from a wrapped phase map, each S-type region must contain at least one R-type region. Otherwise, the S-type region after phase unwrapping may still have an unknown constant phase difference compared to the true absolute phase map. The reference pixels in other S-type regions are not able to be utilized due to the isolation of the region. For spatial phase unwrapping approach, the R-type regions in the phase maps only spread over a limited area on the phase map (If the R-type regions spread over all S-type regions, the approach becomes temporal phase unwrapping). For example, the method for creation of reference pixels proposed in Ref. [24] is able to build a strip-shaped R-type region with a width of one fringe. When measuring a surface with unknown shape, the topological structure of the phase map acquired is unpredictable. Hence there is no guarantee that every S-type region include a R-type region. Hence, the phase unwrapping process may fail at some regions.

Besides the above limitation, spatial phase unwrapping also assumes that the difference of absolute phase values between adjacent pixels is less than π. A difference greater than π may be handled incorrectly. In reality, surfaces that are steep or discontinuous from the camera's perspective often cause phase jumps greater than π. In such cases, spatial phase unwrapping may fail.

2.3.1.3 Summary

When measuring continuous surfaces (from the camera's perspective) with fairly simple shapes, spatial phase unwrapping method works very well and requires fewer projection patterns than temporal phase unwrapping method. However, if the surface being measured has a complex shape or discontinuity, spatial phase unwrapping may fail. In this dissertation, measurement of surfaces with complex shapes are considered. Hence, spatial phase unwrapping is not used.

2.3.2 Temporal Phase Unwrapping

2.3.2.1 Overview

In spatial phase unwrapping method, the absolute phase value of a pixel can be determined only if the following two requirements are both satisfied: 1) At a certain point at least one of the pixel's neighbors has known absolute phase value; and 2) the difference between the absolute phase values of the pixel and its neighbor is less than π. If any of the two requirements are not satisfied, the spatial phase unwrapping method will fail.

Compared to spatial phase unwrapping method, temporal phase unwrapping does not rely on the knowledge of neighbor pixels' phase values to determine a pixel's absolute phase value. In stead, it calculates the absolute phase value by using the pixel's intensity values in a sequence of images, which are obtained by projecting a sequence of carefully designed fringe patterns.

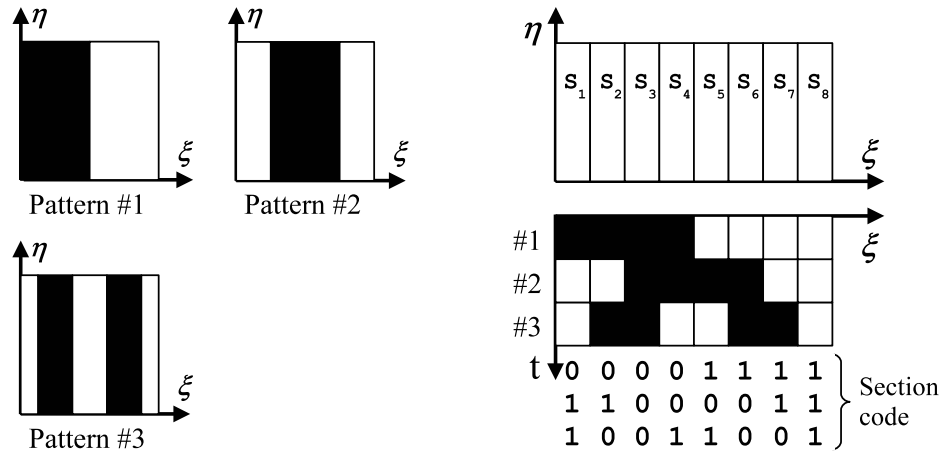
In the following sections, a number of successful and popular algorithms based on temporal phase unwrapping are presented.

2.3.2.2 Binary Code Method

Binary code method is a structured lighting technique which uses projection patterns with only black and white fringes [41]. Such fringe patterns are called binary fringe patterns and a few examples are shown in Fig. 2.8(a). Binary code method can be used by itself for phase measurement, or as a phase unwrapping tool for other phase measurement techniques, such as phase-shifting method [29]. The basic idea of binary code method is to construct a set of binary fringe patterns with different fringe arrangements such that the projection pattern space can be partitioned into a number of sections and each section can be uniquely identified by its binary intensities in the sequence of patterns.

An example of the projection patterns constructed by using binary code method is shown in Fig. 2.8, which are generated by the gray-code algorithm [42], a popularly used algorithm of binary code method. Depending on the resolution of phase measurement to be achieved, the number of projection patterns to be used may vary. The higher the resolution requirement, the more projection patterns are needed. Figure 2.8(a) shows 3 projection patterns only for demonstration purpose. By using the 3 patterns in a sequence, the projection pattern space, $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane, can be partitioned into 8 sections with each section having a unique intensity sequence (see Fig. 2.8(b)). When using "0" for indication of black fringes and "1" for indication of white fringes, a section's intensity sequence can be represented by a binary code, called section code. For example, when the patterns shown in Fig. 2.8(a) are used in the order of #1, #2 and #3, the projection intensities at section S_4 are black, black and white, consequently. The corresponding section code is 001. In phase measurement, the projection pattern space $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ also defines a distribution of absolute phase values. Therefore each section is associated with a certain range of absolute phase values.

By using more than 3 binary patterns, the projection pattern space can be par-



(a) Binary fringe patterns generated

(b) The resulting partitioning of the projection pattern space, $\boldsymbol{\xi}$-$\boldsymbol{\eta}$

Figure 2.8: Binary fringe patterns generated by the gray-code algorithm and the resulting partitioning of the projection pattern space

titioned to a finer resolution. Take the gray-code algorithm as an example. If a number of N_p patterns are used, the number of sections defined in the projection pattern space, $\boldsymbol{\xi}$-$\boldsymbol{\eta}$, could reach 2^{N_p}. When projecting these binary patterns to a surface for phase measurement, the corresponding images of the surface have dark and bright fringes. By performing an intensity thresholding, the grayscale images can be converted to 1-bit bitmaps. Therefore, with a sequence of N_p images, each pixel in the camera's image plane, which corresponds to a point on the surface being measured, can be associated with N_p binary values, forming a N_p-bit binary code. This binary code of the pixel is also the section code of a corresponding partition defined in the projection pattern space. Since every partition in the projection pattern space is related to a certain range of absolute phase values, the possible range of the pixel's absolute phase value can be determined. The phase unwrapping process can hence be done by adding multiples of 2π to the wrapped phase value of the pixel, which can be obtained from phase measurement techniques such as phase-shifting method, to make it consistent with the known range of the pixel's absolute phase value.

When using a fairly large number of binary patterns, usually more than 10, the binary code method can be used as a phase measurement technique by itself. In this case, the range of the absolute phase value of an arbitrary pixel given by the binary code method is small enough such that it can be used as a determined absolute phase value.

Different designs of binary projection patterns yield different accuracy, robustness and efficiency in phase measurement and phase unwrapping. Gärtner et. al. [43] gave the following benchmarks for evaluating the performance of a binary code algorithm:

- **Unique identification**: Every code in the set of section codes defined by a binary code algorithm needs to be unique, i.e. no duplicates of codes is allowed. An example can be found in Fig. 2.8, in which the section codes defined range from 000 to 111 without recurrence.

- **Ability of self-normalizing**: To retrieve the section codes for individual pixels, the grayscale images need to be converted to 1-bit bitmaps by intensity thresholding. Due to the influence of environment light and the reflection property of surfaces, the grayscale intensities of the dark and the bright fringes in the images may vary as the fringes are located in different regions of the image. Therefore, a fixed threshold value for thresholding the whole image may not work properly. For maximum robustness of the thresholding algorithm, the solution is to select a threshold value for each pixel based on its intensities w.r.t. the projection of the black and the white fringes respectively. This requires that, for a set of binary patterns defined, every point is in a black fringe for a certain pattern and in a white fringe for another pattern. This means that, section codes with all "0"s or all "1"s are not allowed.

- **The Hamming-distance of adjacent section codes needs to be equal to 1**: The Hamming-distance of two codes is defined as the number of disagreed bits

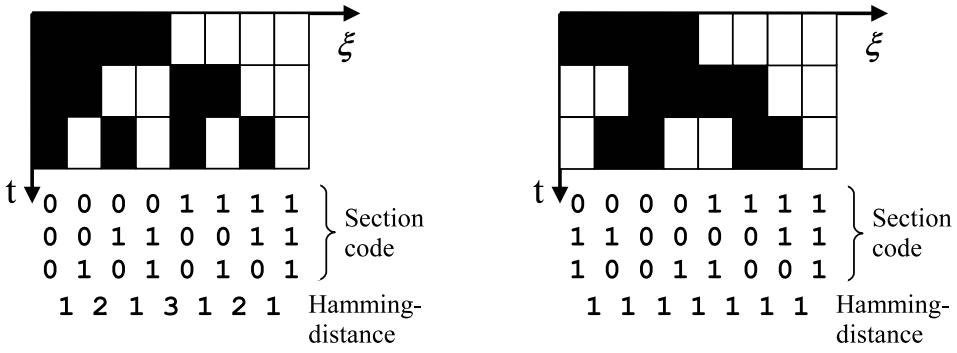
when comparing two codes in a bitwise manner. For example, the Hamming-distance between "10011001" and "10111011" is 2 because two bits in the codes, the 3rd and the 7th, are different.

The requirement on the Hamming-distance of adjacent section codes comes from the following concern: Due to the finite size of pixels, it happens often that one pixel is under the projection of a black fringe and a white fringe at the same time, i.e. the boundary of the black and the white fringe falls somewhere inside the pixel. In such case, the pixel has a grayscale intensity between the maximum and the minimum intensities it could get. Depending on which fringe is dominant on the pixel as well as the level of noise, the binarized intensity can be either "0" or "1", which leads to an uncertainty in the corresponding phase range calculated. However, if the Hamming-distance of adjacent section codes is always 1 and there is only one ambiguous bit in the retrieved binary code, which is the most frequent case, the phase range with uncertainty considered is guaranteed to be double of the regular range. For larger Hamming-distances, the magnitude of uncertainty is larger.

Figure 2.9 shows the Hamming-distances of adjacent section codes generated by the dual-code algorithm and the gray-code algorithm. As we can see, the Hamming-distances of the gray-code algorithm is always 1, which is one of the properties that make it a favorable algorithm for binary code method.

Gärtner et. al. [43] also discussed the efficiency of different binary code algorithms, where efficiency is defined as the number of sections in the projection pattern space versus the number of binary patterns used. A few algorithms with high efficiency, e.g. the extended Maximum Zero X-ing code and the Lemming code, were presented in their paper.

Binary fringe patterns can be generated by glass grating devices with high pre-



(a) Hamming-distance achieved by the dual-code algorithm

(b) Hamming-distance achieved by the gray-code algorithm

Figure 2.9: The Hamming-distance of binary patterns achieved by different algorithms

cision and high resolution [1]. Therefore, binary code method is a favorable phase measurement technique for SMFP systems with glass grating based projection units. For SMFP systems based on computer projectors, there are other phase unwrapping methods which require fewer projection patterns than binary code method. Hence, binary code method is used in this dissertation.

2.3.2.3 Algorithm Proposed by Huntley and Saldner

Huntley and Saldner [44, 45, 46] developed a temporal phase unwrapping algorithm which uses a number of sinusoidal fringe patterns with different fringe pitches. The intensity distributions of the patterns can be expressed using the following equation:

$$I^{(P)}(\xi,\eta,t,n) = \frac{I^{(P)}_{max}}{2}\left[1+\sin\left(2\pi t\big(\xi-1/2\big)+\frac{(n-1)\pi}{2}\right)\right] \qquad (2.25)$$
$$\xi,\eta\in[0,1]\;;\quad t=1,\cdots,T\;;\quad n=1,\cdots,4$$

where (ξ,η) are the normalized coordinates in the projection pattern space $\boldsymbol{\xi}$-$\boldsymbol{\eta}$, t is the number of fringes in a pattern, n represents the step of phase shift, $I^{(P)}_{max}$ is the maximum intensity of projection, and $I^{(P)}(\xi,\eta,t,n)$ is the projection intensity

of point(ξ, η) in the pattern (t, n). The intensity distributions of the patterns also defined the distributions of the absolute phase values. Let $\Phi(\xi, \eta, t)$ denote the phase distribution in pattern $(t, 1)$, which has zero phase shift. $\Phi(\xi, \eta, t)$ can be written as the following:

$$\Phi(\xi, \eta, t) = 2\pi t(\xi - 1/2) \tag{2.26}$$

By projecting the fringe patterns on the surface being measured, a set of images can be acquired. Let $I(i, j, t, n)$ denote the image that corresponds to the pattern (t, n), where (i, j) are the pixel coordinates in the image. From the 4 images that correspond to the same fringe number t, the associated phase map of the surface, $\Phi(i, j, t)$, satisfies the following equation:

$$\tan\big(\Phi(i, j, t)\big) = \frac{I(i, j, t, 1) - I(i, j, t, 3)}{I(i, j, t, 2) - I(i, j, t, 4)} \tag{2.27}$$

For every pixel (i, j), the difference between its phase values $\Phi(i, j, t)$ and $\Phi(i, j, t-1)$ can be derived as the following:

$$\begin{aligned} \Delta\Phi(i, j, t) &= \Phi(i, j, t) - \Phi(i, j, t-1) \\ &= 2\pi t\big[\xi_{i,j} - 1/2\big] - 2\pi(t-1)\big[\xi_{i,j} - 1/2\big] \\ &= 2\pi\big[\xi_{i,j} - 1/2\big] \end{aligned} \tag{2.28}$$

where $\xi_{i,j}$ is the ξ-coordinate in the projection pattern space that corresponds to pixel (i, j). Since the value of $\xi_{i,j}$ is in the range of $[0, 1]$, we can get that $\Delta\Phi(i, j, t)$ is in the range of $[-\pi, \pi]$. Therefore, by utilizing Eqn. 2.27, the following equation for computing $\Delta\Phi(i, j, t)$ can be derived:

$$\Delta\Phi(i, j, t) = \arctan\left(\frac{\tan\big(\Phi(i, j, t)\big) - \tan\big(\Phi(i, j, t-1)\big)}{1 + \tan\big(\Phi(i, j, t)\big) \cdot \tan\big(\Phi(i, j, t-1)\big)}\right) \tag{2.29}$$

The phase map that corresponds to the fringe patterns with the highest fringe number, $\Phi(i, j, T)$, can be calculated from the following equation:

$$\Phi(i, j, T) = \Phi(i, j, 1) + \sum_{t=2}^{T} \Delta\Phi(i, j, t) \tag{2.30}$$

where $\Phi(i, j, 1)$ can be computed from Eqn. 2.27 with an inverse tangent operation on both sides, since $\Phi(i, j, 1)$ is in the range of $[-\pi, \pi]$ and therefore no ambiguity is involved. The phase map $\Phi(i, j, T)$ has the highest phase accuracy and is the final result of the phase measurement.

Compared to the binary code method, the phase unwrapping algorithm proposed by Huntley and Saldner is more favorable to SMFP systems using computer projectors, because it fully utilizes the projection intensity range of the projector and hence yields higher phase measurement accuracy. However, this algorithm requires a large number of fringe patterns for a measurement, which makes it unsuitable for applications that demand a fast measurement speed.

2.3.2.4 Algorithm Proposed by Zhao et. al.

Zhao et. al. [47] proposed a phase unwrapping algorithm which uses fewer fringe patterns than the one developed by Huntley and Saldner (see Section 2.3.2.3) while achieves similar performance on phase unwrapping. The basic idea of their algorithm is to use two (or more) phase maps with different phase sensitivities, which are obtained by using sinusoidal fringe patterns with different fringe pitches. A detailed description of the algorithm is as following:

- Firstly, a wrapped phase map with high phase sensitivity, denoted by $\phi(i, j)$, is obtained by using the phase-shifting method and fringe patterns with a small fringe pitch. The relationship between $\phi(i, j)$ and its unwrapped counterpart,

$\Phi(i,j)$, can be expressed using the following equation:

$$\Phi(i,j) = \phi(i,j) + 2\pi\, s(i,j) \tag{2.31}$$

where (i,j) are the pixel coordinates in the image and $s(i,j)$ are unknown integers to be solved.

- The other phase map, $\Phi_0(i,j)$, is also obtained from phase-shifting but using fringe patterns with only one sinusoidal fringe. Therefore the acquired phase map $\Phi_0(i,j)$ is an absolute phase map without the phase wrapping problem. By applying certain restrictions to the fringe patterns used for the acquisition of $\Phi(i,j)$ and $\Phi_0(i,j)$, the following relationship between the two phase maps can be achieved:

$$\Phi(i,j) = c_{h,0}\, \Phi_0(i,j) \tag{2.32}$$

where $c_{h,0}$ is a constant. By combining Eqns. 2.31 and 2.32, the equation for solving $s(i,j)$ can be derived:

$$s(i,j) = \text{round}\left(\frac{c_{h,0}\, \Phi_0(i,j) - \phi(i,j)}{2\pi}\right) \tag{2.33}$$

- Once $s(i,j)$ is known, the absolute phase map $\Phi(i,j)$ can then be calculated by substituting $s(i,j)$ into Eqn. 2.31.

This phase unwrapping algorithm can also be extended to using 3 or more phase maps with different phase sensitivities for an improved immunity to phase errors. Due to its many attractive properties, such as excellent phase unwrapping performance, easy implementation and the requirement of a small number of projection patterns, the algorithm proposed by Zhao et. al. has been used in many latest SMFP systems [3, 48].

2.3.2.5 Summary

Compared to spatial phase unwrapping approach, temporal phase unwrapping methods are able to handle surfaces with more complicated shapes and are less subject to phase errors, although they generally require more projection patterns. Therefore, temporal phase unwrapping methods are more popularly used in recent SMFP systems.

In this dissertation, a modified version of Zhao's algorithm is proposed, which uses fewer projection patterns but achieves similar performance.

2.4 Camera Models and Camera Calibration Methods

2.4.1 Camera Models

A camera model is a set of mathematical equations and related parameters for describing the geometric relationship between the 3-D scene and its 2-D image captured by a camera. The most commonly used camera model is the pinhole camera model, which describes the formation of image as a perspective projection from 3-D space to a 2-D image plane. It is simple and reasonably accurate, hence it has been used in many camera-related applications or been taken as a starting point for advanced modeling of cameras. In the research of SMFP technique, many of the existing mathematical models for SMFP systems have used the pinhole camera model for describing cameras and projectors [22, 24].

The pinhole camera model does not consider lens distortions (e.g. radial distortion, de-centering distortion and thin prism distortion), which exist in most real cameras. Advanced camera models provide more accurate modeling of cameras by

considering the influence of lens distortions to image formation. A common approach is to use a lens distortion model which can be integrated into the basic pinhole camera model. A popularly used lens distortion model was proposed by Brown [49, 50], for which a detailed mathematical description is given in Section 3.2.1. The pinhole camera model with Brown's lens distortion model has been proved to be very accurate and hence used in many camera-based 3-D measurement techniques, such as photogrammetry [8].

In this dissertation, the pinhole camera model with Brown's lens distortion model is used in the proposed mathematical model for SMDFP systems.

2.4.2 Camera Calibration Methods

A camera model defines a number of parameters that are related to the optical or geometric features of the camera's lens and/or imaging sensor. These parameters are generally unable to be measured directly. For different cameras, i.e. different lens and imaging sensor configurations, the values of parameters are usually different.

Camera calibration is a procedure to estimate the values of parameters (defined by the camera model) for individual cameras [52, 53]. Traditional camera calibration methods require special equipments or facilities, such as light collimator or plumbing lines [50]. Since late 80's, more convenient and flexible calibration methods have been developed, e.g. requiring only a few images of a calibration object taken by the camera from multiple perspectives [54].

Zhang [27] proposed a calibration method which estimates camera parameters, including lens distortion coefficients, by taking pictures of a planar pattern at multiple orientations. The relative positions of the feature points in the pattern need to be known, while the orientations of the pattern can be arbitrary and are not required to be known a priori. The computation of the parameter values is done in two steps.

At the first step, a least-square method is used to obtain the transformation matrix which relates the 3-D positions of the feature points to their 2-D projections in the images. At the second step, a non-linear optimization process is performed to find out the set of parameter values that gives the minimum residual error after projection.

Heikkilä and Silvén [55] extended the two-step approach proposed by Zhang and added compensations for the extraction of feature points and correction of the distorted image coordinates. A detailed description of their approach by using a calibration plate with circular feature points can be found in the literature [26].

In our research, we used the camera calibration algorithm and software library developed by the computer vision research group at the California Institute of Technology [56, 57].

Chapter 3

System Model and Algorithms

This chapter provides detailed descriptions of the mathematical model and algorithms developed for SMDFP [63, 64]. Section 3.1 explains the goals, state of the art and challenges in the development of models and algorithms. Section 3.2 explains the camera model and the lens distortion model used, which are required to understand the material presented in the rest of the chapter. Section 3.3 describes the proposed mathematical model for SMDFP and the related algorithms for shape measurement, which include the algorithm for construction of absolute phase map (Section 3.3.3), algorithm for construction of point cloud (Section 3.3.4), and algorithm for estimation of sensor parameters (Section 3.3.5). The implementation of the model and algorithms is described in Section 3.4 and the results are presented. Section 3.5 finally summarizes this chapter.

3.1 Introduction

As an emerging technique for 3-D measurements, SMDFP is still in a development stage and its potentials have not been fully exploited. To make it more attractive to industrial 3-D measurement applications, we need to looking for improvements in the

following aspects: measurement accuracy, measurement speed, ease of calibration, and the capability of handling complex shapes.

One way to achieve these goals is to improve the hardware. For example, to use projectors and cameras with higher resolution and higher signal-to-noise ratio to improve measurement accuracy; and to employ better synchronization between the projector and the camera to reduce measurement time [65]. However, improving the hardware is not always helpful and on the other hand it may increase system cost as well. Another approach for achieving these goals is to develop better mathematical models and algorithms for SMDFP systems.

The mathematical model used for a SMDFP system and the related algorithms for construction of point clouds are crucial to the system's measurement accuracy. As discussed in Section 2.1, existing models can be classified into two categories, namely (optical) geometry based models and calibration matrix based models [18]. Between the two, geometry based models have been more popularly used and studied, mainly because they have the potential to describe the optical physics of SMDFP systems with higher accuracy. A very promising model based on geometric approach was proposed by Legarda-Sáenz et. al. recently [19]. By fully considering the 3-D perspectives and lens distortions of the projector and the camera, Legarda-Sáenz's model is able to describe the optical geometry of SMDFP systems very accurately. However, this model involves a large number of parameters (30 in total) and an accurate acquisition of these parameters requires a complex calibration process. To make the calibration process easier while maintaining the accuracy achieved, improvements need to be made to the model and new algorithms need to be developed.

In a shape measurement performed by SMDFP system, the projector needs to project a sequence of light patterns and the camera needs to capture the images accordingly. Besides the hardware issues (e.g. the refreshing rate of the projector, the exposure time of the camera and the synchronization between the projector and the

camera), the time spent on the acquisition process is determined by the number of projection patterns used. For time-critical applications such on 100% on-line inspection of parts, it is desirable to have a minimal measurement time. Hence, methods that involve fewer projection patterns are preferred. Unfortunately, existing methods that use a small number of patterns (typically 4 or 5) are unable to handle complex shapes [24]. On the other hand, methods such as binary code [29] and phase-shifting with multiple fringe frequencies [3, 48] have exellent capability of handling complex shapes but they generally require more patterns (typically 12 or more). After a careful study of the phase-shifting method with multiple fringe frequencies, we discovered that images acquired by using this method contain redundant information which may be utilized to reduce the number projection patterns needed.

As a summary of the above, improvements of SMDFP systems in terms of measurement accuracy, speed and ease of calibration require the development of new model and algorithms.

3.2 Mathematical Preliminaries

3.2.1 Pinhole Camera Model with Lens Distortion Model

The pinhole camera model with modeling of lens distortions has been shown to be an accurate mathematical model for cameras. It is widely used in areas such as photogrammetry, machine vision and computer graphics [66]. The pinhole camera model with lens distortion model can be described by a series of transformations presented below (see Fig. 3.1):

1. **Transformation from the world coordinate frame to the camera coordinate frame**: In the pinhole model, each camera defines a camera coordinate frame (Cartesian), which is determined by the cameras optics and image for-

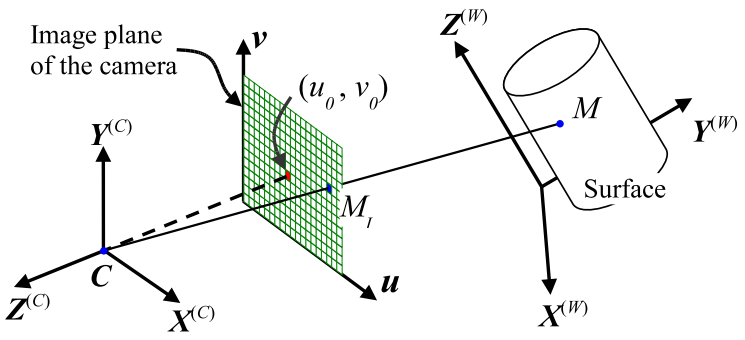


Figure 3.1: Pinhole camera model

mation hardware, e.g. the CCD. The coordinate transformation between the world coordinate frame and the camera coordinate frame can be described by a rotation matrix $\boldsymbol{R}_{W2C}$ and a translation vector $\boldsymbol{T}_{W2C}$:

$$\left[x_M^{(C)}, y_M^{(C)}, z_M^{(C)}\right]^T = \boldsymbol{R}_{W2C}\left[x_M^{(W)}, y_M^{(W)}, z_M^{(W)}\right]^T + \boldsymbol{T}_{W2C} \tag{3.1}$$

where $\left[x_M^{(W)}, y_M^{(W)}, z_M^{(W)}\right]^T$ is the coordinates of point M in the world coordinate frame and $\left[x_M^{(C)}, y_M^{(C)}, z_M^{(C)}\right]^T$ is the coordinates of M in the camera coordinate frame.

2. **Perspective projection to the image plane**: As a convention, the image plane of the perspective projection is defined as the plane perpendicular to $Z^{(C)}$-axis and intersecting it at $z^{(C)} = -1$. The two axes of the image plane, $\boldsymbol{u}$ and $\boldsymbol{v}$, are parallel to $X^{(C)}$- and $Y^{(C)}$-axis respectively. The principal point of the perspective projection is defined as the intersection of $Z^{(C)}$-axis and the image plane, whose pixel coordinates are (u_0, v_0). Let M_I denote the projection of point M on the image plane. The coordinates of M_I in the camera coordinate

frame, i.e. $\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T$, can be calculated as follows:

$$\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T = \left[\frac{x_M^{(C)}}{-z_M^{(C)}}, \frac{y_M^{(C)}}{-z_M^{(C)}}, \frac{z_M^{(C)}}{-z_M^{(C)}}\right]^T \tag{3.2}$$

3. **Lens distortion model**: Due to the distortions of optical lenses, real cameras do not comply with perspective projection perfectly. The real position of M_I is generally not at the nominal coordinates, i.e. $\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T$, but shifted a bit in the $\boldsymbol{u}$–$\boldsymbol{v}$ plane and ends up at $\left[x_M^{(d)}, y_M^{(d)}, -1\right]^T$. The most popularly used lens distortion model in photogrammetry is the one introduced by Brown [49, 50], in which the relationship between the distorted projection position, $\left[x_M^{(d)}, y_M^{(d)}, -1\right]^T$, and the idealized projection position, $\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T$, is defined as follows:

$$\begin{aligned}\begin{bmatrix} x_M^{(d)} \\ y_M^{(d)} \end{bmatrix} = &\left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{bmatrix} x_M^{(n)} \\ y_M^{(n)} \end{bmatrix} + \\ &\begin{bmatrix} 2k_3 x_M^{(n)} y_M^{(n)} + k_4 \left(r^2 + 2\left(x_M^{(n)}\right)^2\right) \\ k_3 \left(r^2 + 2\left(y_M^{(n)}\right)^2\right) + 2k_4 x_M^{(n)} y_M^{(n)} \end{bmatrix}\end{aligned} \tag{3.3}$$

where $r^2 = \left(x_M^{(n)}\right)^2 + \left(y_M^{(n)}\right)^2$ and k_s $(s = 1, \ldots, 5)$ are the radial and tangential distortion coefficients of the lenses.

4. **Transformation from camera coordinates to pixel coordinates**: For a digital camera (typically CCD or CMOS based), the position of point M_I is digitized by the CCD/CMOS chip and presented in pixel coordinates. Denote $\left[x_M^{(p)}, y_M^{(p)}, 1\right]^T$ as the homogeneous pixel coordinates of M_I, which can be cal-

culated from the following equation:

$$\begin{bmatrix} x_M^{(p)} \\ y_M^{(p)} \\ 1 \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} x_M^{(d)} \\ y_M^{(d)} \\ 1 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} f_x & \alpha f_x & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

where $\boldsymbol{A}$ is the camera's intrinsic matrix, in which (u_0, v_0) is the pixel coordinates of the principal point, f_x and f_y are the scale factors for axes $\boldsymbol{u}$ and $\boldsymbol{v}$ respectively, and α is a coefficient describing the skewness of the axes $\boldsymbol{u}$ and $\boldsymbol{v}$.

3.2.2 Inverse Transformations of Pinhole Camera Model

The SMFP technique involves the construction of 3-D point clouds from 2-D images, which often requires the calculation of the world coordinates of points from their pixel coordinates in the images. Hence, the inverse transformations of pinhole camera model are often needed. In the following, mathematical descriptions of the inverse transformations are given:

- **From pixel coordinates to camera coordinates**: The position of point M_I (with lens distortions) on the image plane can be calculated from its pixel coordinates by using the inverse camera intrinsic matrix:

$$\begin{bmatrix} x_M^{(d)} \\ y_M^{(d)} \\ 1 \end{bmatrix} = \boldsymbol{A}^{-1} \begin{bmatrix} x_M^{(p)} \\ y_M^{(p)} \\ 1 \end{bmatrix} \tag{3.5}$$

  where $\left(x_M^{(d)}, y_M^{(d)}\right)$ are in camera coordinates.

- **The idealized projection position**: With the lens distortion model given in the previous section, the conversion of a point's idealized projection position to

its real projection position cannot be reversed analytically. That is, there is no analytical forms for $x_M^{(n)}$ and $y_M^{(n)}$ by taking $x_M^{(d)}$ and $y_M^{(d)}$ as arguments, since Eqn 3.3 cannot be inversed. However, $\left(x_M^{(n)}, y_M^{(n)}\right)$ can be solved from Eqn 3.3 by using numerical methods, such as Newton iteration.

- **The inverse of perspective projection**: The 3-D position of a point M in the space cannot be decided by merely knowing its projection position M_I on the image plane, because one dimensional information is missing. In the pinhole camera model, a point M_I on the image plane defines a ray that starts from the projection center C and passes through M_I. The corresponding point M in the 3-D space lies somewhere on the ray CM_I. For SMFP technique, the depth information of M is given by its absolute phase value.

- **Transformation from the camera coordinate frame to the world coordinate frame**: The transformation of the coordinates of a point, M, from the camera coordinate frame to the world coordinate frame can be done as the following:

$$\begin{bmatrix} x_M^{(W)} \\ y_M^{(W)} \\ z_M^{(W)} \end{bmatrix} = \boldsymbol{R}_{W2C}^{-1} \left(\begin{bmatrix} x_M^{(C)} \\ y_M^{(C)} \\ z_M^{(C)} \end{bmatrix} - \boldsymbol{T}_{W2C} \right) \tag{3.6}$$

  where $\boldsymbol{R}_{W2C}$ is the rotation matrix and $\boldsymbol{T}_{W2C}$ is the translation vector.

3.3 Model and Algorithms

3.3.1 Overview

In the following sections, the proposed mathematical model for SMDFP systems and the related algorithms are explained in detail.

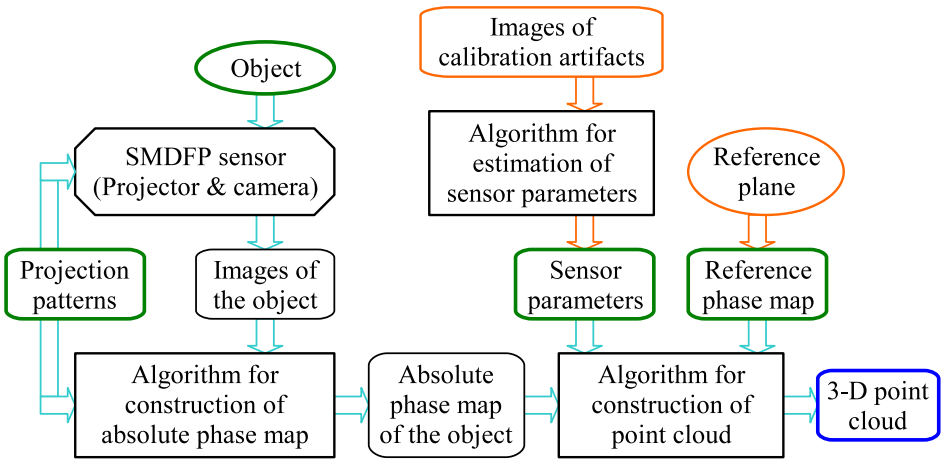


Figure 3.2: Schematic diagram of the relationship of algorithms involved in SMDFP

Section 3.3.2 describes the mathematical model, which provides an accurate modeling of the optical geometry of SMDFP systems by fully considering the influence of 3-D perspectives and the lens distortions of the projector and the camera.

Based on the proposed mathematical model, three algorithms were developed, namely the algorithm for construction of absolute phase map, the algorithm for construction of point cloud and the algorithm for estimation of sensor parameters. The relationship between the algorithms is illustrated in Fig. 3.2.

Section 3.3.3 explains the algorithm for construction of absolute phase map. This algorithm takes the images of an object, which are obtained under the projections of phase-shifted fringe patterns, and generates an absolute phase map of the object. By fully exploiting the information stored in the images from phase-shifting, the proposed algorithm uses fewer projection patterns than existing algorithms without compromising performance.

Section 3.3.4 explains the algorithm for construction of point cloud. This algorithm takes three inputs: the absolute phase map of the object, the sensor parameters and a reference phase map. The output of the algorithm is a dense 3-D point cloud

that represents the surface of the object being measured. With the use of a reference phase map, the proposed algorithm simplifies the calibration process of SMDFP systems and handles the projector's lens distortions automatically.

Finally, Section 3.3.5 explains the algorithm for accurate estimation of the sensor parameters, which are used by the point cloud construction algorithm.

3.3.2 Mathematical Model for SMDFP Systems

The mathematical model used for describing SMDFP systems consists of two parts, a camera model and a projector model.

- **Camera model**: The camera model describes the geometric relationship between the 3-D shapes of objects and their 2-D images in the camera. The pinhole camera model with modeling of lens distortions, as explained in Section 3.2.1, is used.

- **Projector model**: The projector model describes the geometric relationship between a 2-D projection pattern and the resulting light intensity distribution in the 3-D space. Since a computer projector (LCD or DMD based) acts as an inverted digital camera from the optical geometry perspective[1], it can also be modeled accurately by using the pinhole camera model (with lens distortion model) as for the camera. By using the pinhole camera model, a Cartesian coordinate frame $X^{(P)}Y^{(P)}Z^{(P)}$ is defined for the projector (see Fig. 3.3). The image plane defined by $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ axes represents the DMD/LCD chip of the projector. The pixel coordinates of the principal point in the image plane is (ξ_0, η_0). All transformations related to the projector model are analogous to the ones defined for the camera (see Section 3.2.1 for details).

[1]For digital cameras, lights from outside the camera pass through lenses and hit the CCD/CMOS sensor, where image is formed; While for the case of computer projectors, lights reflected by DMD chip (or transiting via LCD) pass through lenses and hit objects in the 3-D space.

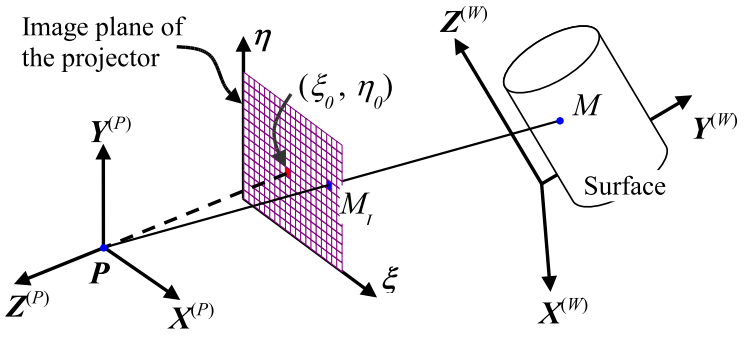


Figure 3.3: A computer projector can be described by the pinhole camera model

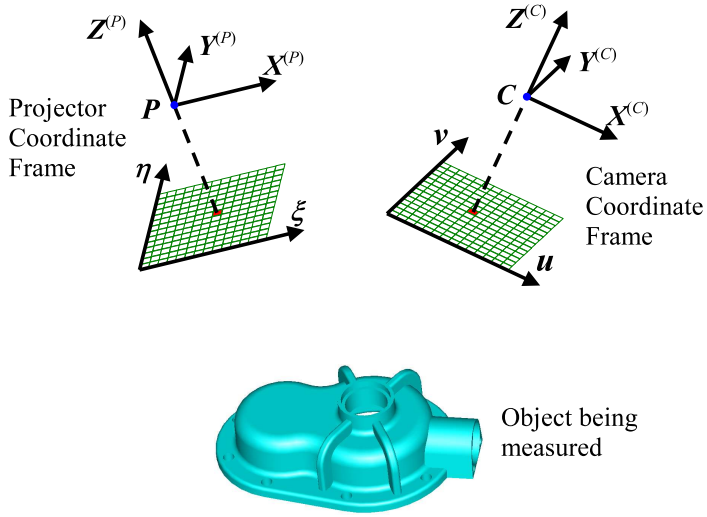
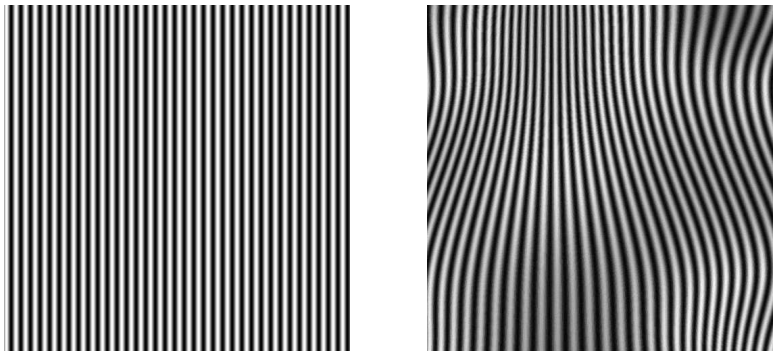


Figure 3.4: Mathematical model for SMDFP systems



(a) Sinusoidal fringe pattern with fixed fringe pitch

(b) Generalized fringe pattern

Figure 3.5: Generalized fringe pattern vs. sinusoidal fringe pattern

The complete mathematical model for SMDFP systems with one projector and one camera is illustrated in Fig. 3.4. Compared to some of the existing models based on optical geometry [22, 24], this model is more accurate due to the use of the full pinhole camera model and the modeling of lens distortions. It also has the minimum constraints on system alignment, which allows maximal flexibility in system setup. For example, it does not require the optical axes of the projector and the camera to be coplanar, which is an assumption made by many existing models.

3.3.3 Algorithm for Construction of Absolute Phase Map

3.3.3.1 Phase-shifting Using Generalized Fringe Patterns

The concept of generalized fringe pattern is introduced to distinguish itself from the traditional sinusoidal fringe pattern (with fixed fringe pitch). A sinusoidal fringe pattern contains straight fringes and has a sinusoidal intensity profile at cross-sections, as the one shown in Fig. 3.5(a). A generalized fringe pattern is a fringe pattern in which the fringes can be curved and the fringe pitch can vary continuously throughout the pattern. An example of generalized fringe pattern is shown in Fig. 3.5(b). The sinusoidal fringe pattern is a special case of generalized fringe pattern.

The proposed algorithm for construction of absolute phase maps uses generalized fringe patterns. The conventional phase-shifting technique is revised to incorporate the use of generalized fringe patterns. By using a new mathematical description, the generation of a set of projection patterns for absolute phase measurement using phase-shifting method can be presented as a three-step process as follows:

- **Construction of phase function**: A phase function $\Phi^{(P)}(\xi, \eta)$ is defined in the image plane of the projector, i.e. the $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ plane. $\Phi^{(P)}(\xi, \eta)$ must be continuous in the $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ plane and monotonic in either $\boldsymbol{\xi}$ or $\boldsymbol{\eta}$ direction, depending on the position of the camera w.r.t. the projector. These two constraints are required by the algorithm for point cloud construction, which will be explained in Section 3.3.4.

- **Conversion of the phase function to fringe pattern**: The phase function $\Phi^{(P)}(\xi, \eta)$ is converted to light projection pattern by certain encoding method. Light properties that can be used for encoding include intensity, color [67], etc. In practice, light intensity, without the involvement of color, is most popularly used since the intensity of light can be measured accurately by photo sensors such as CCD. A widely used modulation function for converting phase values to light intensities is the sinusoidal function. For digital projectors, in which light intensity is presented in grayscales, the sinusoidal modulation can be described by the following equation:

$$I^{(P)}(\xi, \eta) = \frac{I^{(P)}_{max}}{2}\Big[1 + \sin\big(\Phi^{(P)}(\xi, \eta)\big)\Big] \tag{3.7}$$

  where $I^{(P)}(\xi, \eta)$ is the grayscale value of point (ξ, η) in the projection pattern and $I^{(P)}_{max}$ is the maximum grayscale in the pattern.

  Sinusoidal function has features that make it particularly suitable as the modu-

lation function for phase-shifting method. For example, if the projection is out-of-focus at the object's surface, the light intensity distribution of the projection pattern will be locally averaged over the surface. However, the intensity change would not affect the phase values restored by the phase recovery algorithm, i.e. the phase-shifting technique can still resolve the phase values correctly despite the out-of-focus projection. The mathematical explanation for this is, a "locally averaged" sinusoidal function will remain as a sinusoidal function with the same wavelength and phase offset. The only thing that will change is the amplitude. In this dissertation, only sinusoidal modulation of phase functions is discussed.

By using the sinusoidal modulation, a large variety of fringe patterns can be defined by appropriately constructed phase functions. For example, a vertical sinusoidal fringe pattern (as the one shown in Fig. 3.5(a)) can be defined by the following phase function:

$$\Phi^{(P)}(\xi, \eta) = c \cdot \xi \tag{3.8}$$

where c is a constant.

- **Phase-shifting of the fringe pattern**: The phase-shifting process of a fringe pattern is to add constant phase-shift value(s) in the phase function $\Phi^{(P)}(\xi, \eta)$, in the whole $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ space defined, for a sequence of steps.

  Take the N-step phase-shifting algorithm with uniform phase shifts as an example. The phase-shifted projection patterns can be described using the following equation:

$$I_n^{(P)}(\xi, \eta) = \frac{I_{max}^{(P)}}{2} \left[1 + \sin \left(\Phi^{(P)}(\xi, \eta) + \frac{2\pi(n-1)}{N} \right) \right] , \quad n = 1, \cdots, N \tag{3.9}$$

  Accordingly, the images of the surface under the projections can be written as

follows (a detailed proof is presented in Appendix A.1):

$$I_n(i,j) = A(i,j) + B(i,j)\,\sin\left(\Phi(i,j) + \frac{2\pi(n-1)}{N}\right)\ , \quad n = 1, \cdots, N \tag{3.10}$$

where (i,j) are the pixel coordinates, and for a given pixel (i,j), $\Phi(i,j)$ is the pixel's "base" absolute phase value, $A(i,j)$ and $B(i,j)$ are both constants (for $n = 1, \cdots, N$), and $I_n(i,j)$ is the pixel's intensity in the n-th image.

The equation for computing the wrapped phase map is as the following:

$$\phi(i,j) = \mathbf{arctan}^*\left(\frac{\sum_{n=1}^{N}\left[I_n(i,j)\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n(i,j)\sin\left(2\pi\frac{n-1}{N}\right)\right]}\right) \tag{3.11}$$

where the function $\mathbf{arctan}^*(\cdots)$ has *two* arguments and is defined as follows:

$$\mathbf{arctan}^*\left(\frac{f}{g}\right) = \begin{cases} \arctan\left(\dfrac{f}{g}\right)\ , & \text{if } g \geq 0 \\ \arctan\left(\dfrac{f}{g}\right) + \pi\ , & \text{if } g < 0 \text{ and } f \geq 0 \\ \arctan\left(\dfrac{f}{g}\right) - \pi\ , & \text{if } g < 0 \text{ and } f < 0 \end{cases} \tag{3.12}$$

The relationship between the wrapped phase map $\phi(i,j)$ and the absolute phase map $\Phi(i,j)$ can be written as

$$\phi(i,j) = \mathrm{mod}\big(\Phi(i,j)\ ,\ 2\pi\big) \tag{3.13}$$

3.3.3.2 Construction of Absolute Phase Map

The functionality of the phase map construction algorithm is to generate an absolute phase map from a set of images which are obtained using phase-shifting method. As mentioned in Related Research (Section 2.3.2.4), the phase map construction

algorithm with the use of multiple fringe frequencies has been used in many latest research work in SMFP technique [3, 19]. The algorithm provides high accuracy in phase measurement, is robust to noises in projections and images, and can handle all sorts of surface discontinuities. In the following context, this algorithm will be referred to as MFF for brevity of interpretation. The phase map construction algorithm we developed is a modified version of MFF algorithm. It uses fewer projection patterns than MFF but achieves the same performance. By using extended mathematical definitions, our algorithm is also able to deal with generalized fringe patterns.

A common scheme of the conventional MFF algorithm uses 4-step phase-shifting and 3 fringe frequencies, meaning a total number of 12 phase-shifted fringe patterns. Our algorithm uses 8 projection patterns only, which can be defined using the following equation:

$$I_n^{(P),k}(\xi,\eta) = \frac{I_{max}^{(P)}}{2}\left[1+\sin\left(c_k\cdot\Phi^{(P)}(\xi,\eta)+\frac{(n-1)\pi}{2}\right)\right] \tag{3.14}$$

where $k = 0, 1, 2$ represent different levels of fringe patterns, $n = 1, \ldots, 4$ represent different phase-shift values, (ξ, η) is the pixel coordinates in the projection pattern, $\Phi^{(P)}(\xi, \eta)$ is the phase function, c_k are scale coefficients, $I_{max}^{(P)}$ is the maximum intensity in the patterns, and $I_n^{(P),k}(\xi, \eta)$ are the intensity values of pixel (ξ, η).

For any phase function $\Phi^{(P)}(\xi, \eta)$, Eqn. 3.14 defines a set of 8 fringe patterns. An example of such fringe pattern set is shown in Fig. 3.6. The 8 patterns are divided into 3 levels by phase sensitivity (corresponding to the scale coefficient c_k). Level-0 has the lowest phase sensitivity (i.e. c_0 is the smallest in $\{c_k\}$, $k = 0, 1, 2$) and contains two patterns with 0 and $\pi/2$ phase shift respectively (corresponding to $n = 1, 2$). Level-2 has the highest phase sensitivity and contains four patterns corresponding to $n = 1$ to 4. Level-1 has the intermediate phase sensitivity and also contains two patterns ($n = 1, 2$). The concept of different "level" of fringe patterns is an analogy to the

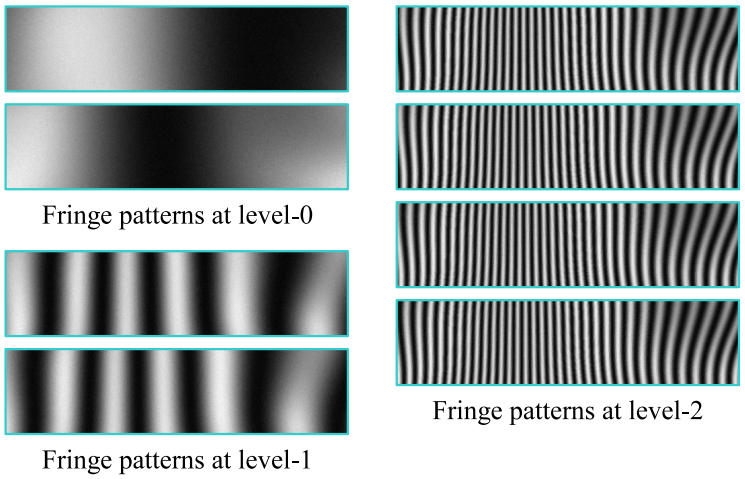


Figure 3.6: A set of 8 (generalized) fringe patterns used in phase map construction

"fringe frequency" in MFF algorithm. Here, the term "level" is introduced to avoid confusion with fringe frequency, since in a generalized fringe pattern the fringe pitch (and hence the fringe frequency) may vary continuously throughout the pattern.

Projecting the 8 fringe patterns as defined above on the object being measured, the corresponding images of the object can be described using the following equation:

$$I_n^{(k)}(i,j) = A(i,j) + B(i,j)\sin\left(\Phi^{(k)}(i,j) + \frac{(n-1)\pi}{2}\right) \tag{3.15}$$

where (i,j) is the pixel coordinates in the image, $k = 0, 1, 2$ represent different levels of fringe patterns and $n = 1, \ldots, 4$ represent different phase-shift values. For each pixel (i,j), $I_n^{(k)}(i,j)$ is the pixel's intensity in the image that corresponds to the projection pattern at level k and n-th phase-shift; $\Phi^{(k)}(i,j)$ is the absolute phase value of the pixel's corresponding point on the object's surface. Assuming that both the camera and the projector have a fairly large depth-of-view and the reflection of the object surface is linear, $A(i,j)$ and $B(i,j)$ are both constants for pixel (i,j) in all the 8 images.

The procedure to compute the absolute phase map of the object surface by using

the 8 images is described below, in which Φ denotes the absolute phase maps and ϕ denotes the wrapped phase maps. A flow chart of the procedure is shown in Fig. 3.7.

- Firstly, the wrapped phase map $\phi^{(2)}$ is computed as follows:

$$\phi^{(2)}(i,j) = \mathbf{arctan}^* \left(\frac{I_1^{(2)}(i,j) - I_3^{(2)}(i,j)}{I_2^{(2)}(i,j) - I_4^{(2)}(i,j)} \right) \tag{3.16}$$

  where the function $\mathbf{arctan}^*(\cdots)$ is defined in Eqn. 3.12.

- Secondly, the coefficients $A(i,j)$ defined in Eqn. 3.15 is calculated using the following equation:

$$A(i,j) = \frac{1}{4} \left[I_1^{(2)}(i,j) + I_2^{(2)}(i,j) + I_3^{(2)}(i,j) + I_4^{(2)}(i,j) \right] \tag{3.17}$$

  Utilizing the $A(i,j)$ obtained, the wrapped phase maps $\phi^{(0)}$ and $\phi^{(1)}$ can be computed as follows:

$$\phi^{(k)}(i,j) = \mathbf{arctan}^* \left(\frac{I_1^{(k)}(i,j) - A(i,j)}{I_2^{(k)}(i,j) - A(i,j)} \right) , \quad k = 0, 1 \tag{3.18}$$

- Finally, the absolute phase maps are calculated from the wrapped phase maps through the following procedure:

  Notice that $\phi^{(0)}$ is an absolute phase map because it is obtained from the level-0 fringe patterns, which are designed to have a low phase sensitivity, e.g. with a phase range of $[-\pi, \pi]$, to avoid phase wrapping. Therefore, we have $\Phi^{(0)} = \phi^{(0)}$. From the relationship between the fringe patterns at different levels (see Eqn. 3.14), we can get that for all pixel (i,j) the following equations hold:

$$\begin{aligned} \Phi^{(1)}(i,j) &= (c_1/c_0) \cdot \Phi^{(0)}(i,j) \quad \text{and} \\ \Phi^{(2)}(i,j) &= (c_2/c_1) \cdot \Phi^{(1)}(i,j) \end{aligned} \tag{3.19}$$

where c_0, c_1 and c_2 are coefficients defined in Eqn. 3.14. It can also be known from the relationship between an absolute phase map and its wrapped counterpart that

$$\Phi^{(k)}(i,j) = \phi^{(k)}(i,j) + 2\pi \cdot n_k(i,j) \ , \quad k = 1,2 \tag{3.20}$$

where $n_k(i,j)$ are unknown integers. By combining Eqn. 3.19 and 3.20, the absolute phase map $\Phi^{(1)}(i,j)$ and $\Phi^{(2)}(i,j)$ can be solved. $\Phi^{(2)}(i,j)$ has the highest phase sensitivity and is the final result of the phase map construction.

The idea described above for reducing the number of projection patterns required in phase map construction can be applied to any phase-shifting strategy that contains redundant image intensity information. For the generic N-step phase-shifting algorithm with uniform phase shifts, whose images can be described as

$$I_n(i,j) = A(i,j) + B(i,j)\sin\left(\Phi(i,j) + \frac{2\pi(n-1)}{N}\right) \ , \quad n = 1,\ldots,N \tag{3.21}$$

the corresponding equation for computing $A(i,j)$ is as follows:

$$A(i,j) = \frac{1}{N}\sum_{n=1}^{N} I_n(i,j) \tag{3.22}$$

Notice that although assumed to be a constant for all 8 images, $B(i,j)$ (as defined in Eqn. 3.15) is not utilized to further reduce the number of patterns required in phase map construction. This is because, in practice the value of $B(i,j)$ may vary slightly for different level of fringe patterns due to the out-of-focus projection on the object's surface. However, this does not affect the value of $A(i,j)$ at all.

An example of phase map construction by using the proposed algorithm with 8 projection patterns is shown in Fig. 3.8.

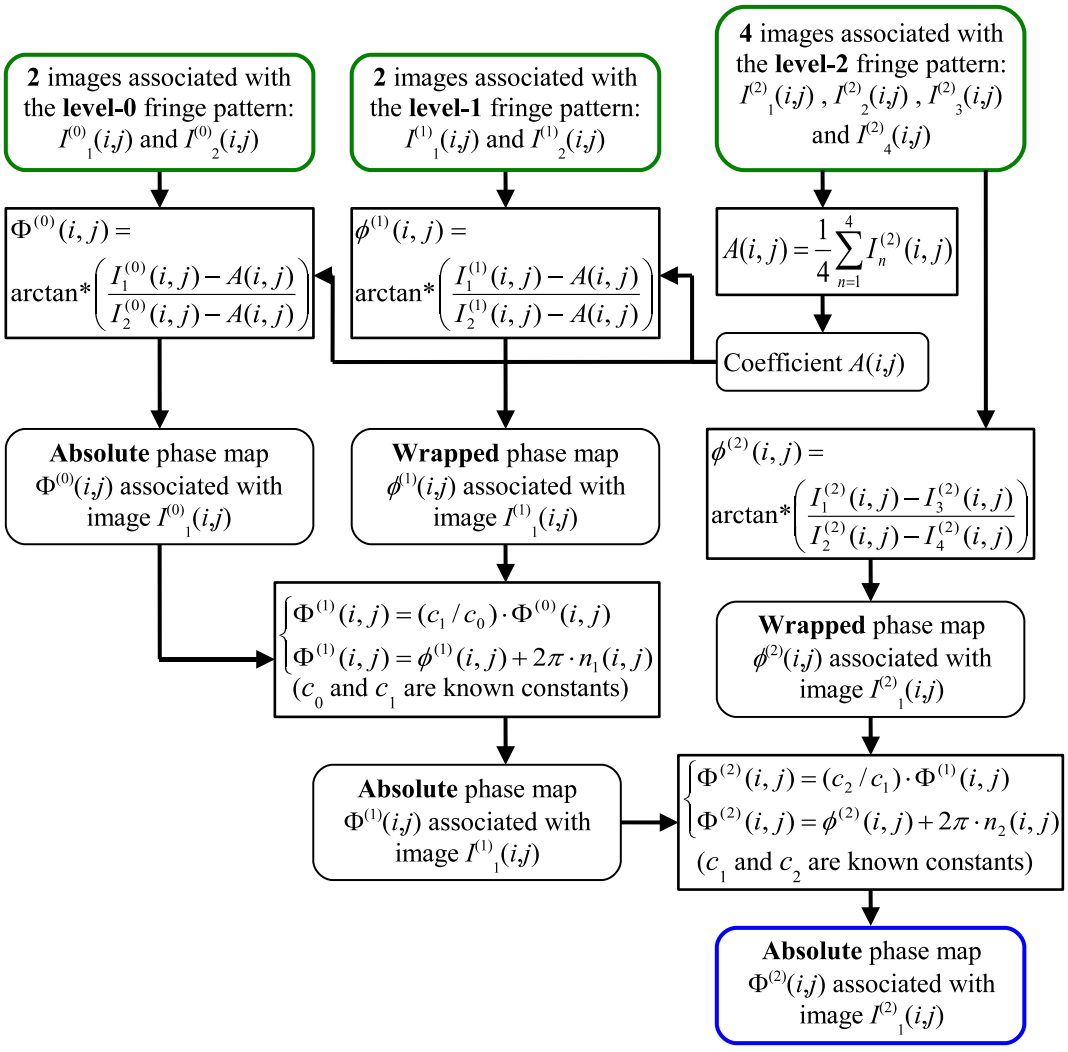
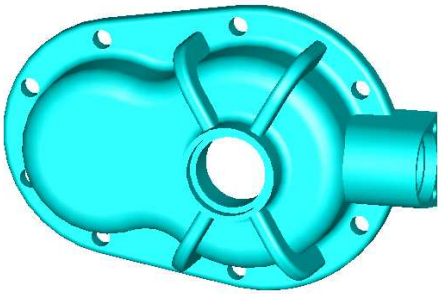
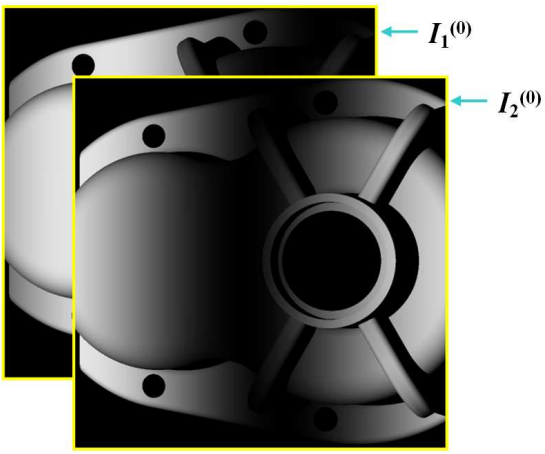


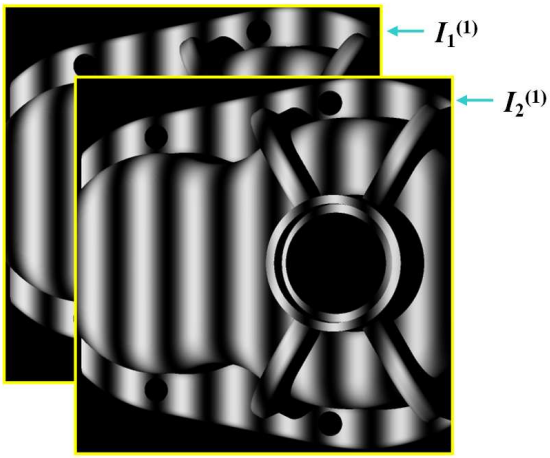
Figure 3.7: Flow chart of the proposed algorithm for absolute phase map construction by using 8 fringe patterns



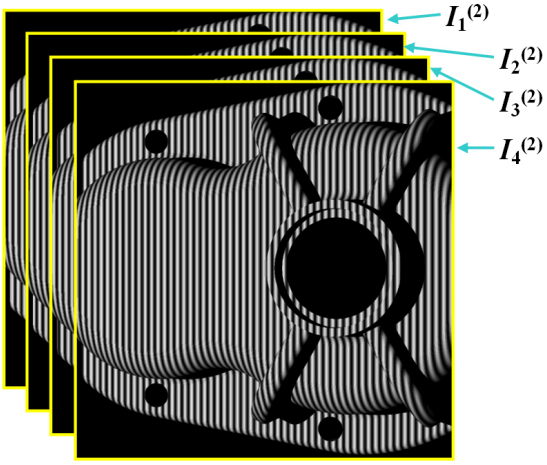
(a) The part being measured



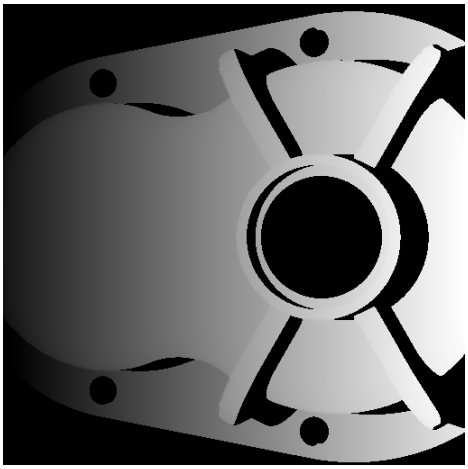
(b) The two images from level 0: $I_1^{(0)}$ and $I_2^{(0)}$



(c) The two images from level 1: $I_1^{(1)}$ and $I_2^{(1)}$



(d) The four images from level 2: $I_1^{(2)}$, $I_2^{(2)}$, $I_3^{(2)}$ and $I_4^{(2)}$



(e) The absolute phase map of the part

Figure 3.8: An example of phase map construction by using the proposed algorithm

3.3.4 Algorithm for Construction of Point Cloud

3.3.4.1 Description of the Algorithm

The point cloud construction algorithm converts the absolute phase map of an object to a dense 3-D point cloud that represents the object's surface. The algorithm requires the following data to be known a priori:

- All intrinsic parameters of the camera, i.e. the intrinsic matrix $\boldsymbol{A}$ and the lens distortion coefficients k_s $(s = 1, \ldots, 5)$. Definitions of the parameters can be found in Section 3.2.1.
- The position of the projector's projection center, $P(P_x, P_y, P_z)$, with respect to the camera coordinate frame. (All spatial coordinates referred in this section are w.r.t. the camera's coordinate frame if not otherwise declared)
- The absolute phase map of a reference plane, denoted by $\Phi_R(u, v)$, and the position and orientation of the reference plane, which is represented by a point on the plane, O_R, and the plane's normal vector $\boldsymbol{n}_R$.

Let $\Phi(u, v)$ denote the absolute phase map of the object being measured. For each pixel (u, v) with a valid absolute phase value, a 3-D point can be possibly generated. The algorithm for computing the 3-D coordinates of point M, corresponding to pixel (u, v), is described at below. Figure 3.9 shows an illustration of the geometric relationships involved in the computation and Fig. 3.11 is a flow chart of the algorithm. Detailed proofs of the equations are presented in Appendix A.2.

Let point M_I denote the idealized position of the center of pixel (u, v) on the image plane of the camera. It can be known from the pinhole camera model that point M lies on line $\overline{CM_I}$. Let point M_R denote the intersection of line $\overline{CM_I}$ with the reference plane. Similarly, let point N_R denote the intersection of line $\overline{PM}$ with the reference plane and let N_I denote the intersection of line $\overline{CN_R}$ with the image plane

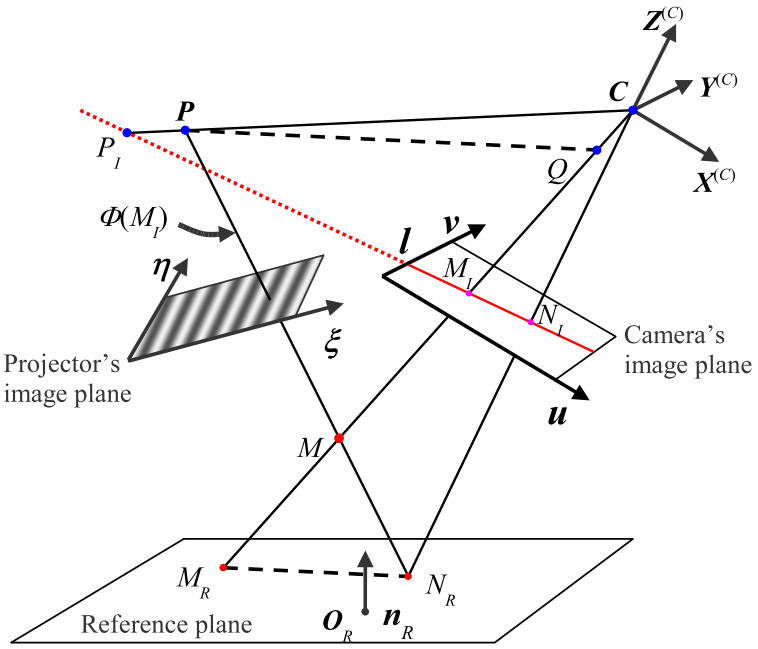


Figure 3.9: Computing the 3-D coordinates of a point M from its absolute phase value

of the camera. Define point Q as the intersection of line $\overline{CM_I}$ with the plane that is parallel to the reference plane while passes through point P. It can be seen from the definitions that, points C, P, Q, M_I, N_I, M_R, N_R and M are coplanar.

The procedure to compute the 3-D position of point M is as follows:

1. As a first step, locate the pixel coordinates of point N_I by searching in the reference phase map $\Phi_R(u, v)$ along line $\overline{P_I M_I}$ for a point with phase value $\Phi(M)$ (see Fig. 3.10), in which P_I is the projection of point P on the camera's image plane $\boldsymbol{u}$–$\boldsymbol{v}$. Due to the restrictions imposed on the projection patterns used in phase-shifting (see Section 3.3.3.1), the distribution of phase values along line $\overline{P_I M_I}$ is guaranteed to be monotonic. Hence, the location of point N_I is unique if one exists.

2. The 3-D coordinates of points M_I and N_I are calculated from their pixel coordinates, by using the pinhole camera model and the camera's intrinsic parameters

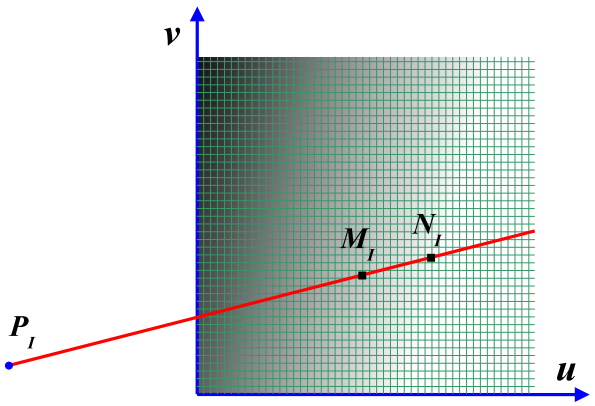


Figure 3.10: Finding the pixel coordinates of point N_I in the reference phase map Φ_R

(see Section 3.2.2). Accordingly, the 3-D coordinates of points M_R and N_R can be calculated from line-plane-intersections by using the following equations:

$$M_R = C + \left(\frac{\overrightarrow{CO_R} \cdot \boldsymbol{n}_R}{\overrightarrow{CM_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CM_I} \,, \quad N_R = C + \left(\frac{\overrightarrow{CO_R} \cdot \boldsymbol{n}_R}{\overrightarrow{CN_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CN_I} \tag{3.23}$$

where O_R and $\boldsymbol{n}_R$ represent the position and orientation of the reference plane. Similarly, the 3-D coordinates of point Q can be calculated from the following equation:

$$Q = C + \left(\frac{\overrightarrow{CP} \cdot \boldsymbol{n}_R}{\overrightarrow{CM_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CM_I} \tag{3.24}$$

3. Finally, the 3-D position of point M can be calculated as follows:

$$M = Q + \left(\frac{\left|\overrightarrow{PQ}\right| \left|\overrightarrow{QM_R}\right|}{\left|\overrightarrow{PQ} + \overrightarrow{M_R N_R}\right| \left|\overrightarrow{CM_I}\right|} \right) \overrightarrow{CM_I} \tag{3.25}$$

3.3.4.2 Comparison with Existing Algorithms

The proposed algorithm for point cloud construction is a hybrid of the geometric approach (Section 2.1.1) and the calibration matrix based approach (Section 2.1.2).

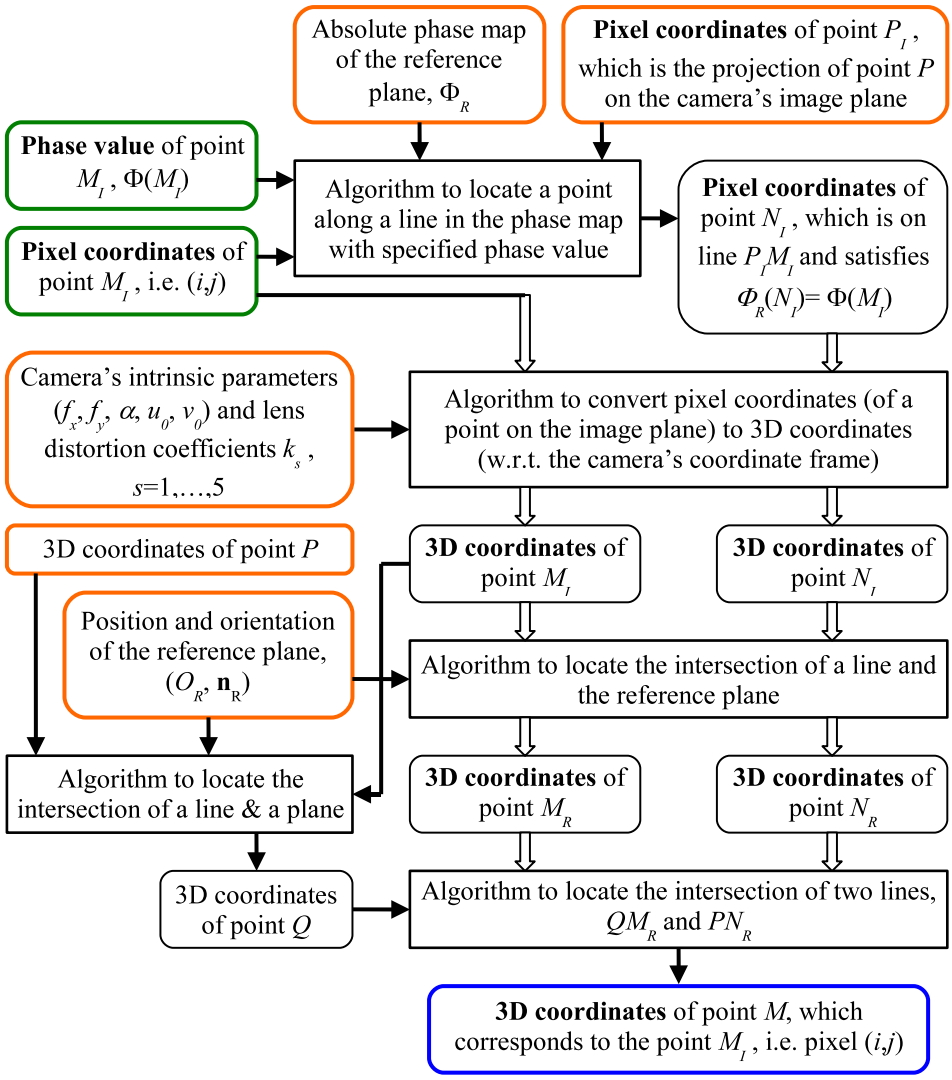
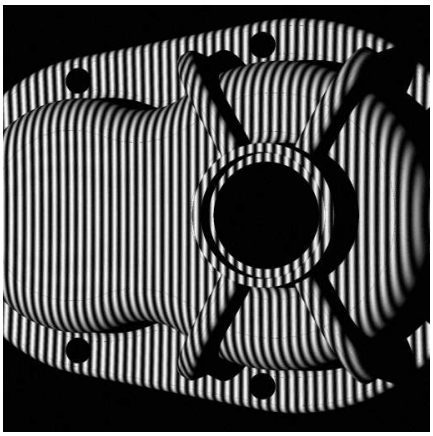


Figure 3.11: Flow chart of the proposed algorithm for point cloud construction

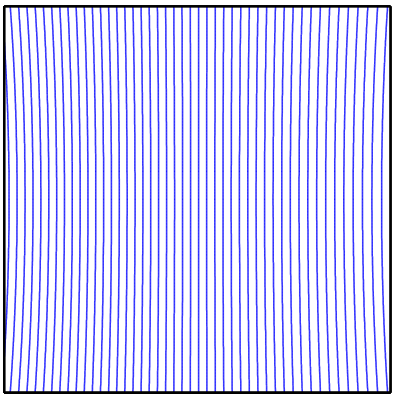
It requires full knowledge of the camera parameters, as well as the geometric information of the reference plane. For the projector parameters, only the position of the projection center is needed explicitly. The rest of the geometric information of the projector is given by the reference phase map in an implicit way. Compared to the pure geometric approach [19], this hybrid approach requires only about half of the parameters. As a result, the parameter estimation algorithm becomes simpler and numerically more stable, since there are fewer variables in the nonlinear minimization process. Although a reference phase map and the geometric information of the reference plane are required in addition, they are fairly easy to acquire. The hybrid approach does not lose any generality or accuracy by ignoring the majority of projector parameters, as all projector parameters are reflected in the algorithm, explicitly or implicitly. Compared to the calibration matrix based approach, this hybrid approach is more accurate since it utilizes an accurate camera model instead of approximate polynomial interpolations for the construction of point clouds. Moreover, it requires much less memory for the computation and the calibration process is made easier and more flexible.

With the use of a reference phase map, the proposed algorithm handles the lens distortions of the projector automatically. This feature can be demonstrated by the example shown in Fig. 3.12. A part was measured by using a projector with lens distortion and the results obtained by using the proposed algorithm and an algorithm that does not consider projector distortions were compared. The part used in the example has a dimension of approximately $180mm$(W) $\times 120mm$(H) $\times 55mm$(D) and a picture of the part used in the example can be found in Fig. 3.8(a). Figure 3.12(a) shows an image of the part under a sinusoidal fringe pattern that is projected using a projector with radial lens distortion (radial distortion coefficient $k_1 = -2.0$). As a comparison, an image of the part under the same fringe pattern but using a projector without distortion is shown in Fig. 3.12(c). The difference between the two images

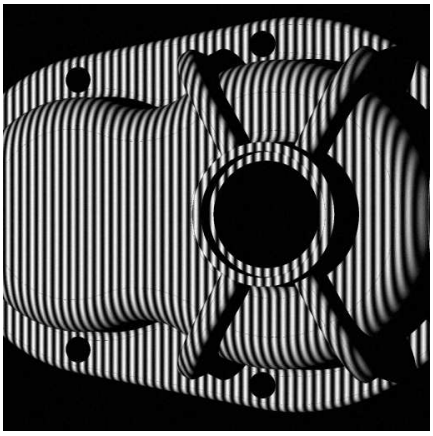
can hardly be found by visual examination except at the edge areas. However, the influence of the projector's radial lens distortion can be seen more easily from the reference phase maps acquired under different conditions. Figure 3.12(b) and 3.12(d) show the contours of the phase maps, from which it can be seen that the phase contours are distorted at edge areas of the phase map that was acquired with projector distortion. When the projector distortion is not taken into account, the constructed point cloud contains significant error, especially at the edge areas where the distortion of projection is large (see Fig. 3.12(e)). However, the influence of the distortion can be compensated automatically by using the proposed algorithm (see Fig. 3.12(f)).



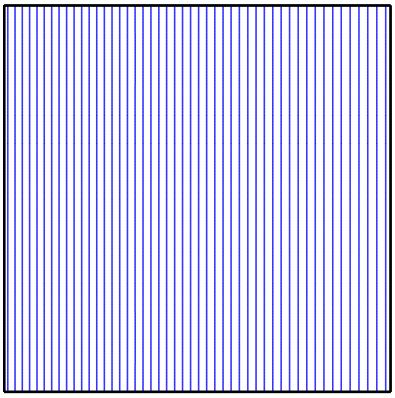
(a) Image of the part under fringe projection **with** projector distortion



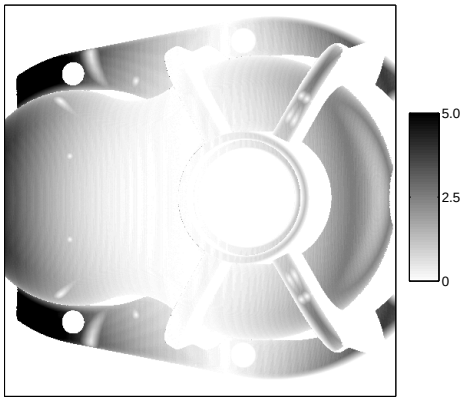
(b) Contour display of the reference phase map **with** projector distortion



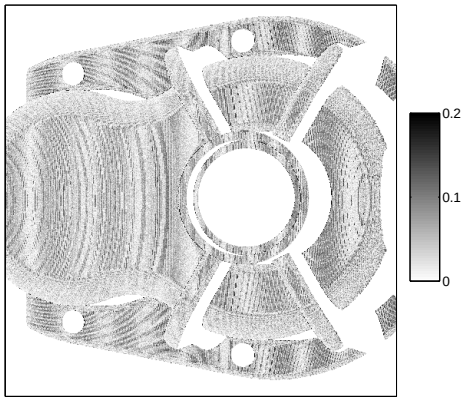
(c) Image of the part under fringe projection with **no** projector distortion



(d) Contour display of the reference phase map with **no** projector distortion



(e) Error distribution of the point cloud constructed using an algorithm that does not consider projector distortion



(f) Error distribution of the point cloud constructed using the proposed algorithm

Figure 3.12: Automatic compensation of projector distortion by using the proposed algorithm for point cloud construction

3.3.5 Algorithms for Estimation of Sensor Parameters

3.3.5.1 Overview of Algorithms

The parameter estimation algorithm can be divided into two parts, the estimation of camera parameters and the estimation of the projector's projection center.

The estimation of camera parameters, i.e. camera calibration method, has been intensively studied since late 80's [53] and many improvements have been made in recent years [27, 26]. The algorithms developed have been widely used by research and industrial applications that require high-accuracy camera models, e.g. photogrammetry. We used the camera calibration algorithm and software library that were developed by the computer vision research group at the California Institute of Technology [56].

A projector is very similar to a camera in terms of optical geometry and can be described accurately by a pinhole camera model with modeling of lens distortion (see Section 3.3.2). The parameter estimation algorithm for projector can hence be developed by adapting the existing camera calibration methods. However, such algorithms have rarely been discussed in the research of SMDFP technique until recently in a paper by Legarda-Sáenz et al. [19].

For the point cloud construction algorithm proposed in Section 3.3.4, the only required knowledge on projector parameters is the position of the projector's projection center, i.e. point P as shown in Fig. 3.9. Tailored for this specific requirement, an algorithm was developed which estimates the projector's projection center only. This algorithm requires all camera parameters to be known, therefore a camera calibration needs to be performed beforehand.

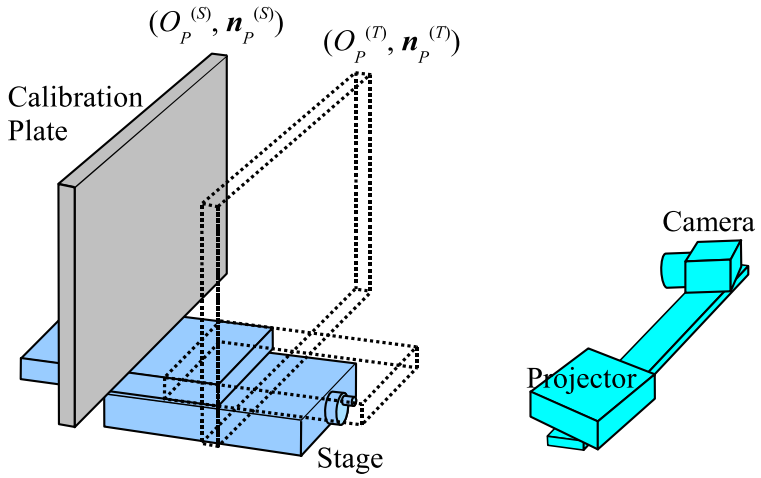


Figure 3.13: System setup for estimation of the projector's projection center

3.3.5.2 Estimation of the Projector's Projection Center

The procedure for estimating the projector's projection center involves measuring a flat plane at two different positions. A detailed description of procedure is given below. The system setup for the calibration is illustrated in Fig. 3.13.

- At the first step, a flat calibration plate is placed at the farther side of the measurement volume. The position and orientation of the plate, denoted by $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$, can be estimated by using the calibrated camera. Two absolute phase maps of the calibration plate, namely $\Phi_V^{(S)}(u,v)$ and $\Phi_H^{(S)}(u,v)$, are obtained by using vertical and horizontal fringe patterns respectively (see Fig. 3.14). For each pixel (i,j) in the camera's image plane, its corresponding point on the plane $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$, denoted by $S_{i,j}$, can be located by using the camera model and camera parameters. With the acquired absolute phase maps $\Phi_V^{(S)}$ and $\Phi_H^{(S)}$, each point $S_{i,j}$ is associated with two phase values, $\Phi_V^{(S)}(i,j)$ and $\Phi_H^{(S)}(i,j)$. This phase value pair $\left(\Phi_V^{(S)}(i,j), \Phi_H^{(S)}(i,j)\right)$ determines an unique point on the image plane of the projector.

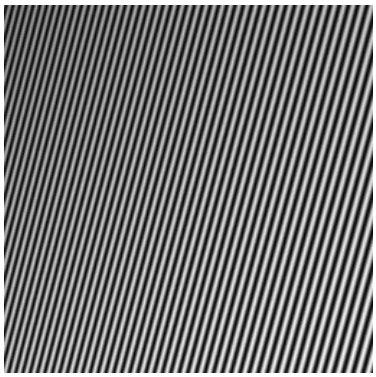
- At the second step, the calibration plate is moved to $\left(O_P^{(T)}, \boldsymbol{n}_P^{(T)}\right)$, which is at the nearer side of the measurement volume. Although in practice the calibration plate is moved by using a stage and hence the positions of the plate are (nearly) parallel, it is not required by the algorithm. The precise values of $O_P^{(T)}$ and $\boldsymbol{n}_P^{(T)}$ can be obtained either by using the calibrated camera or from the movement of the plate which can be measured by other means, e.g. CMM. Again, two absolute phase maps of the calibration plate are acquired by using vertical and horizontal fringe patterns. The two phase maps are denoted by $\Phi_V^{(T)}(u, v)$ and $\Phi_H^{(T)}(u, v)$ respectively.

Utilizing the acquired phase maps of the calibration plane (i.e. $\Phi_V^{(S)}$, $\Phi_H^{(S)}$, $\Phi_V^{(T)}$ and $\Phi_H^{(T)}$), as well as the positions and orientations of the plane where the phase maps are obtained (i.e. $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$ and $\left(O_P^{(T)}, \boldsymbol{n}_P^{(T)}\right)$), the projection center of the projector can be estimated by using the following algorithm:

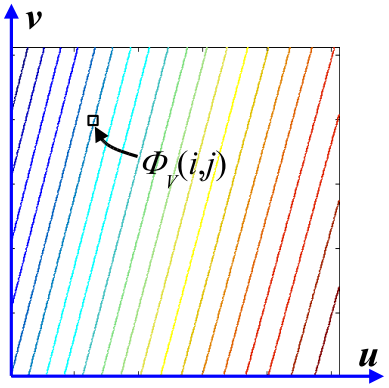
- For each point $S_{i,j}$ on plane $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$, whose phase value pair are $\left(\Phi_V^{(S)}, \Phi_H^{(S)}\right)$, a point on plane $\left(O_P^{(T)}, \boldsymbol{n}_P^{(T)}\right)$ can be located, which has the same phase value pair as $S_{i,j}$. Let $T_{i,j}$ denote this point and let $T_{i,j}^{(I)}$ denote its projection on the camera's image plane $\boldsymbol{u}$-$\boldsymbol{v}$. Notice that the subscripts i and j of $T_{i,j}$ are for reference to point $S_{i,j}$. They are not the corresponding pixel coordinates of point $T_{i,j}$. The location of point $T_{i,j}^{(I)}$ in the image plane $\boldsymbol{u}$–$\boldsymbol{v}$ is determined by finding the intersection of two phase contour lines

$$\begin{cases} \Phi_V^{(T)} = \Phi_V^{(S)}(S_{i,j}) \\ \Phi_H^{(T)} = \Phi_H^{(S)}(S_{i,j}) \end{cases} \tag{3.26}$$

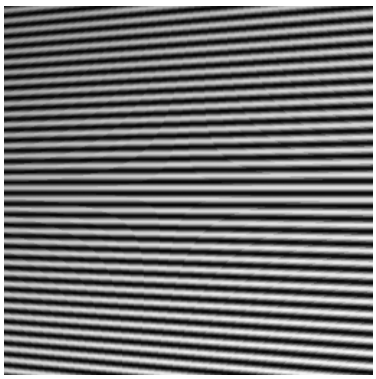
  in the $\boldsymbol{u}$–$\boldsymbol{v}$ space. The algorithm for doing this will be explained later. Once $T_{i,j}^{(I)}$ is located, the position of point $T_{i,j}$ on plane $\left(O_P^{(T)}, \boldsymbol{n}_P^{(T)}\right)$ can be calculated from the camera model and camera parameters.



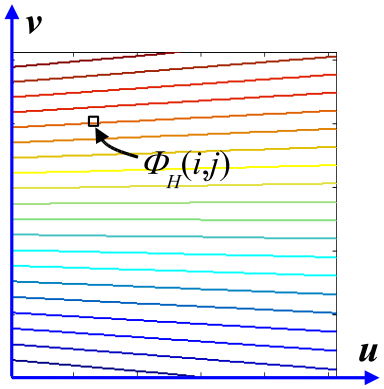
(a) One of the images obtained for the acquisition of Φ_V by using vertical fringe patterns



(b) Contour display of the acquired phase map Φ_V



(c) One of the images obtained for the acquisition of Φ_H by using horizontal fringe patterns



(d) Contour display of the acquired phase map Φ_H

Figure 3.14: Acquisition of two absolute phase maps of the calibration plate, Φ_V and Φ_H

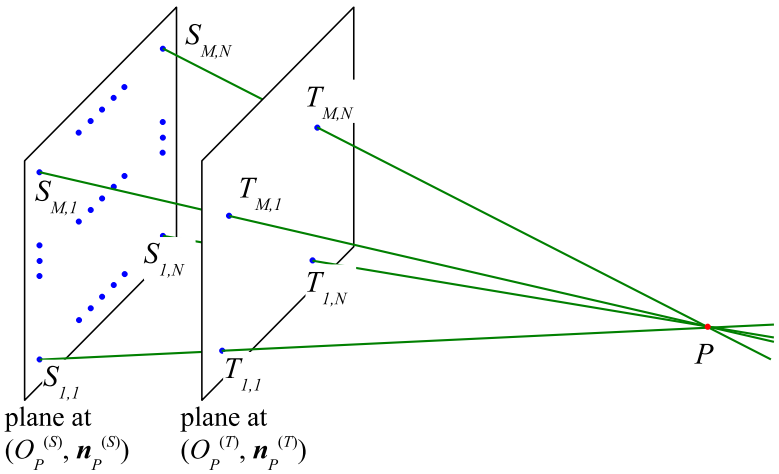


Figure 3.15: Estimation of the projector's projection center

- Since point $S_{i,j}$ and point $T_{i,j}$ share the same phase value pair, line $\overline{S_{i,j}T_{i,j}}$ must pass through the projection center of the projector P, according to the pinhole camera model. Therefore, the position of P can be located from the intersection of all such lines that connect a point on plane $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$ with its corresponding point on plane $\left(O_P^{(T)}, \boldsymbol{n}_P^{(T)}\right)$. The points on plane $\left(O_P^{(S)}, \boldsymbol{n}_P^{(S)}\right)$ that do not have corresponding points due to the shifted phase range at different plane positions, are skipped. Because the intersection of multiple 3-D lines is an over-constrained problem, the position of point P is computed using a least-square procedure.

The algorithm for locating point $T_{i,j}^{(I)}$ on the image plane $\boldsymbol{u}$–$\boldsymbol{v}$ by using given phase value pair $\left(\Phi_V^{(S)}(S_{i,j}), \Phi_H^{(S)}(S_{i,j})\right)$ is as follows:

1. Starting from an initial position $T_0(u_0, v_0)$ (the superscript I and subscripts i and j of $T_{i,j}^{(I)}$ are dropped for brevity), search in the phase map $\Phi_H^{(T)}(u, v)$ along $\boldsymbol{v}$-direction for a point $T_1(u_1, v_1)$ such that

$$\begin{cases} u_1 = u_0 \\ \Phi_H^{(T)}(u_1, v_1) = \Phi_H^{(S)}(S_{i,j}) \end{cases} \tag{3.27}$$

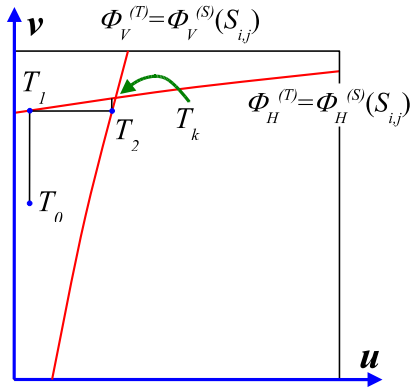


Figure 3.16: Finding the intersection of two phase contour lines

The selection of the initial point $T_0(u_0, v_0)$ can be arbitrary. The location of v_1 is calculated through interpolation in the phase map, therefore sub-pixel accuracy can be achieved (see Fig. 3.16).

2. search in the phase map $\Phi_V^{(T)}(u, v)$ along $\boldsymbol{u}$-direction for a point $T_2(u_2, v_2)$ such that

$$\begin{cases} v_2 = v_1 \\ \Phi_V^{(T)}(u_2, v_2) = \Phi_V^{(S)}(S_{i,j}) \end{cases} \tag{3.28}$$

3. Repeat the zigzag search path until: (1) $|u_k - u_{k-1}|$ and $|v_k - v_{k-1}|$ are both within the designated tolerance, which means the point $T_{i,j}^{(I)}$ has been found; Or (2) u_k or v_k is out of the boundaries of the $\boldsymbol{u}$-$\boldsymbol{v}$ space, which means a bad starting point or the desired point does not exist.

For most practical SMFP system setups, the phase contours in phase map Φ_V are nearly vertical and the phase contours in Φ_H are nearly horizontal. Therefore, the locating of point $T_{i,j}^{(I)}$ by using the above algorithm converges fast. Although the initial point $T_0(u_0, v_0)$ can be selected arbitrarily, a good initial guess can further speed up the calculation and avoid missing intersections. An excellent candidate for the initial point is the calculated approximate intersection by considering the phase

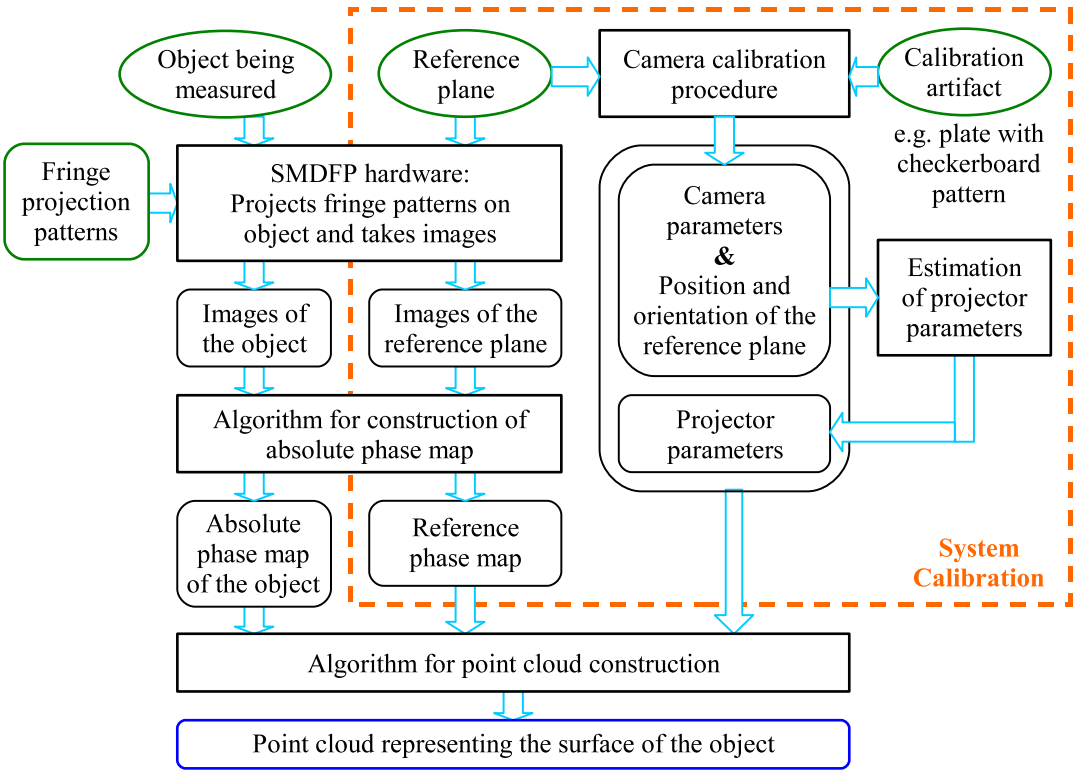


Figure 3.17: Relationship of the proposed algorithms from a system perspective

contours as straight lines in the $\boldsymbol{u}$-$\boldsymbol{v}$ space.

3.3.6 Relationship Between Algorithms

The mathematical model and related algorithms described in the previous sections (Section 3.3.2 to 3.3.5) form a complete algorithmic infrastructure for SMDFP systems. From a system perspective, the functionalities of the algorithms and the data flows between them can be illustrated by Fig. 3.17.

A SMDFP system needs to be calibrated before performing actual measurements. At the first step of the calibration, the camera parameters are estimated by using an artifact with accurately known geometry or feature points. In our experiments, we used a flat plate with a checkerboard pattern printed on top. After the camera is

calibrated, it can be used to measure the position and orientation of the calibration plate by means of photogrammetry. The second step of the calibration is the estimation of projector parameters (i.e. the position of the projector's projection center), in which the camera parameters acquired in the first step are utilized. After this step, all sensor parameters required are known. The last step of the calibration is to acquire a reference phase map, which is accomplished by placing a flat plate at around the center of the system's measurement volume and constructing an absolute phase map of the plate.

After a SMDFP system is calibrated, it is ready for shape measurements. First, a set of predefined fringe patterns are projected on the object and the corresponding images are recorded. The fringe patterns used in the measurement are the same as used in the acquisition of the reference phase map. The recorded images are processed and an absolute phase map of the object is obtained. By using this phase map, together with the sensor parameters and the reference phase map that were acquired during calibration, a 3-D point cloud can be constructed which represents the surface of the object been measured.

3.4 Implementation and Results

3.4.1 Software and Hardware

The proposed mathematical model and algorithms were programmed in C++. A simple SMDFP hardware (i.e. the sensor) was built for carrying out shape measurement experiments. The hardware consists of a computer projector (BenQ PB2220 DMD projector, 1024×768 pixels, 1700 ANSI lumen), a programmable B/W digital camera (Dragonfly^{TM} made by Point Grey Research, 1/3" Sony CCD, 640×480 pixels) and a few accessories. The projector and the camera are both fixed to an aluminum frame

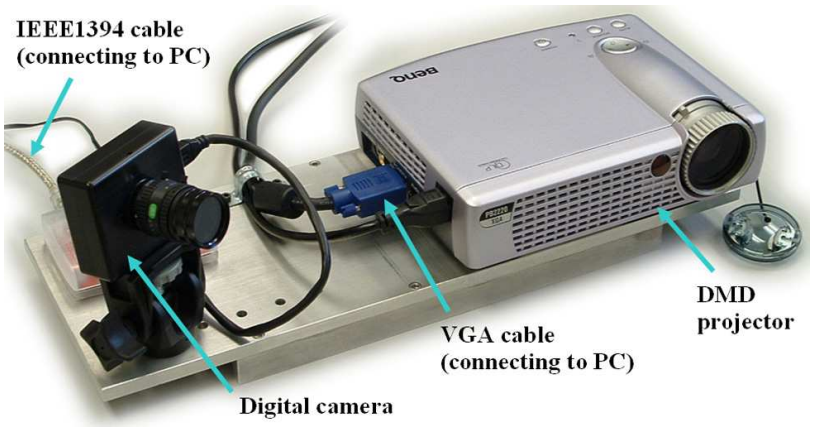


Figure 3.18: Photograph of the hardware made for experiments

and connected to a computer. A photograph of the hardware is shown in Fig. 3.18. In a shape measurement cycle, the computer sends projection patterns (in the form of grayscale bitmaps) to the projector through VGA interface, and the camera sends back the images through an IEEE-1394 (a.k.a. FireWire) cable.

A computer program was developed for hardware control. A few engineering issues were addressed, such as the calibration of the projection intensity and the coordination between the projector and the camera. In the following paragraphs, these two issues will be explained in detail.

Commercial computer projectors are designed for the visual reception of human eyes. The light intensity generated by the computer projector is usually not linear to the amplitude of the input electric signal, which is made so on purpose. However, this nonlinear characteristic of the projection intensity, usually referred as "gamma effect", is not desired in SMDFP technique. Hence, the projection intensity needs to be compensated. In our experiments, this is done by first finding the response curve of the projection intensity and then adjusting the intensity values in the projection patterns based on the reciprocal of the response curve.

The coordination between the projector and the camera means more than com-

manding the camera to capture an image while letting the projector generate a designated pattern. The exposure time of the camera needs to be compatible with the refreshing rate of the projector. As mentioned in Section 1.1.4, DMD-based computer projectors generate grayscale intensities by a sequence flickering of micro-mirrors. As a result, the projector needs a fixed period of time to finish generating a designated grayscale pattern, and this generation cycle repeats at the refreshing rate of the projector (usually from 60Hz to 100Hz). From the camera's perspective, the light pattern produced by the projector flickers at a frequency of a few to tens of kHz, although it may appear stable to human eyes. Therefore, in order to capture the correct projection intensities, the camera's exposure time needs to be set as multiples of the projector's refreshing cycle.

3.4.2 Measurement of Complex Shapes

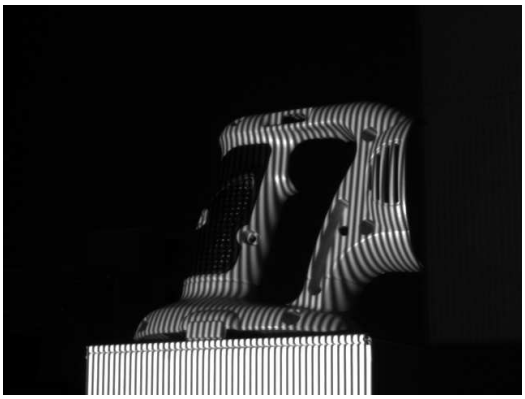
Using the software and hardware developed, a large number of measurements were made on over 20 parts with varied shapes. The system worked very well for all the parts except having difficulty in dealing with metallic surfaces, which is a common problem for most shape measurement techniques based on structured light.

A few examples of the measurements are shown in Figs. 3.19 to 3.22. All measurements were conducted under regular indoor lighting condition. No surface treatment was applied. The constructed point clouds are presented without any post-processing.

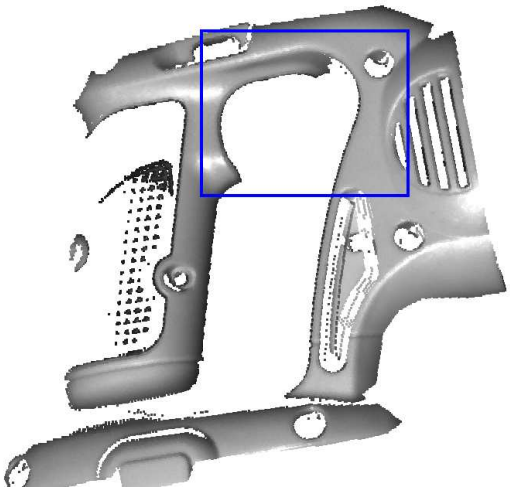
As can be seen from the results, the developed system is able to handle surface discontinuities (e.g. holes, slots and shadowed areas) and some degree of specular reflection (i.e. the typical specular reflection from a plastic surface). It can also be noticed that, surfaces that are not shone by the projector or invisible to the camera cannot be measured. For example, the three side faces of a key in the keyboard, as shown in Fig. 3.20, are left out. This is a limitation of all shape measurement

techniques based on triangulation, but can be solved by taking multiple measurements at different angles and merging the point clouds acquired. However, the registration and merging of point clouds [68] is not one of the research issues discussed in this dissertation, hence this problem was not addressed.

(a) Photograph of the drill housing



(b) Image of the drill housing under the level-2 fringe pattern with 0 phase-shift

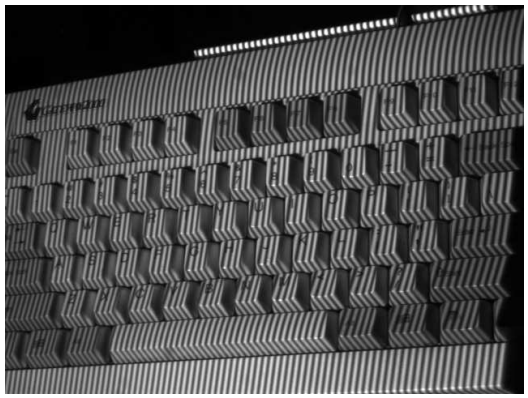


(c) Rendering of the point cloud constructed

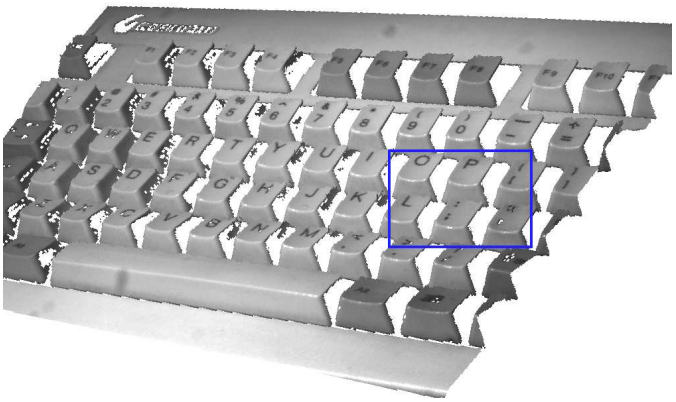
(d) Enlarged view of a mesh rendering of the point cloud (in the blue window shown in (c))

Figure 3.19: Measurement of a plastic drill housing

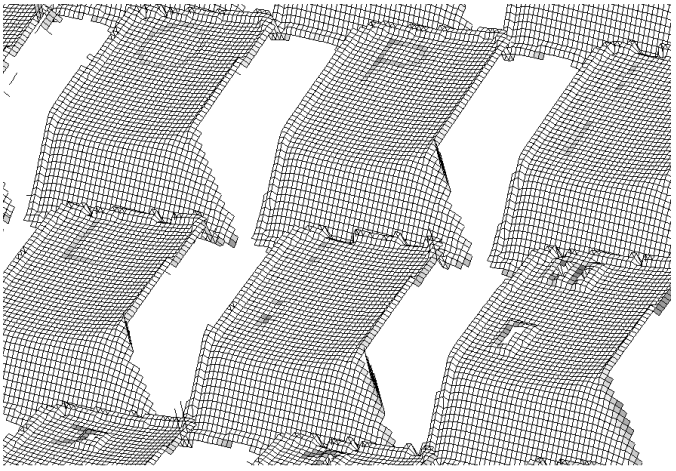
(a) Photograph of the computer keyboard



(b) Image of the keyboard under the level-2 fringe pattern with 0 phase-shift



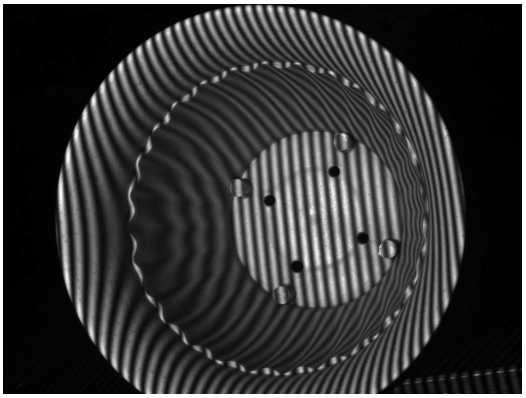
(c) Rendering (with texture mapping) of the point cloud constructed



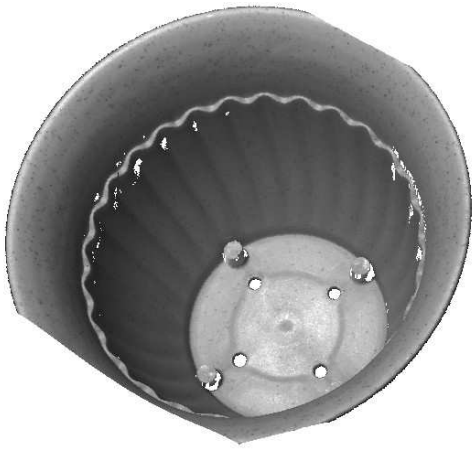
(d) Enlarged view of a mesh rendering of the point cloud (in the blue window shown in (c))

Figure 3.20: Measurement of a computer keyboard

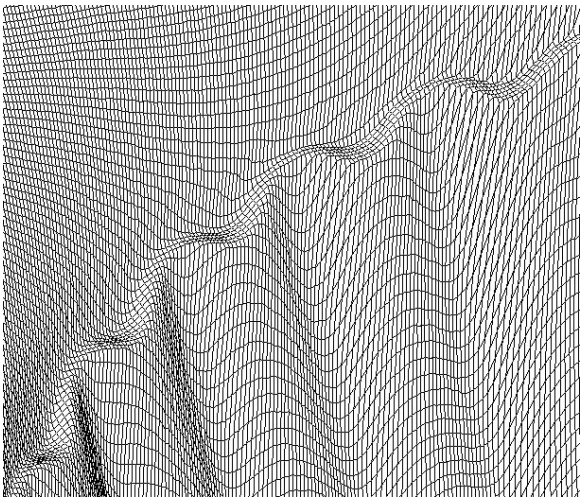
(a) Photograph of the flowerpot



(b) Image of the flowerpot under the level-2 fringe pattern with 0 phase-shift



(c) Rendering (with texture mapping) of the point cloud constructed



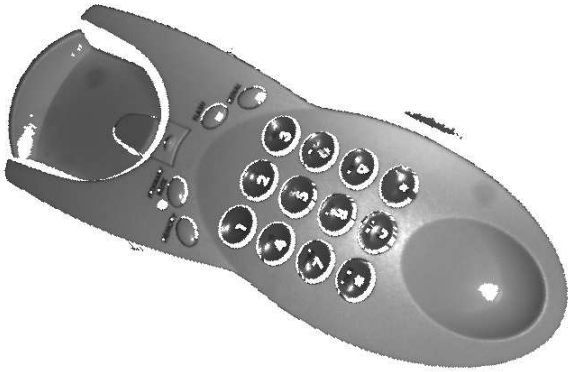
(d) Enlarged view of a mesh rendering of the point cloud

Figure 3.21: Measurement of a plastic flowerpot

(a) Photograph of the front panel of a desktop computer

(b) Rendering (with texture mapping) of the point cloud constructed from measurement of the computer panel

(c) Photograph of the telephone been measured



(d) Rendering (with texture mapping) of the point cloud constructed from measurement of the telephone

(e) Photograph of the shoe been measured

(f) Rendering (with texture mapping) of the point cloud constructed from measurement of the shoe

Figure 3.22: Measurements of varied artifacts

3.4.3 Evaluation of Measurement Accuracy

3.4.3.1 Measurement of Gauge Part No. 1

In order to evaluate the measurement accuracy achieved by using the developed software and hardware, two gauge parts with accurately known geometry were measured.

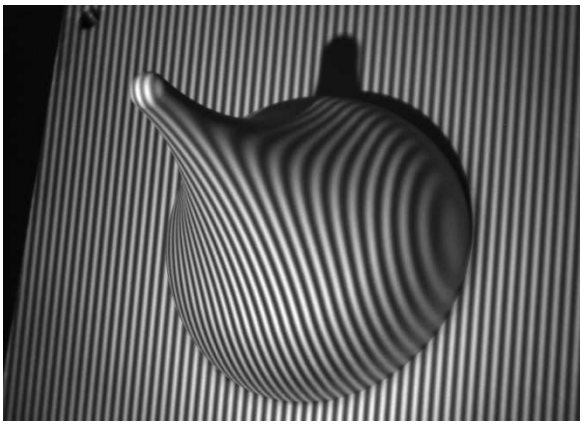
The first gauge part has a curved cone shape with a dimension of $128mm$ (L) $\times$ $128mm$ (W) $\times$ $151mm$ (H). Figure 3.23(a) shows a picture of the part. The part was made of aluminum alloy and machined very accurately to a predefined CAD model. According to a CMM measurement of the part, the maximum deviation of the part from the CAD model is $35\mu m$ and the standard deviation is $11\mu m$. Before measurement, the part was painted using a spray paint (Spot-Check SKD-S2 developer) to avoid the specular reflection from the metallic surface of the part under light projection. The paint gives the part a diffuse and reasonably uniform coating, which makes the measurement much easier. The thickness of the coating is estimated to be around $100\mu m$. A picture of the part after painting is shown in Fig. 3.23(b).

The accuracy of the point cloud obtained from measurement is evaluated by comparing it against the CAD model of the part. In order to do so, the point cloud is first transformed to the coordinate frame of the CAD model, since the point cloud is constructed in the camera's coordinate frame. The coordinate transformation is established by finding the best fit of the point cloud to the CAD model. In our experiments, this task was accomplished by using a commercial reverse engineering software [69]. Figure 3.24 shows the point cloud and the CAD model of the part after the coordinate transformation. Since the raw point cloud obtained from measurement contains too many points (about 200,000 points) for the reverse engineering software to process, only an uniformly selected subset of the points are included in data processing.

After the coordinate transformation, the divergence between the point cloud and

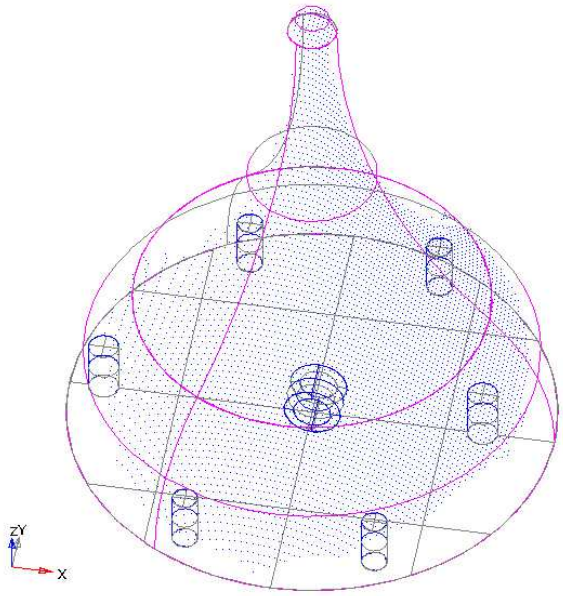
(a) Photograph of the part

(b) Photograph of the part with paint applied

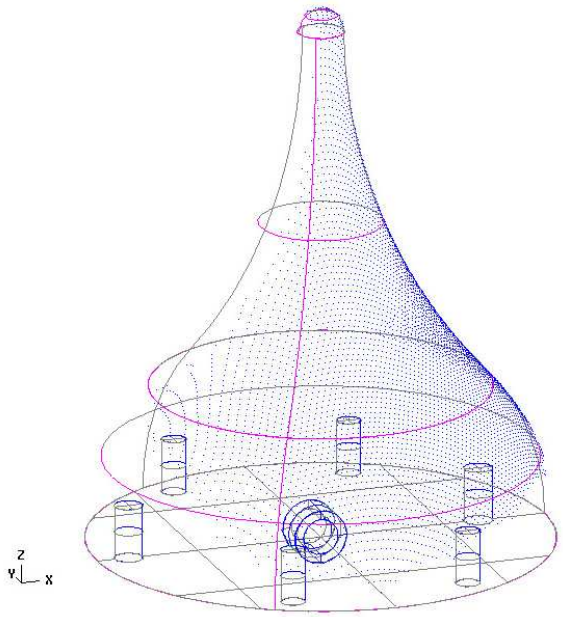


(c) Image of the part under fringe projection

Figure 3.23: Measurement of gauge part No. 1



(a) View from angle A



(b) View from angle B

Figure 3.24: The acquired point cloud after being transformed to the CAD model's coordinate frame

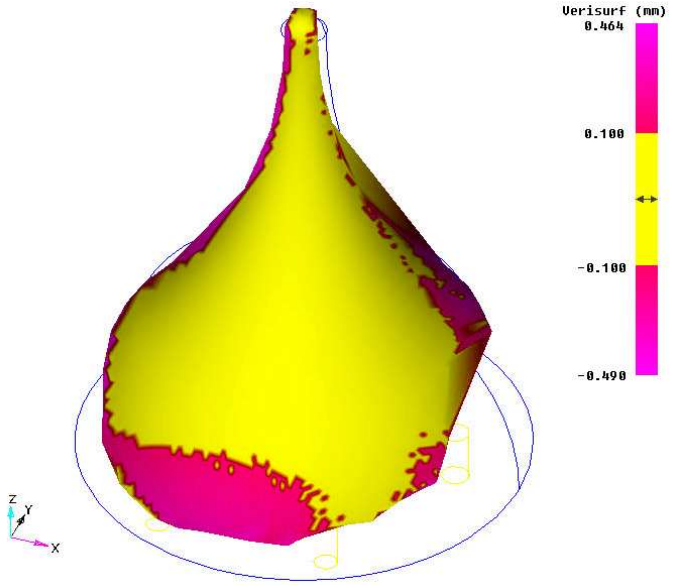
the CAD model is calculated by finding the shortest distance to the CAD model surface for each point in the point cloud. The distribution of the divergence is shown in Fig. 3.25, in which yellow indicates points that are within the designated tolerance and pink indicates points out of tolerance. Statistical analysis of the deviation shows that, 85% of the points are within the tolerance of $\pm 0.1mm$ and 50% of the points are within the tolerance of $\pm 0.05mm$. The RMS value of the deviation is $0.094mm$. As mentioned above, the shape of the part (with paint applied) and the CAD model of the part do not match perfectly, mainly due to the thickness of the paint layer that was applied on the part's surface. Hence, it is safe to say that, the RMS value of the actual deviation between the point cloud acquired and the painted part is less than $0.094mm$.

As can be seen from Fig. 3.25, the regions on the surface of the part with bigger deviation are mostly close to the edge of the point cloud. Referring to Fig. 3.23(c), it can be seen that in these regions, the fringes appear in the image of the part are either too sparse or of low intensity contrast. These problems could lead to bigger errors in the acquired phase map of the part and hence in the constructed point cloud.

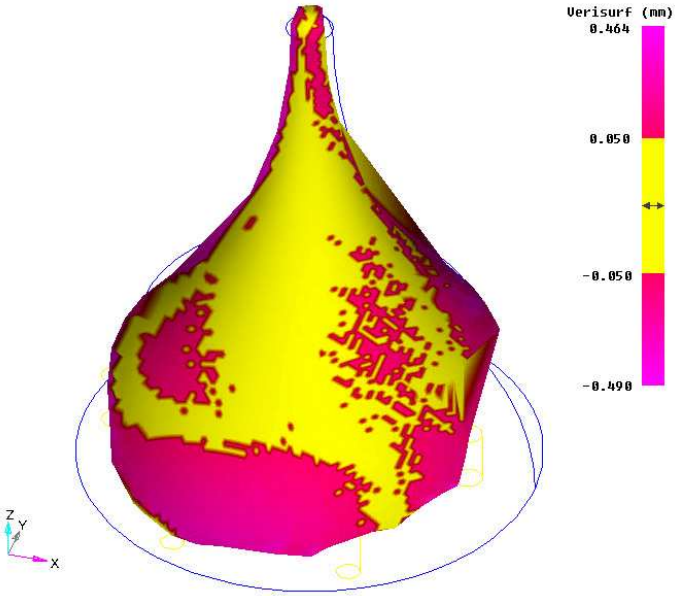
3.4.3.2 Measurement of Gauge Part No. 2

The second gauge part has a pyramid shape and a dimension of around $260mm$ (L) $\times 145mm$ (W) $\times 130mm$ (H). It is made of steel and has five plates mounted on the top and side faces, with holes and slots on them. Figure 3.26(a) shows a picture of the part. The plates on the part were machined very accurately and they are very flat. According to a CMM measurement report, the flatness of the plates ranges from $11\mu m$ to $38\mu m$. Here, the flatness of the plate is defined as the absolute range of the plate's deviation from a perfect plane.

Three plates on the part were measured, namely planes A, B and C, as marked in Fig. 3.26(a). Before measurement, a spray paint was applied to the surface of the



(a) Tolerance = $\pm 0.1mm$



(b) Tolerance = $\pm 0.05mm$

Figure 3.25: Divergence between the acquired point cloud and the CAD model

Plane	Number of Points	Deviation Range (μm)	RMS of Deviation (μm)
A	7546	[-96, 64]	20
B	4025	[-127, 87]	23
C	1909	[-82, 51]	14

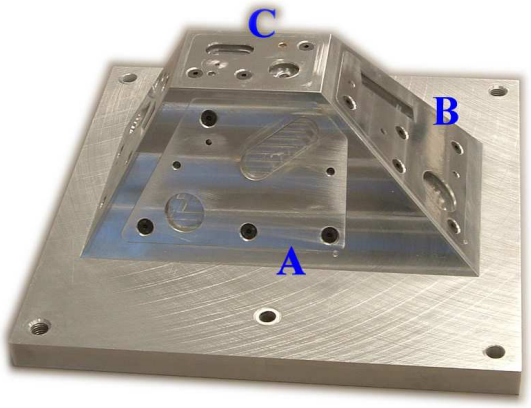
Table 3.1: Plane fitting results of the acquired point cloud

part to avoid specular reflection. An image of the part under fringe projection, which was obtained during the measurement, is shown in Fig. 3.26(b). The constructed point cloud was segmented and points belonging to the same plane (A, B or C) were grouped. Each group of points were fitted to a plane and the deviations of the points from the plane were calculated. Table 3.1 shows the statistical result of the plane fittings, where all values except the number of points are in unit of μm. Considering the unevenness of the thickness of the paint that was applied on the part's surface, the deviations of the plane fittings are actually very small, which indicates that the measurement accuracy achieved by using the developed SMDFP system is very good.

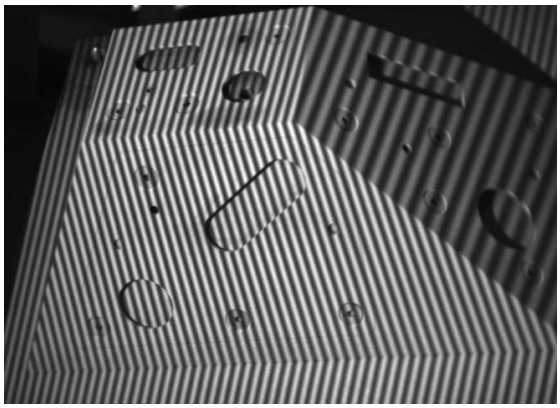
3.4.4 Discussion

The developed SMDFP hardware and software worked well in the series of experiments conducted. With the use of the new algorithm for phase map construction, the number of projection patterns required in a measurement is reduced from 12 to 8, hence the speed of measurement is increased. The algorithm also offers excellent capability of handling surface discontinuities. In the measurements conducted, all cases of surface discontinuity were processed correctly.

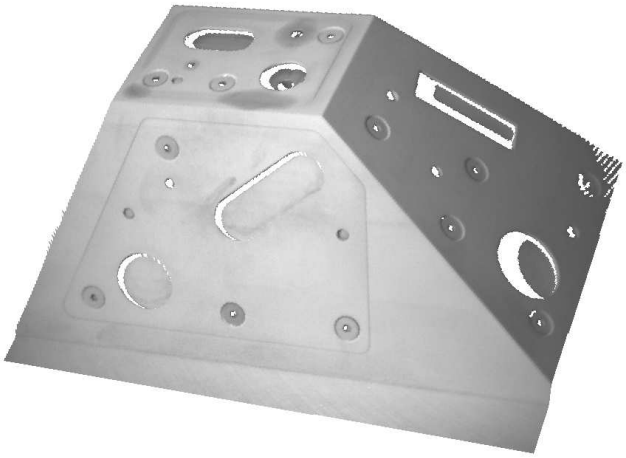
The calibration procedure of the system is fairly easy to carry out by using the proposed parameter estimation algorithms. If after a calibration, the relative positions of the projector and the camera is changed, such as due to the need for a different measurement sensitivity, a recalibration of the system can be done quickly.



(a) Photograph of the part



(b) Image of the part under fringe projection



(c) Rendering of the point cloud constructed

Figure 3.26: Measurement of gauge part No. 2

The measurement result of the gauge parts shows that, the developed SMDFP system is able to achieve a measurement accuracy better than $94\mu m$ over a measurement volume of $250mm$(W) $\times 200mm$(H) $\times 220mm$(D). We believe there is still room for improvement of accuracy by performing a more accurate system calibration.

One of the major improvements that can be made is the plate with checkerboard pattern that was used for camera calibration. In the experiments, we built the plate by printing out a checkerboard pattern on a paper and then pasting it on a flat aluminum plate. The resulting artifact is not as precise as desired due to the uneven pasting and the possible stretching of the paper. A more precise calibration artifact will improve the accuracy of the camera parameters acquired, hence the accuracy of the projector parameters, and eventually the accuracy of shape measurements. Another issue that affects the evaluated accuracy is the paint coating that was applied to the gauge parts before the measurements. The thickness of the paint coating adds additional uncertainties to the shapes of the parts. Therefore, the evaluated accuracy of the system ($94\mu m$), which includes the uncertainties caused by the paint coating, is quite conservative to be taken as the measurement accuracy achieved by the system.

Another issue worth mentioning is that, the $94\mu m$ accuracy achieved is based on a particular hardware configuration. The measurement accuracy a system could achieve also depends on factors other than system model and shape measurement algorithms. For example, accuracy can be improved by placing the projector and camera at a larger angle, or by using projector and camera with higher resolutions.

3.5 Summary

As an emerging technique for 3-D measurements, SMDFP is still in a developing stage and its potentials have not been fully exploited. To make it more attractive to industrial applications, improvements in the following aspects are desired: measure-

ment accuracy, measurement speed, ease of calibration and the capability of handling complex shapes. The mathematical model and algorithms presented in this chapter were developed to achieve these goals.

By including the influence of 3-D perspectives and lens distortions of the projector and camera, the proposed mathematical model provides an accurate modeling of the optical geometry of SMDFP systems, which gives a mathematical basis that is required for achieving high-accuracy shape measurements. Based on this model, three related algorithms were developed, namely the algorithm for construction of absolute phase map, algorithm for construction of point cloud and algorithm for estimation of sensor parameters. By fully exploiting the information in the images obtained from phase-shifting, the developed algorithm for phase map construction uses fewer projection patterns than existing algorithms without compromising performance. As a result, the measurement speed of SMDFP systems is improved. The developed algorithm for point cloud construction requires fewer sensor parameters than similar algorithms in its class, e.g. algorithm proposed by Legarda-Sáenz et. al. [19]. Hence, it simplifies the calibration process of SMDFP systems without compromising accuracy. With the use of a reference phase map, the majority of projector parameters are represented and utilized in an implicit manner in the construction of point clouds. Compared to Legarda-Sáenz's method, which uses projector parameters explicitly, the new approach produces equal or better accuracy, since in general reference phase maps can be acquired more accurately than projector parameters. The new algorithm also handles projector's lens distortions automatically.

The proposed mathematical model and algorithms were implemented in software and a simple SMDFP hardware was also built. Based on the developed system, a number of physical experiments were conducted. Over 30 parts with fairly complex shapes were measured and the results show that the proposed model and algorithms work very well. In the measurements of two gauge parts, the developed system demon-

strated a measurement accuracy of better than $94\mu m$ (RMS of the estimated measurement error) over a measurement volume of $250mm$(W) $\times 200mm$(H) $\times 220mm$(D). It is believed that the accuracy can be improved by performing a more accurate system calibration.

Chapter 4

Simulator for SMDFP Systems

This chapter describes the simulator developed for SMDFP systems, which has been used as a tool for testing the models and algorithms proposed in this dissertation. Section 4.1 explains the basic idea of the simulator, its advantages compared to the experimental and analytical approaches for the study of SMDFP technique, and the feasibility of implementing the simulator to achieve high fidelity. Section 4.2 gives a detailed description on the implementation of the simulator. Some examples of using the simulator for virtual measurements are presented in Section 4.3. A summary of this chapter is given in Section 4.4.

4.1 Introduction

From the input/output point of view, A SMDFP sensor can be considered as a device which takes an object and a projection pattern as inputs and produces the corresponding image of the object under the illumination of the pattern. A simulator of SMDFP systems, as will be discussed in this chapter, is a software for simulating the functionalities of SMDFP sensors, e.g. pattern projection and image formation. It has the same inputs and output as a SMDFP sensor does, while the only differences

are: 1) the input object is given in the form of a geometric model in stead of a physical form; and 2) the output is generated by computations instead of going through a physical imaging process.

A simulator for SMDFP systems is a great tool for system studies and for quick testing of new ideas in this technique. So far, most research work on SMDFP technique uses experiments as the only way to verify the correctness and effectiveness of models and algorithms. As another way to study these issues, simulation approach has its own advantages.

One of the major advantages of simulation approach is that, it can be used to perform a large number of virtual shape measurements with various system setups and objects. Therefore, a comprehensive and systematic study on SMDFP can be carried out while expending only moderate efforts.

Another advantage of simulation approach is its capability to control system components very accurately and as desired, which is normally not possible for operating physical systems. The convenience and complete control of system could greatly facilitate the study of SMDFP systems. Let us take the study of measurement accuracy as an example. A SMDFP system has many error sources, such as noise in the light projection and images and errors in the estimated sensor parameters. For real systems, these error sources are either fixed properties of the hardware or determined by the calibration methods used. Hence the magnitudes of the errors are difficult to control and they are unable to be eliminated. However, with the simulation approach, these error sources can be controlled accurately and individually. As a result, the study of the influence of error sources to the measurement accuracy is much easier.

Compared to the analytical approach for system study, simulation is a much more practical solution, because SMDFP systems are nonlinear systems with a large number of parameters, therefore performing a purely analytical study is very challenging.

The simulation of SMDFP systems can also be performed with high fidelity. The

two major components of SMDFP system, i.e. the projector and the camera, have well established mathematical models which have been proven to be very close to physical laws. The light reflection of surfaces, which is another important aspect in the simulation of SMDFP systems, is also a well studied topic and accurate models are available. Therefore, a high-fidelity simulator for SMDFP systems is feasible, and with such a tool the results of simulations are able to reveal the true situations when real systems are in place.

4.2 Approach

4.2.1 Mathematical Models Used

The simulation of SMDFP systems needs to accurately model three major physical processes: the light projection of the projector, the surface reflection of the object and the image formation in the camera. Associated with these processes, the following mathematical models need to be considered:

- The projector model, which describes the relationship between the 2-D projection pattern and the projected light intensity distribution in the 3-D space;

- The surface reflection model, which describes the relationship between the incident light intensity and the reflected light intensity (to different directions) on the surface of the object;

- And the camera model, which describes the relationship between the 3-D shape of the object, the light intensity reflected from its surface and the corresponding 2-D image formed in the camera.

As mentioned in Sections 2.4 and 3.3.2, the pinhole camera model with the lens distortion model proposed by Brown [49, 50] has been proven to be an accurate mathematical model for both cameras and projectors. Hence, this model is used in the simulator for describing the geometric aspects of the projector and the camera. On top of the pinhole camera model (and the lens distortion model), the defocusing of projector and the intensity attenuation of light along its travel path are also considered.

The surface reflection model used in the simulator is the classic Phong model [70], which is a model widely used in computer graphics for scene rendering. From the physics point of view, Phong model is only an approximate model of surface reflection. However, for the optical phenomena involved in SMDFP technique, Phong model is accurate enough to reveal the influence of surface reflection to shape measurement without causing serious fidelity issues, due to the reasons:

- First of all, the SMDFP technique we have focused on does not involve the use of color information for shape measurement. Hence, the inaccurate color modeling of Phong model is avoided. On the other hand, the surface reflection in terms of light intensities that is described by Phong model is close to physical laws, and that part of Phong model is used in the simulator.

- Second of all, with the use of phase-shifting technique, the measurement of phase values is determined by relative light intensities, not the absolute light intensities. Therefore, a small offset in all light intensity values, such as that caused by inaccurate estimation of the intensity of ambient light, will only cause negligible difference in phase measurement (and hence shape measurement).

- Thirdly, in the case that the projected light intensity is the only variable and everything else is fixed, the principles of Phong model guarantee a linear relationship between the projected light intensity and the reflected light intensity.

This property agrees with the fundamental assumption of the phase-shifting method and is also true for most reflective materials.

Based on these mathematical models, when given the sensor parameters (i.e. the parameters of the projector and the camera), the CAD model of the object, and a grayscale bitmap as the projection pattern, the output image is constructed by using the ray tracing algorithm, which will be explained in detail in the following section.

4.2.2 Ray Tracing Algorithm

The developed simulator for SMDFP systems uses the ray tracing technique to compute the output image when given the sensor parameters of the system, the CAD model of the object being measured, and the projection pattern in the form of a grayscale bitmap image.

Ray tracing is a technique used in computer graphics for generating realistic scene renderings [71, 72]. Compared to the techniques designed for real-time rendering, such as OpenGL, ray tracing is much closer to physical laws although it is also computationally more demanding. The basic idea of the ray tracing technique is the following: In order to compute the light intensity received by a pixel, say pixel (i, j), on the imaging sensor of the camera, a ray is "fired" from the projection center of the camera, going through the center of that pixel and shooting into the space; If the ray intersects with a reflective surface in the space, say at point M, the received illumination intensity at M will be calculated; The light intensity received by pixel (i, j) can then be calculated based on the illumination intensity and surface normal direction at M, the position of M, and some of the camera parameters.

In the simulator developed, the ray tracing algorithm used has been tailored in the following aspects:

- **Integration effect of the imaging sensor**: The pixels on the imaging sensor

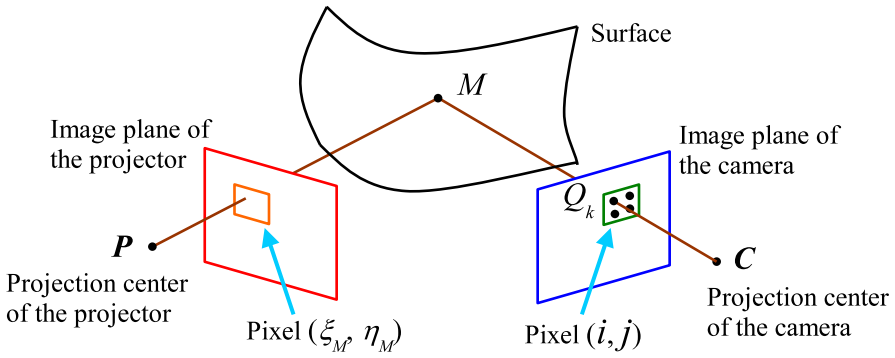


Figure 4.1: Computing the light intensity received by a pixel on the imaging sensor

(usually a CCD or CMOS sensor) of a camera have finite sizes. Each pixel collects the light that is reflected from a finite area on the object's surface instead of from a single "sharp" point. To accurately reflect this, the ray tracing algorithm needs to fire a "beam", instead of a ray, from a pixel. The light reflected from the surface facet that intersects with the "beam", needs to be integrated to be given as the light intensity received by the pixel.

In the ray tracing algorithm implemented in the simulator, this process is approximated by firing multiple rays from a single pixel and then integrating the light intensity values returned by the individual ray tracings. The selection of sample points inside a pixel and the integration scheme complies with the Legendre-Gauss quadrature. A schematic diagram of this procedure is shown in Fig. 4.1 and a detailed description is as follows:

1. For each pixel (i, j) on the camera's image plane, the positions of a set of 2-D Gauss quadrature nodes are calculated. Let $Q_k(i, j)$ denote the nodes for pixel (i, j), where $k = 1, \ldots, K$.
2. For each node $Q_k(i, j)$, a ray tracing process is carried out and the corresponding light intensity received, denoted by $I_k(i, j)$, is calculated.

3. The overall light intensity received by pixel (i, j) is calculated using following equation:

$$I(i,j) = \sum_{k=1}^{K} w_k I_k(i,j) \tag{4.1}$$

where $w_k (k = 1, \ldots, K)$ is the weight for node Q_k as defined in Legendre-Gauss quadrature.

- **Illumination by the projector**: In SMDFP technique, the major light source is the projector. As explained earlier, the optical geometry of a projector can be described accurately by the pinhole camera model with the lens distortion model. Hence, the illumination intensity at any point in the space that is provided by a projector can be calculated based on this model. Taking the scenario depicted in Fig. 4.1 as an example, the illumination intensity at point M provided by the projector is determined by the intensity of pixel (ξ_M, η_M) in the projection pattern, which is the projected position of point M on the projector's image plane. With the consideration of the defocusing issue of the projector, the situation will be slightly different, which will be discussed in more detail in Section 4.2.3.

  Another issue that needs to be addressed regarding the illumination is the shadow. If point M can be actually illuminated by the projector or shadowed by some part of the object itself. This can be easily checked by conducting another ray tracing process starting from the projector, i.e. to check if the ray can reach point M or is blocked by a surface on the way.

- **Calculation of the reflected light intensity**: As stated earlier, the surface reflection model used in the simulator is the Phong model. Since none of the processes in the simulator involves colors (also true for the real systems we have focused on), the color modeling in Phong model is ignored. Except that, the rest

of the components in Phong model, such as the modeling of diffuse reflection, specular reflection and ambient light, are all considered in the calculation of the reflected light intensity. The intensity attenuation of the light along its travel path, i.e. from the projector to the object's surface and from the surface to the camera, is also taken into account.

4.2.3 Simulation of Projector Defocusing

Pinhole camera model assumes infinitesimal lens aperture and hence every point in the 3-D space is mapped to a single point in the image plane and vice versa. No focusing issue is involved. For a projector, this means every point in the projection pattern is projected as a sharp point in the 3-D space, not matter at what distance the projection is formed. In reality, this is not true due to the finite lens aperture of a projector, in which case a point in the projection pattern would only be projected as a sharp point at a specific distance to the projector, i.e. the in-focus position. Away from that position, the projection would gradually become blurred, which is often referred as the defocusing issue.

Most commercial computer projectors have large lens apertures, therefore the defocusing issue of the projector needs to be considered in the simulation to achieve high fidelity. For the cameras used in SMDFP systems, they are generally operated at very small lens apertures, due to the high illumination intensity provided by the projectors. Hence the defocusing issue of the camera can be safely ignored within the measurement volume of the system.

A popularly used defocusing model can be found in literature [73, 74]. The basic idea of this model is the "blur circle" concept. In the context of a projector, this concept can be interpreted as that, if the light projection from a point on the projector's image plane is not received at the focus distance, it would form a blurred

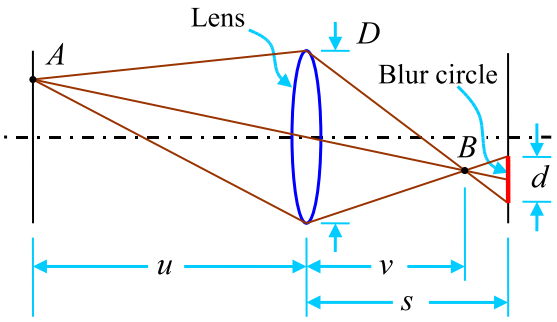


Figure 4.2: The definition of blur circle in the defocusing model

spot instead of a sharp point. A schematic diagram of the "blur circle" concept is shown in Fig. 4.2, where A is a point on the projector's image plane and B is the focused projection of A in the space. If the surface receiving the projection is placed away from B, the light coming from A would spread out uniformly in a circular area as shown in the figure. The diameter of the "blur circle", d, which indicates the magnitude of defocusing, can be calculated using the following equation:

$$d = \frac{D}{v}|s - v| \tag{4.2}$$

where D is the the lens aperture, v is the focus distance (of A), and s is the distance from the surface to the lens.

When projector defocusing is considered, the calculation of the illumination intensity (given by the projector) at a point, M, in the space can be done through the following procedure:

- Firstly, the diameter of the blur circle at point M, denoted by d_M, can be calculated using following equation, which is derived from Eqn. 4.2:

$$d_M = \frac{D}{v^{(P)}}\left|z_M^{(P)} + v^{(P)}\right| \tag{4.3}$$

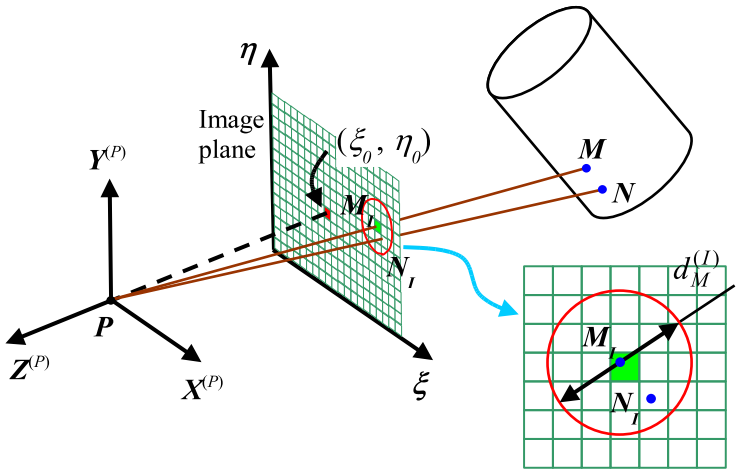


Figure 4.3: Calculation of illumination intensity when projector defocusing is considered

where $v^{(P)}$ is the focus distance, which is given as a projector parameter, and $z_M^{(P)}$ is the z-coordinate of point M in the projector coordinate frame, $X^{(P)}Y^{(P)}Z^{(P)}$.

- Let M_I denote the projection of point M on the image plane of the projector. From the defocusing model described above, it can be known that point M is not only illuminated by M_I but also by the points in the close neighboring area of M_I. This neighboring area is a circular region centering at M_I and with a diameter given by the following equation:

$$d_M^{(I)} = \frac{d_M}{\left|z_M^{(P)}\right|} = \frac{D}{v^{(P)}} \left| \frac{z_M^{(P)} + v^{(P)}}{z_M^{(P)}} \right| \tag{4.4}$$

- The illumination intensity at point M, $I_M^{(P)}$, can then be calculated from an integration of the projection intensities in the circular region defined by M_I and

$d_M^{(I)}$ in the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane:

$$I_M^{(P)} = \iint\limits_{(\xi-\xi_M)^2+(\eta-\eta_M)^2 \leq \left(d_M^{(I)}\right)^2} I^{(P)}(\xi,\eta)\, d\xi d\eta \tag{4.5}$$

where (ξ_M, η_M) are the coordinates of point M_I on the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane (the projector's image plane) and $I^{(P)}(\xi,\eta)$ is the light intensity at (ξ,η). One thing worth mentioning is, the influence of light intensity attenuation will be added to the value of $I_M^{(P)}$ before being taken as the final illumination intensity received at point M.

4.2.4 Emulation of Hardware Noise

In the simulation study of SMDFP technique, it is often required to emulate the hardware noise that existed in real systems, such as to study the influence of error sources to the accuracy of shape measurement. Despite the varied types of hardware noise involved in SMDFP systems, many of them can be emulated by noise in either the input projection pattern or the output image. For example, noise with different distributions in the projection pattern can be used to emulate the noise caused by the DMD/LCD chip of the projector, the unevenness of the projector's light source, and the fluctuation of environmental light.

4.3 Implementation and Results

The proposed simulator for SMDFP systems was implemented in C++. A schematic diagram of the major components of the simulator, as well as the inputs and output, are shown in Fig. 4.4. The ACIS® geometric kernel [75, 76] is used in the ray tracing algorithm to acquire the positions, surface normal directions of the intersection points

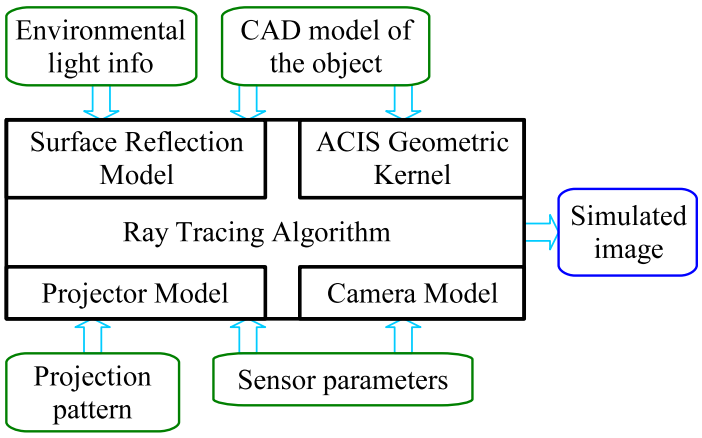


Figure 4.4: Schematic diagram of the major components in the simulator

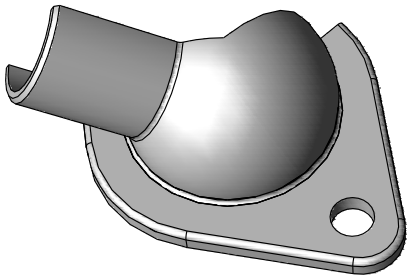
of rays with the CAD model. It is also used to retrieve the surface reflection properties of the part if provided in the CAD model.

A number of images created by using the simulator with different parts and under different conditions are presented in Fig. 4.5 and 4.6.

4.4 Summary

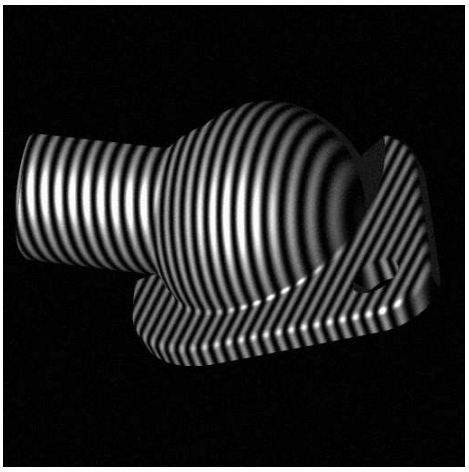
The simulator for SMDFP presented in this chapter uses proven models and algorithms for the simulation of the pattern projection and image capture processes of SMDFP systems. The pinhole camera model with lens distortion model used is known to be very accurate for describing the optical geometries of cameras and projectors; and the ray tracing algorithm adopted is a well-known technique for realistic scene rendering. Therefore, the fidelity of the simulator is guaranteed by the proven accuracy of the models and algorithms used.

The developed simulator has been used in the research conducted in this dissertation on a few topics, e.g. the development of system models and algorithms (Chapter 3), the estimation of measurement uncertainties (Chapter 6) and the use of

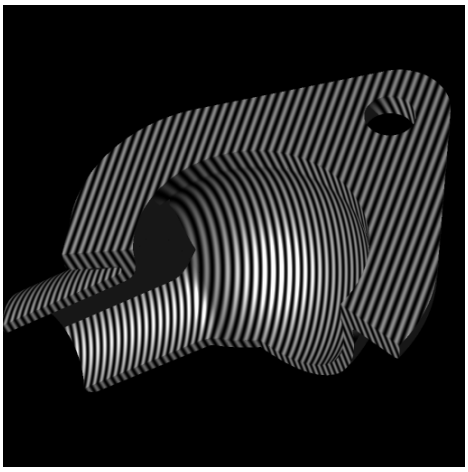


(a) A rendering of the CAD model of the part

(b) Image of the part under fringe projection (diffuse surface reflection)

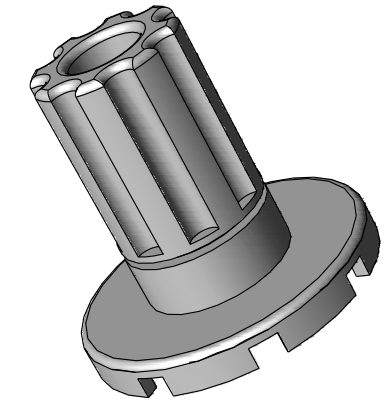


(c) Image of the part under fringe projection (specular surface reflection, noise added)

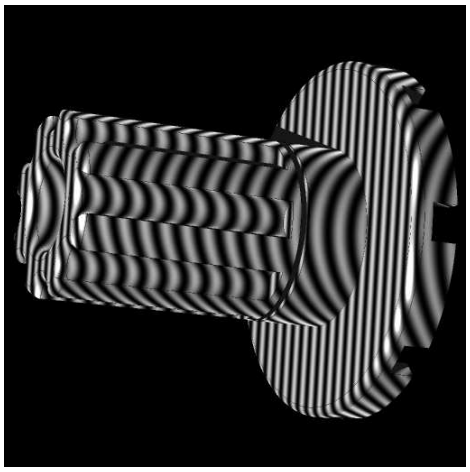


(d) Image of the part under fringe projection (part placed at a different angle, pattern with a difference fringe number used)

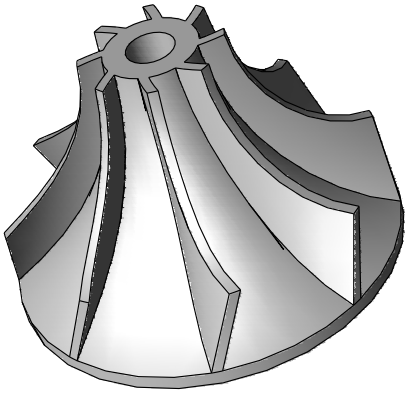
Figure 4.5: Simulated images of a part under different conditions



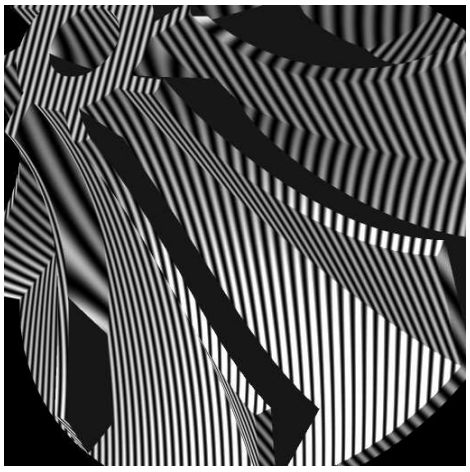
(a) CAD model of part S02



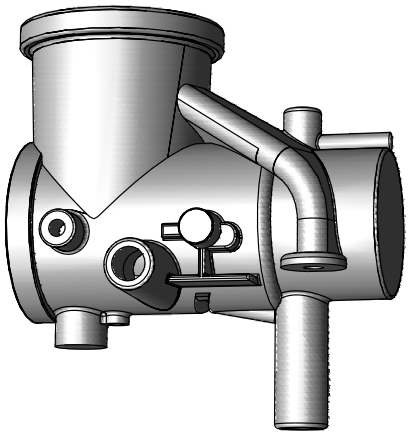
(b) Image of part S02 under fringe projection



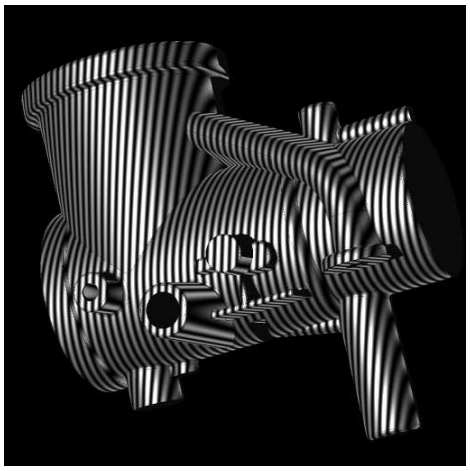
(c) CAD model of part S03



(d) Image of part S03 under fringe projection



(e) CAD model of part S04



(f) Image of part S04 under fringe projection

Figure 4.6: Simulated images of varied parts under fringe projections

adaptive projection patterns (Chapter 5). It has shown to be a very useful tool for testing new models and algorithms and for system studies. For all the research work where simulation was involved, experiments were also carried out to verify the results acquired and the conclusions drawn from simulations. The comparisons showed that, the results given by the two approaches match very well, which is another proof that the developed simulator is of very good fidelity.

Chapter 5

Adaptive Projection Patterns

This chapter describes the use of adaptive projection patterns in SMDFP technique. Section 5.1 describes the basic idea of adaptive projection pattern as well as its advantages compared to fringe patterns with fixed fringe pitches. Section 5.2 explains the details of using adaptive projection patterns for shape measurements, including the measurement procedure, prospective applications and the requirements on system model and algorithms. Details of two new algorithms are presented. Section 5.3 describes how to integrate the adaptive patterns to the current system infrastructure. Section 5.4 summaries this chapter.

5.1 Introduction

5.1.1 Problems Associated with Fixed-Pitch Fringe Patterns

Most existing SMDFP systems use straight fringe patterns with fixed fringe pitch for projections. The intensity distribution in the patterns is a periodic function (e.g. sinusoidal or square waveform) in one direction and uniform in the orthogonal direction. A popularly used type is sinusoidal fringe pattern, as the one shown in

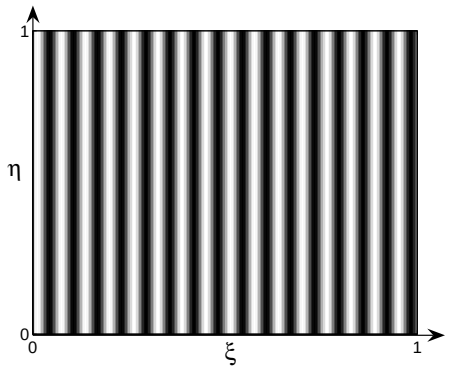


Figure 5.1: A projection pattern with fixed fringe pitch

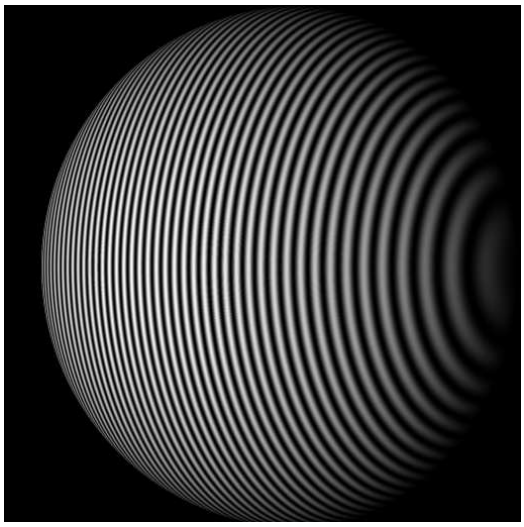
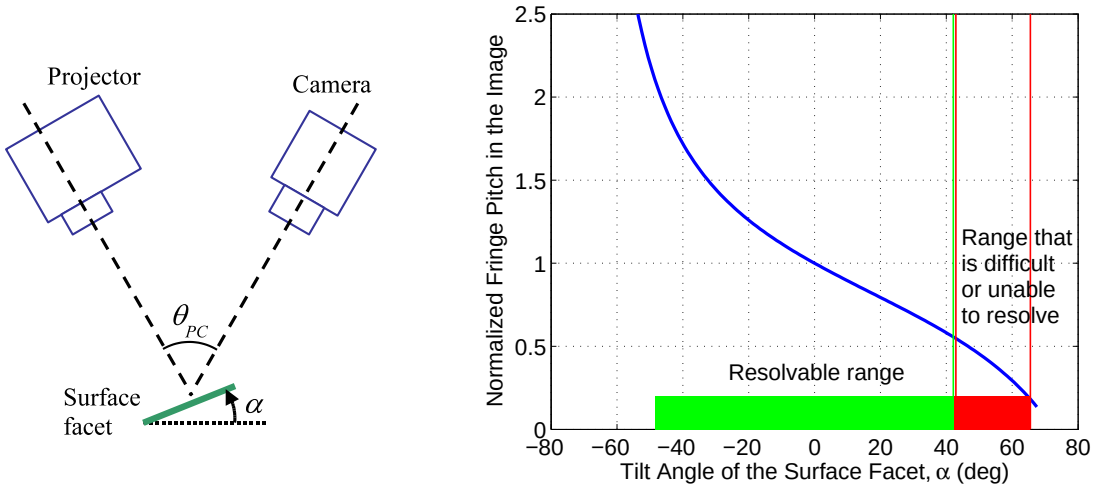


Figure 5.2: Image of a sphere under the projection of a fixed-pitch fringe pattern

Fig. 5.1. The intensity distribution in the pattern is sinusoidal along $\boldsymbol{\xi}$-axis and uniform along $\boldsymbol{\eta}$-axis. Here, "fixed fringe pitch" refers to the pitch of fringes in one pattern. In systems using temporal phase unwrapping techniques (see Section 2.3.2.4), multiple fringe patterns with varied fringe pitches are used. Although the fringe pitch used may vary from pattern to pattern, it remains to be a constant in each pattern.

When projecting a fixed-pitch fringe pattern on an object, the corresponding image generally shows curved fringes with the pitch of fringes varies across the image. Figure 5.2 shows the image of a sphere under the projection of a sinusoidal fringe pattern. The fringes in the image are bent and the fringe pitch gets larger from the left to the right. When the measurement setup and the fringe pattern used are fixed, the local fringe pitch in the image depends on the normal direction and position of the object's surface facet w.r.t. the sensor. Figure 5.3 shows an example of the relationship between the local fringe pitch and the normal direction of surface facet. In this particular case, the angle between the optical axes of the camera and the projector, denoted by θ_{PC}, is 35^o. A small plate, representing a surface facet, is



(a) Schematic diagram of the measurement setup

(b) Fringe pitch vs. the tilt angle of surface facet, α

Figure 5.3: Relationship between the local fringe pitch in image and the normal direction of surface for a typical measurement setup

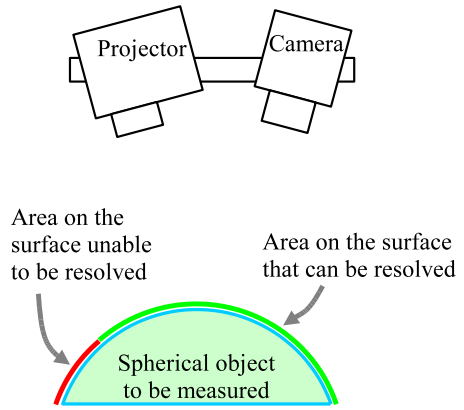
placed at the intersection of the optical axes of the camera and the projector (see Fig. 5.3(a)). The tilt angle of the plate is α, where $\alpha = 0$ is the plate orientation perpendicular to the bisection of the optical axes. Under the projection of a sinusoidal fringe pattern, the fringe pitch appearing in the image changes as the facet rotates. The relationship between the fringe pitch (normalized by the fringe pitch value at $\alpha = 0$) and the tilt angle of the facet, α, is shown in Fig. 5.3(b). As α increases from around 40^o to above 60^o, the normalized fringe pitch turns less than 0.5 and decreases rapidly thereafter.

Fixed-pitch fringe pattern is easy to model and able to achieve good measurement accuracy in most cases. For many mathematical models of SMDFP, fixed-pitch fringe patterns are the only patterns that can be used. However, with the restriction of constant fringe pitch in a pattern, it is sometimes difficult to achieve both good accuracy and maximum measurement coverage from a single measurement.

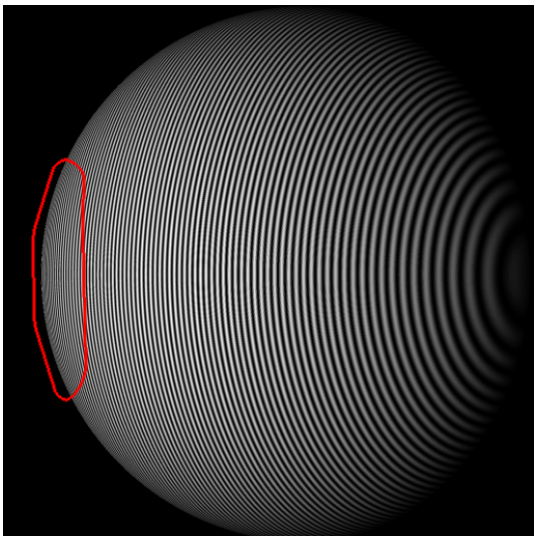
Figure 5.4 shows the problem of fixed-pitch fringe patterns when measuring a

spherical object. With the measurement setup as shown in Fig. 5.4(a), the complete top surface of the spherical object is visible to both the camera and the projector. Therefore, theoretically the whole top surface can be measured. However, due to the object's large range of surface normal direction, the local fringe pitch appearing in the image varies significantly from area to area. As a result, the fringes in certain areas of the images are so crowded that the construction of phase map (and hence the construction of point cloud) would fail at these areas. Figure 5.4(b) shows an image of the spherical object obtained in measurement under the projection of a fringe pattern with 121 fringes. The area in the image circled by red line is where the phase map construction fails. The fringes inside that area are visually indistinguishable. The overcrowding of fringes in the images can be solved by using projection pattern with fewer fringes. Figure 5.4(c) shows an image of the object under the projection of a pattern with 49 fringes. In this case, the complete top surface can be resolved in the measurement. However, since the fringe number in the projection pattern has decreased, the overall measurement accuracy dropped accordingly. Figure 5.4(d) shows the RMS of measurement error and the size of unresolvable area (in number of pixels) when different number of fringes are used in projection. As the fringe number increases from 49 to 121, the measurement error drops while at the same time the size of unresolvable area increases.

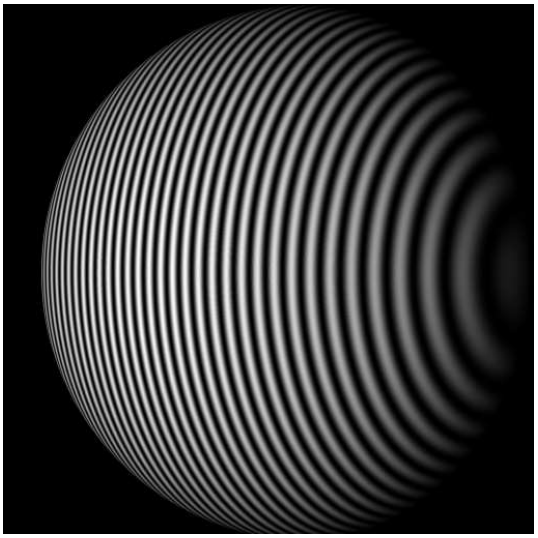
In conclusion, when fixed-pitch fringe patterns are used in measurements of surfaces with dramatically changing normal directions, a compromise between measurement accuracy and measurement coverage has to be made. In many occasions, in order to guarantee a certain level of measurement accuracy, some areas on the surface may not be resolvable due to the overcrowding of fringes.



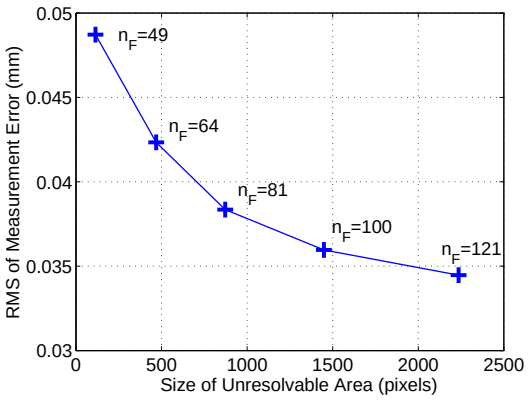
(a) The measurement setup and the resulting measurement coverage



(b) The phenomenon of fringe overcrowding in left edging area of the images ($n_F = 121$)



(c) No fringe overcrowding when using projection pattern with a much smaller fringe number ($n_F = 49$)



(d) Measurement error and the size of unresolvable area when different fringe numbers are used

Figure 5.4: Problem with fixed-pitch fringe patterns when measuring a spherical object

5.1.2 Possible Solutions with Fixed-Pitch Fringe Patterns

There are a few solutions to achieve both good accuracy and maximum measurement coverage while still use fixed-pitch fringe patterns. However, these solutions also introduce certain amount of overhead or have other disadvantages which make them inappropriate for applications that require a minimum measurement time.

One of the solutions is to move the sensor (or equivalently the object) to measure the surface areas from a second perspective. Take the measurement of the spherical object as an example. As illustrated in Fig. 5.5, when the sensor is at position 1, it is able to measure most part of the object's top surface (shown in the figure in green) except the far left portion (shown in red). This unresolvable portion, however, can be measured by moving the sensor to position 2 where a second measurement can be made. By merging the results from the two measurements, a point cloud representing the complete top surface of the object can be generated. The disadvantage of this solution is that, either the sensor or the object has to be moved during the measurement even though the complete surface of the object is visible to the sensor (which means, visible to both the camera and the projector) from a single perspective. The moving of the sensor or the object prolongs the measurement time and introduces additional errors.

Another solution is to maintain the positions of the sensor and the object, and use additional projection pattern(s) to resolve the areas that are unresolvable by using "regular" fringe pattern. Again take the measurement of the spherical object as an example. As explained earlier, the unresolvable area caused by fringe overcrowding in the images can be resolved by using patterns with a fringe number smaller than regular. Figure 5.4(b) shows an image of the spherical object under a projection pattern with 121 fringes, which is the regular case. Most part of the surface can be measured with good accuracy except the far left area (circled in red) which is unresolvable.

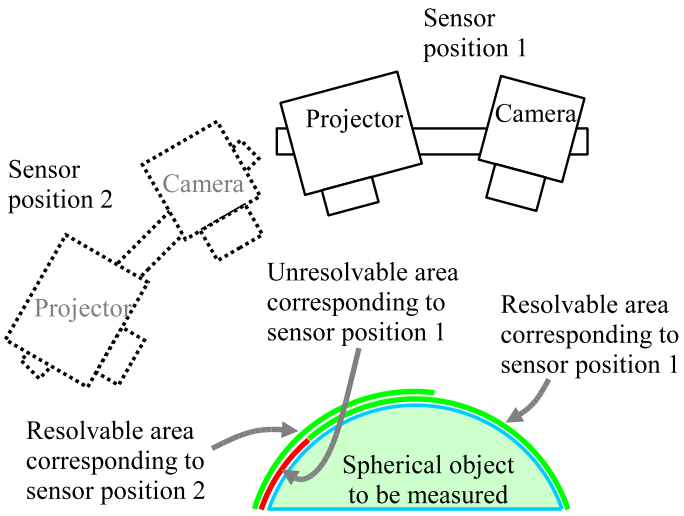


Figure 5.5: Move the sensor to achieve full measurement coverage

However, this failed area can be resolved by using a pattern with 49 fringes (an image is shown in Fig. 5.4(c)), although in this case the measurement accuracy for the rest of the surface is not satisfactory. Therefore, the complete surface can be measured by using a regular pattern with large fringe number plus an additional pattern with a smaller fringe number, followed by a merging of two point clouds. The final result accomplishes both good accuracy and full measurement coverage. The disadvantage of this solution is the increased measurement time due to additional pattern projection and image acquisition. Using the projection scheme proposed in Section 3.3.3.2 as an example, it means that 4 more projections have to be made in addition to the original 8 projections, which results in a 50% increase in measurement time.

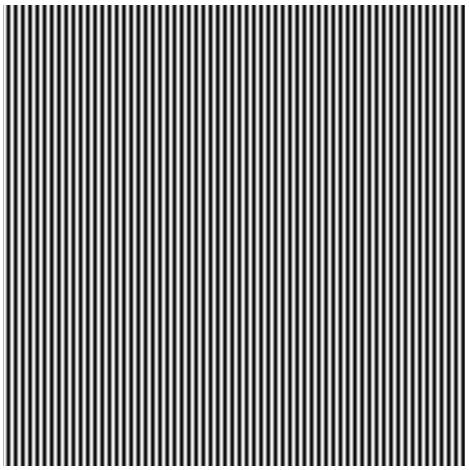
5.1.3 Projection Patterns with Variable Fringe Pitch

A projection pattern with variable fringe pitch is a fringe pattern in which the local fringe pitch may vary from area to area. Figure 5.6(c) shows an example of such patterns. The fringe pitch in the middle left area of the pattern is significantly larger

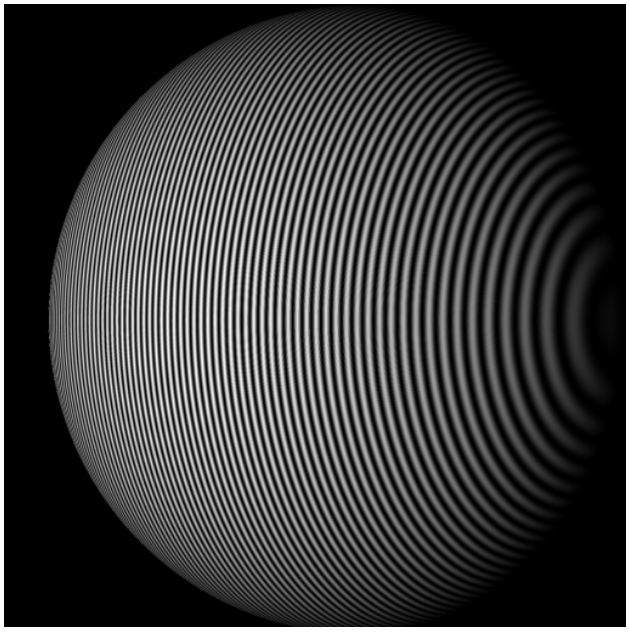
than the fringe pitch in the rest of the pattern. An adaptive projection pattern is a pattern with variable fringe pitch that is designed for a particular object and measurement setup to achieve improved performance.

Adaptive projection patterns can solve the problem of fringe overcrowding in images as mentioned earlier with fixed-pitch fringe patterns. Recall the idea of using pattern with reduced fringe number to solve the problem of fringe overcrowding, which uses different fringe pitches to resolve different areas on the object's surface. The idea of variable fringe pitch is to blend multiple fringe pitches into a single pattern, instead of using two or more fixed-pitch fringe patterns. Different fringe pitches are assigned to different areas in the projection pattern such that the local fringe pitch in the image (of the object) is adequate, meaning that no fringe overcrowding that creates unresolvable areas. Figure 5.6 shows the comparison of images of a spherical object under the projection of a fixed-pitch fringe pattern and an adaptive fringe pattern. In the image obtained from adaptive projection pattern, the fringes are more evenly distributed. No overcrowding of fringes in the far left area as the case with fixed-pitch fringe pattern. For clear display, both projection patterns are shown with uniformly increased fringe pitch than actual.

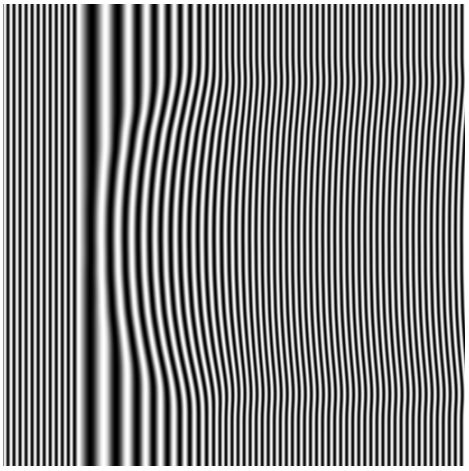
Comparing to the methods for solving fringe overcrowding with fixed-pitch fringe patterns, as explained in the previous section, the approach with adaptive fringe patterns requires no movement of the sensor or the object and no additional projections. Therefore, it is able to achieve the maximum measurement coverage without sacrificing measurement speed. The overall measurement accuracy achievable by using adaptive projection patterns is very close to what was achieved by using multiple fixed-pitch fringe patterns. Figure 5.7 shows the performance comparison between fixed-pitch fringe patterns and adaptive fringe pattern in the measurement of a spherical object. The x-axis represents the size of unresolvable area in number of pixels (total number of pixels is 512×512) and the y-axis represents the RMS of mea-



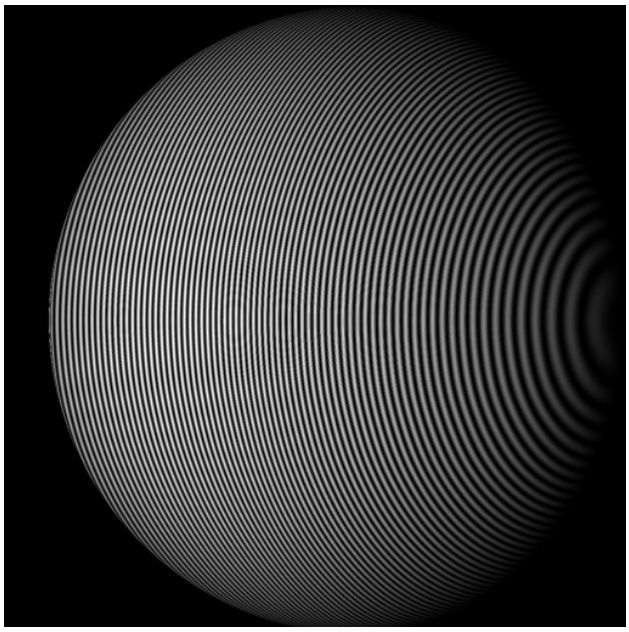
(a) Fixed-pitch fringe pattern (shown with uniformly increased fringe pitch than actual)



(b) Image obtained by using fixed-pitch fringe pattern



(c) Adaptive fringe pattern (shown with uniformly increased fringe pitch than actual)



(d) Image obtained by using adaptive fringe pattern

Figure 5.6: Using adaptive fringe patterns to solve the problem of fringe overcrowding

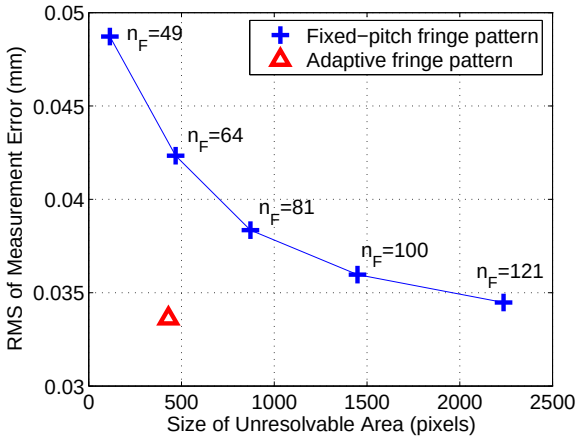


Figure 5.7: Measurement performance: adaptive fringe pattern vs. fixed-pitch fringe patterns in the measurement of a spherical object

surement error. The measurement performance of fixed-pitch fringe patterns with different fringe numbers (n_F) are shown in blue crosses. The adaptive fringe pattern (indicated by a red cross) achieved a better accuracy than the fixed-pitch fringe pattern with 121 fringes, while at the same time it has a measurement coverage better than the fixed-pitch fringe pattern with 64 fringes.

The measurement method with adaptive projection pattern needs a process to automatically generate an adaptive projection pattern for the measurement. It will be shown that, the additional processes will not introduce extra measurement time for certain types of applications.

5.2 Measurement Using Adaptive Projection Patterns

5.2.1 Measurement Procedure

When fixed-pitch fringe patterns are used, the measurement of an object using SMDFP includes the following steps: 1) projects phase-shifted fringe patterns on the object

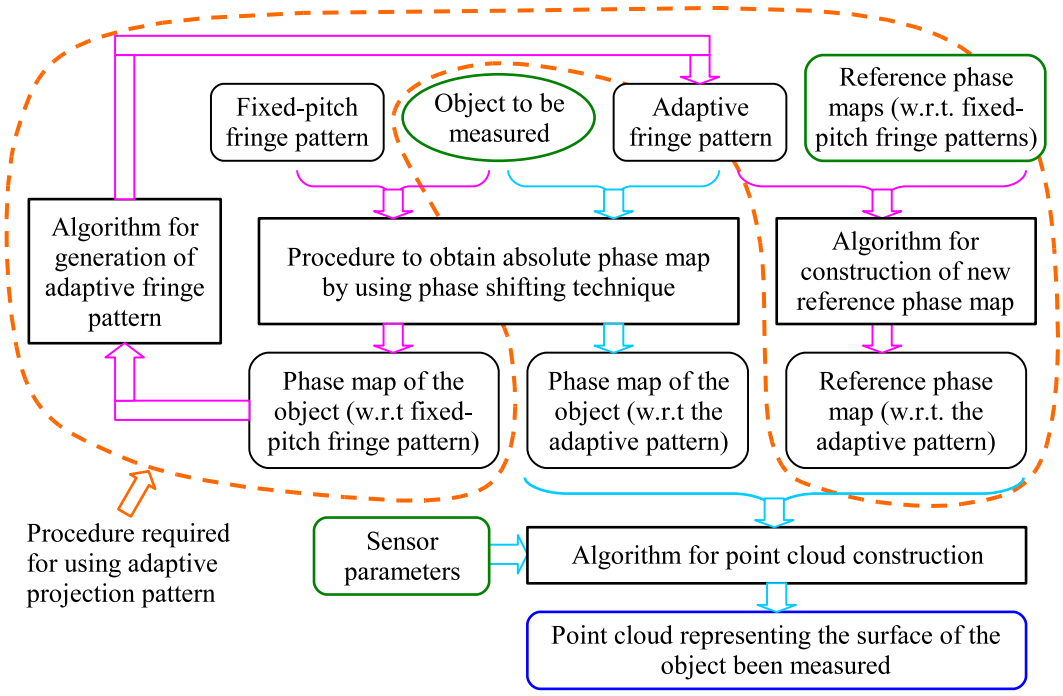


Figure 5.8: Schematic diagram of the measurement workflow when adaptive projection pattern is used

and records the images; 2) constructs an absolute phase map of the object from the images acquired; and 3) constructs a point cloud from the absolute phase map of the object. In case of adaptive projection patterns being used, the measurement procedure requires additional steps for the generation of an adaptive pattern. In order to integrate the use of adaptive patterns to the system infrastructure, a new measurement procedure is proposed. A schematic diagram of the new procedure is shown in Fig. 5.8. A detailed explanation of it is given at below.

- **Generation of adaptive projection pattern**:

  For objects with different shapes or placed at different positions, the adaptive projection patterns that could give optimal measurement performance may be different. Hence, if the shape or position of the object being measured has changed, the adaptive projection pattern needs to be regenerated. The algo-

rithm for the generation of adaptive patterns requires knowledge of the approximate shape of the object, as well as its position and orientation w.r.t. the sensor. These information can either be provided to the system in the form of a geometric model of the object, or acquired by a preliminary measurement. In the latter case, the measurement can be conducted using a fixed-pitch fringe pattern with a medium number of fringes to avoid any possible unresolvable area. This measurement is not intended to give accurate result of the object's shape, however it is appropriate for the generation of adaptive pattern.

The method proposed here for the generation of adaptive pattern is based on the preliminary measurement approach (see Fig. 5.8). First, an absolute phase map of the object is obtained by using a fixed-pitch fringe pattern with a medium fringe number. From the acquired phase map, the areas in the images of the object where the fringe density is too high can be identified. The corresponding areas in the projection pattern can then be located and larger fringe pitches are assigned to these areas in the adaptive projection pattern. Finally, the adaptive pattern is examined from an overall perspective and adjusted accordingly to assure compatibility with the point cloud construction algorithm. A detailed explanation of this algorithm is presented in Section 5.2.4.2.

- **Construction of new reference phase map**:

  The point cloud construction algorithm proposed in this dissertation (see Section 3.3.4 for details) requires a reference phase map, which is an absolute phase map of a flat plane under the projection pattern used for measurement. When a new projection pattern is generated and to be used in future measurements, the reference phase map needs to be reconstructed. The construction of the new reference phase map can be done through an interpolation based approach which takes the adaptive projection pattern as input as well as two "regular"

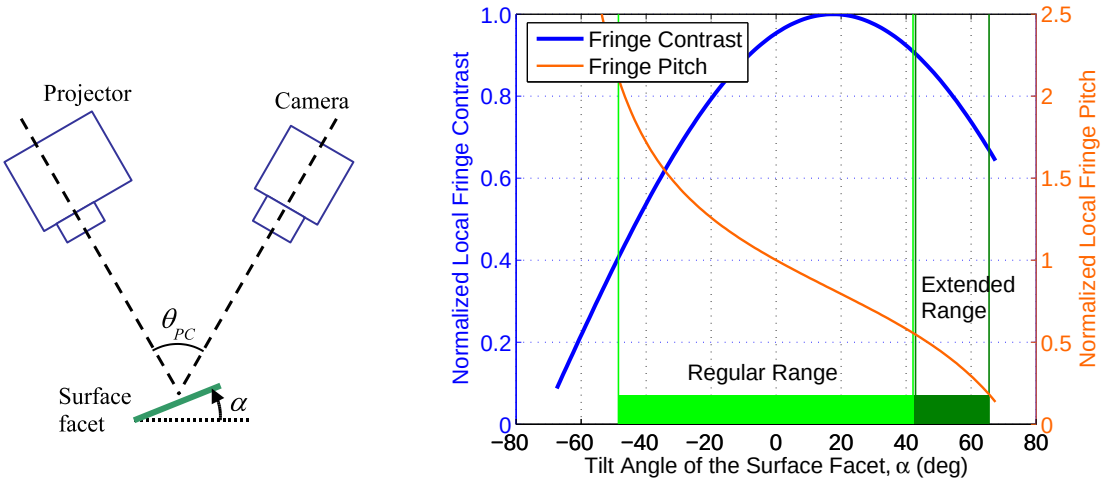
reference phase maps that are obtained using fixed-pitch fringe patterns. A detailed description of this algorithm is presented in Section 5.2.5.

- **Measurement of the object(s)**:

  Once the adaptive projection pattern has been generated and the corresponding reference phase map has been acquired, the procedure of measurement by using the adaptive pattern is the same as with fixed-pitch fringe patterns. That is, a set of phase-shifted adaptive patterns will be projected on the object and the corresponding images will be recorded; From these images, an absolute phase map of the object, and hence a point cloud that represents the object's surface, can be constructed.

5.2.2 Potential Applications

The essential benefit of using adaptive projection pattern is the capability of dealing with surfaces with a large range of surface normal directions. This can be explained by the example shown in Fig. 5.9 (which has been discussed briefly in Section 5.1.1). When a measurement sensor is fixed, the normal direction of a surface facet mainly affects two things in the acquired image: One is the local fringe contrast that corresponds to that facet, and the other is the local fringe pitch. Consider a measurement setup as shown in Fig. 5.9(a), in which the optical axes of the camera and the projector forms an angle of θ_{PC} and intersects at where the surface facet is located. Let α denote the tilt angle of the surface facet, where $\alpha = 0$ is the normal direction that is parallel to the bisection of the optical axes of the camera and the projector. The influence of the facet's normal direction to the local fringe contrast and fringe pitch in this case is shown in Fig. 5.9(b), in which both the fringe contrast and the fringe pitch are normalized by their values at $\alpha = 0$. In a real measurement, the surface to be measured is usually place at an angle such that the major portions of the surface



(a) Schematic diagram of the measurement setup

(b) Normalized local fringe contrast and fringe pitch vs. the tilt angle of surface facet, α

Figure 5.9: Relationship between the local fringe pitch in image and the normal direction of surface for a typical measurement setup

have normal directions at around $\alpha = 0$, because in this way the complete image can have an overall high fringe contrast, which is desirable for the acquisition of an accurate phase map. Due to the requirement of a relatively high fringe contrast, the range of the surface facet's tilt angle α that the sensor can deal with is approximately from -50^o to 80^o. However, if a fixed-pitch fringe pattern is used for the measurement, the upper limit of α will be reduced to around 40^o due to the problem of fringe overcrowding for $\alpha > 40^o$, as been discussed earlier[1]. On the other hand, the use of adaptive projection pattern can avoid the problem of fringe overcrowding and extend the upper limit of α to around 65^o. In Fig. 5.9(b), the range of surface normal direction that can be handled by using fixed-pitch fringe pattern is marked as "regular range", and the additional range that can be gained by using adaptive projection pattern is marked as "extended range".

In summary, when the highest possible measurement accuracy is desired, the use

[1]As can be seen from Fig. 5.9(b), for α greater than 40^o, the normalized fringe pitch drops below 0.6, which could cause fringe overcrowding (see Section 5.1.1)

of adaptive projection pattern is able to extend the range of surface normal direction that can be measured with a single pattern by approximately 20%. Hence, adaptive projection patterns are favorable for measurements of surfaces with a large range of surface normal direction.

The use of adaptive projection pattern in the measurement of an object requires the generation of adaptive pattern. When the approximate shape of the object is not known, a preliminary shape measurement is needed for the pattern generation, adding extra time to the measurement. Therefore, if the adaptive pattern has to be regenerated this way every time a new measurement takes place, the use of adaptive pattern will not provide any significant advantage over fixed-pitch fringe pattern, since similar measurement accuracy and coverage can be possibly achieved in the same time frame by using multiple fixed-pitch fringe patterns. For applications such as on-line parts inspection, the shapes of objects, as well as their positions and orientations w.r.t. the sensor, are very similar from measurement to measurement. The adaptive projection pattern needs to be built only once and can then be used for many successive measurements. In such case, the extra time required for pattern generation is trivial and the gain in overall measurement speed by using adaptive pattern is remarkable.

The preliminary phase map of the object, which is required for generation of adaptive pattern, is obtained by using fixed-pitch patterns with a large fringe pitch. The accuracy of the phase map is not very high, which may result in less optimized adaptive pattern being generated. In the proposed algorithm for generation of adaptive patterns, 2-D filtering of the generated patterns has been used in order to maintain adequate pattern profiles as required by the algorithm for point cloud construction. As a result, the generated adaptive patterns may not produce optimized performance for features on surfaces that are smaller than a certain size.

In conclusion, the use of adaptive projection patterns is an ideal solution for on-line inspection of parts with a large range of surface normal directions.

5.2.3 Requirements on System Model and Algorithms

The use of adaptive projection patterns in measurements requires the support of adequate mathematical models and algorithms. Some of the existing models for SMDFP technique, such as the optical geometry based models as explained in Section 2.1.1, may require major modifications in order to incorporate adaptive projection patterns. A short explanation of the reason is that, for a fixed-pitch fringe pattern, the phase contours in the pattern are straight lines; while in the case of adaptive pattern, the phase contours are generally curves and hence more difficult to deal with.

The mathematical system model, as well as the algorithms for phase map construction and point cloud construction, that were proposed in Chapter 3, can work seamlessly with adaptive projection patterns. The two new algorithms, namely the algorithm for automated generation of adaptive pattern and the algorithm for computation of reference phase map, is explained in the next section.

5.2.4 Automated Generation of Adaptive Projection Patterns

5.2.4.1 Considerations in the Design of Fringe Patterns

In SMDFP technique, fringe patterns are used to obtain phase maps of objects. The accuracy and coverage of the phase maps directly influence the accuracy and coverage of the final measurement result. The goal of designing (or selecting) a fringe pattern is to achieve optimal accuracy and coverage on the phase map(s) acquired by using the designed fringe pattern for projection.

As mentioned in Section 6.3, many error sources could affect the accuracy of phase maps obtained, e.g. imperfect light projection, fluctuation of environmental light and noise in the image. All these error sources contribute eventually to the pixel

intensities in the images of the object, which are acquired under the projections of phase-shifted fringe pattern. Also, the absolute magnitudes of the image intensity errors caused by these sources do not depend on the specific fringe pattern being used. Given a fixed level of image intensity error, the accuracy of the phase maps obtained can be improved through two approaches. One is to maintain high fringe contrast in the images, and the other is to use small fringe pitch in the projection pattern. For the first measure, a higher fringe contrast makes the relative magnitude of the image intensity error smaller and hence achieves better phase accuracy. For the second measure, a smaller fringe pitch leads to a wider phase range and hence makes the relative phase error smaller. Yet, the improvement of fringe contrast in the images and the decrease of fringe pitch in the projection pattern are limited by a number of factors and also contradictory.

First of all, the smallest fringe pitch that can be used in projection patterns is limited by the resolution of the DMD/LCD projector. Fringe patterns generated by DMD/LCD projectors are pixelated, which can cause considerable errors in the phase maps obtained. To reduce the pixelation effect of the projection patterns, the projector can be purposely defocused such that the projection in the measurement volume is out-of-focus [37]. However, the defocusing of the projector also reduces the fringe contrast of the projection, which would in turn reduces the fringe contrast in the images and leads to larger phase errors. When the fringe pitch is smaller than a certain value, the drop in fringe contrast caused by the defocusing of projector is significant. Based on our experiments with three different DMD projectors, we found that 8 pixels is the practical minimum for the fringe pitch to be used in projection patterns.

Secondly, the finite resolution of CCD camera also limits the possible fringe pitch that can be used in projection patterns. As demonstrated earlier (see Section 5.1.1), when a small fringe pitch is used in the projection pattern, for areas on the object's

surface whose normal directions are within a certain range, the corresponding local fringe pitches in the image(s) would be too small and the phase map construction over those areas would fail. We found that in practice, if the local fringe pitch in the image is smaller than 6 or 7 pixels, the errors in the calculated phase values would increase dramatically and possible failure in the phase map construction would occur. The constraint on the minimum fringe pitch imposed by the resolution of camera can also be explained from the perspective of fringe contrast in the images. A detailed explanation is given in Appendix B.1.

In summary, the smallest fringe pitch that can be used in projection patterns is limited by both the projector and the camera's resolutions. It was found experimentally that, in order to achieve a good accuracy the fringe pitch in the projection pattern needs to be at least 8 pixels and the fringe pitch in the images needs to be at least 6 or 7 pixels. On the other hand, the fringe pitches in both the projection pattern and the images need to be kept at the possible minimum.

5.2.4.2 Algorithm for Generation of Adaptive Patterns

The developed algorithm for automated generation of adaptive projection pattern is as the following:

1. **Acquisition of two absolute phase maps of the object, $\Phi^{(V)}$ and $\Phi^{(H)}$:** Two absolute phase maps of the object being measured are obtained by using vertical and horizontal fixed-pitch fringe patterns respectively. Both fringe patterns are selected with a medium fringe pitch to avoid possible failure in the construction of the phase maps. Let $p_{fc}^{(P)}$ denote the smallest fringe pitch for the vertical fringe pattern that could provide satisfactory fringe contrast in the light projections. The medium fringe pitch selected for the vertical fringe pattern can then be written as $c_{p0} \cdot p_{fc}^{(P)}$, where c_{p0} is a coefficient that is generally

greater than 2.

Let $\Phi_P^{(V)}(\xi,\eta)$ and $\Phi_P^{(H)}(\xi,\eta)$ denote the phase distributions in the vertical and the horizontal fringe patterns respectively. $\Phi_P^{(V)}$ and $\Phi_P^{(H)}$ can be expressed as the following:

$$\begin{cases} \Phi_P^{(V)}(\xi,\eta) = \left[2\pi/(c_{p0}\cdot p_{fc}^{(P)})\right]\cdot\xi \\ \Phi_P^{(H)}(\xi,\eta) = \left(2\pi/p_P^{(H)}\right)\cdot\eta \end{cases} \tag{5.1}$$

where $p_P^{(H)}$ is the fringe pitch used in the horizontal fringe pattern.

Let $\Phi^{(V)}(u,v)$ and $\Phi^{(H)}(u,v)$ denote the phase maps that are obtained by using the vertical and the horizontal fringe patterns. With the two phase maps of the object, each point (u_1,v_1) in the camera's image plane can be associated with two phase values, $\Phi^{(V)}(u_1,v_1)$ and $\Phi^{(H)}(u_1,v_1)$. Accordingly, a point in the projection pattern can be located uniquely which has the same phase value pair. That is, if we denote this corresponding point in the projection pattern as (ξ_1,η_1), the following equation holds:

$$\begin{cases} \Phi_P^{(V)}(\xi_1,\eta_1) = \Phi^{(V)}(u_1,v_1) \\ \Phi_P^{(H)}(\xi_1,\eta_1) = \Phi^{(H)}(u_1,v_1) \end{cases} \tag{5.2}$$

As we can see, by combining Eqn. 5.1 and 5.2, the coordinates (ξ_1,η_1) can be calculated from the following equation:

$$\begin{cases} \xi_1 = \left[c_{p0}\cdot p_{fc}^{(P)}/(2\pi)\right]\cdot\Phi^{(V)}(u_1,v_1) \\ \eta_1 = \left[p_P^{(H)}/(2\pi)\right]\cdot\Phi^{(H)}(u_1,v_1) \end{cases} \tag{5.3}$$

2. **Construction of the gradient field of the adaptive pattern,** $[\nabla\Phi]_{\xi}^{(A)}$:

   Let $\Phi_P^{(A)}(s,t)$ denote the phase distribution of the adaptive projection pattern to be built, which is a 2-D array defined on the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane. (s,t) are indices to

elements in $\Phi_P^{(A)}$. Let $[\nabla\Phi]_\xi^{(A)}(s,t)$ be the gradients of $\Phi_P^{(A)}(s,t)$ along $\boldsymbol{\xi}$-axis, i.e.

$$\begin{aligned}[\nabla\Phi]_\xi^{(A)}(s,t) &= \Phi_P^{(A)}(s,t+1) - \Phi_P^{(A)}(s,t) \\ \text{for } \ s &= 1,\cdots,S \ \text{ and } \ t = 1,\cdots,(T-1)\end{aligned} \tag{5.4}$$

where $s = 1$ to S, $t = 1$ to $(T-1)$, and $S \times T$ are the dimensions of $\Phi_P^{(A)}$. In order to build the adaptive projection pattern, the gradient field $[\nabla\Phi]_\xi^{(A)}$ needs to be constructed first as the following:

- Initialize all elements in $[\nabla\Phi]_\xi^{(A)}$ to $2\pi/p_{fc}^{(P)}$, where $p_{fc}^{(P)}$ is the critical fringe pitch of the projection pattern as defined earlier.
- Let $p_{fc}^{(I)}$ denote the smallest local fringe pitch in the images that could provide satisfactory phase reconstruction accuracy. Let $[\nabla\Phi]_u(i,j)$ denote the gradients of phase map $\Phi^{(V)}(u,v)$ along $\boldsymbol{u}$-axis. Create a bitmap mask, $\boldsymbol{M}^{(I)}$, for $[\nabla\Phi]_u(i,j)$ to mark the pixels whose phase gradients are too large. That is, for all pixels (i,j) in the object's phase map who satisfy

  $$[\nabla\Phi]_u(i,j) > \frac{2\pi}{c_{p0} \cdot p_{fc}^{(I)}} \tag{5.5}$$

  where c_{p0} is the coefficient defined in Eqn. 5.1, mark the corresponding pixels in $\boldsymbol{M}^{(I)}$ as 1s and otherwise 0s. The 1-pixels in $\boldsymbol{M}^{(I)}$ form a number of connected components, which can be isolated and labeled by using the region growing algorithm [77].
- Let $R_M^{(I)}(l)$ $(l = 1,\ldots,L)$ denote the connected components in $\boldsymbol{M}^{(I)}$. Create a bitmap mask, $\boldsymbol{M}^{(A)}$, for $[\nabla\Phi]_\xi^{(A)}$ to mark the pixels whose gradient values need to be adjusted. This can be done using the following procedure: For each region $R_M^{(I)}(l)$ in $\boldsymbol{M}^{(I)}$, find its boundary $B_M^{(I)}(l)$; Map this boundary to $\boldsymbol{M}^{(A)}$ by using Eqn. 5.3; Fill the mapped boundary in $\boldsymbol{M}^{(A)}$ to form

a region $R_M^{(A)}(l)$, which corresponds to the region $R_M^{(I)}(l)$ in $\boldsymbol{M}^{(I)}$.

- Consider a marked region $R_M^{(I)}(l)$ $(l = 1, \ldots, L)$ in $\boldsymbol{M}^{(I)}$. For a pixel (i_1, j_1) in this region, compute its corresponding coordinates in the projection pattern by using Eqn. 5.3. Let (s_1, t_1) denote the calculated coordinates of the pixel in the projection pattern. Set the ξ-gradients of this pixel in the adaptive projection pattern as the following:

$$[\nabla\Phi]_\xi^{(A)}(s_1, t_1) = \frac{4\pi^2}{c_{p0}\, p_{fc}^{(I)}\, p_{fc}^{(P)}\, [\nabla\Phi]_u(i_1, j_1)} \tag{5.6}$$

  Perform this operation for all pixels in the region $R_M^{(I)}(l)$. Depending on the mapping from $R_M^{(I)}(l)$ to $R_M^{(A)}(l)$ (i.e. the size of the region in terms of pixels is expanded or shrunk), there might be missed pixels in $R_M^{(A)}(l)$ whose gradient values were not updated. Hence, a search in $R_M^{(A)}(l)$ for missed pixels needs to be done as the final step and the values of these pixels can be updated by means of interpolation.

  The above process is conducted for all marked regions $R_M^{(I)}(l)$, i.e. all pixels in the object's phase map that meet the criteria described in Eqn. 5.5. The substantial meaning of this process is, if the local fringe pitch in an area in the object's images is too small, the corresponding area in the adaptive projection pattern needs to be set with larger fringe pitch.

- Smooth $[\nabla\Phi]_\xi^{(A)}$ to avoid steep changes in the gradient values.

3. **Construction of the phase distribution of the adaptive pattern, $\Phi_P^{(A)}$:** Once the gradient field $[\nabla\Phi]_\xi^{(A)}$ is constructed, the phase distribution of the adaptive projection pattern, $\Phi_P^{(A)}$, can be calculated from discrete integration

of $[\nabla\Phi]_{\xi}^{(A)}$ as the following:

$$\Phi_P^{(A)}(s,t) = \Phi_P^{(A)}(s,t-1) + [\nabla\Phi]_{\xi}^{(A)}(s,t-1)\ , \quad \text{for } s = 1,\cdots,S \text{ and } t = 2,\cdots,T \tag{5.7}$$

This process requires the initialization of the first column of $\Phi_P^{(A)}$, i.e. $\Phi_P^{(A)}(s,1)$ $(s = 1,\cdots,S)$, which is generally set to zeros.

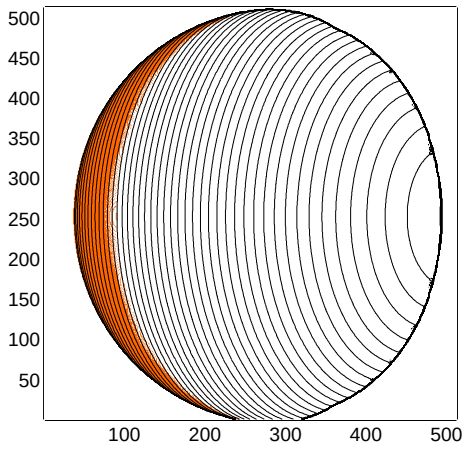
4. **Building the adaptive pattern**:

   The adaptive projection pattern is built from its phase distribution, $\Phi_P^{(A)}$, by applying a sinusoidal modulation. The intensity distribution of the pattern can be expressed using the following equation:

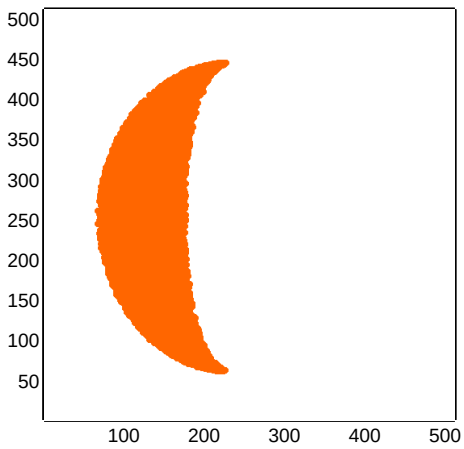
$$I^{(P)}(s,t) = \frac{I_{max}^{(P)}}{2}\left[1 + \sin\left(\Phi_P^{(A)}(s,t)\right)\right] \tag{5.8}$$

   where $I^{(P)}(s,t)$ is the intensity of pixel (s,t) and $I_{max}^{(P)}$ is the maximum intensity in projection pattern.

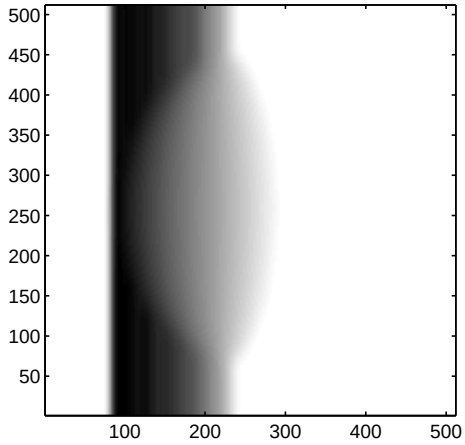
The process of generating an adaptive projection pattern for the measurement of a sphere, is demonstrated in Fig. 5.10. Figure. 5.10(a) shows the phase map of the sphere $\Phi^{(V)}$ (in contours), which is acquired by using the vertical fringe pattern $\Phi_P^{(V)}$. The highlighted region in the phase map indicates pixels whose phase gradients $[\nabla\Phi]_u$ are greater than $2\pi/\left(c_{p0}\cdot p_{fc}^{(I)}\right)$. The corresponding region in the projection pattern, in which the pixels' phase gradients $[\nabla\Phi]_{\xi}^{(A)}$ need to be set smaller than the initial value $2\pi/p_{fc}^{(P)}$, is shown in Fig. 5.10(b). Figure 5.10(c) shows the constructed gradient field of the adaptive pattern, $[\nabla\Phi]_{\xi}$, in a gray-scale image, where darker color indicates smaller gradient value. Figure 5.10(d) shows the constructed phase distribution of the adaptive pattern, $\Phi_P^{(A)}$, in contours. The adaptive projection pattern generated at the end is shown in Fig. 5.10(e).



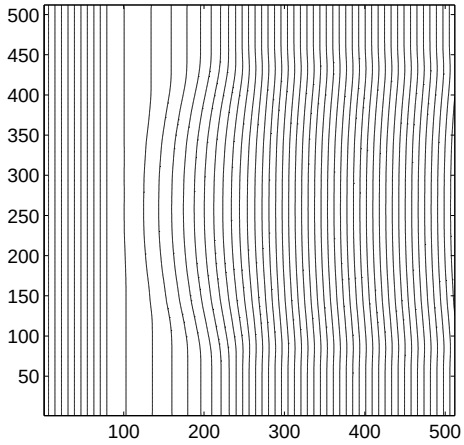
(a) Contours of the sphere's phase map, $\Phi^{(V)}$ (pixels with large phase gradients, $[\nabla\Phi]_u$, are highlighted)



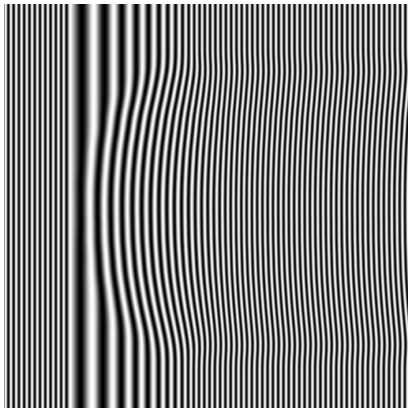
(b) Pixels in the projection pattern whose phase gradients, $[\nabla\Phi]_\xi$, need to be set smaller than their initial values



(c) The constructed gradient field of the adaptive pattern, $[\nabla\Phi]_\xi^{(A)}$ (darker color indicates smaller value)



(d) Contours of the constructed phase distribution of the adaptive pattern, $\Phi_P^{(A)}$



(e) The adaptive projection pattern generated at the end

Figure 5.10: Generation of an adaptive pattern for the measurement of a sphere

A few additional examples of adaptive projection patterns generated for measurements of varied objects can be found in Fig. 5.13(c), 5.15(c) and 5.17(c). The performance of the generated adaptive patterns will be discussed in Section 5.3.

5.2.5 Algorithm for Construction of New Reference Phase Map

As mentioned in Section 5.2.1, once a new adaptive projection pattern has been generated for a particular measurement, a new reference phase map needs to be acquired to support the use of the adaptive pattern in the measurement(s). The acquisition of the new reference phase map can either be done by a set of measurement operations or by pure computation. The measurement approach is to place a flat plane in the measurement volume of the sensor, measure its position and orientation (w.r.t. the sensor) and obtain an absolute phase map of the plane by projecting the newly generated adaptive pattern. As a contrast, the computational approach constructs the new reference phase map through numerical interpolations over the existing regular reference phase map. It does not require any additional operations and hence is the preferred method at most occasions.

This algorithm requires two absolute phase maps of the reference plane as inputs, as well as the generated adaptive pattern. One of the phase maps is the regular reference phase map (denoted by Φ_R at most places in this dissertation), which is obtained by using a vertical fringe pattern. The other phase map is obtained by using a horizontal fringe pattern. Both fringe patterns used are of fixed fringe pitch. Let $\Phi_R^{(V)}(u, v)$ denote the regular reference phase map here, and let $\Phi_P^{(V)}(\xi, \eta)$ be the phase distribution of the corresponding vertical fringe pattern. Similarly, let $\Phi_P^{(H)}(\xi, \eta)$ denote the horizontal fringe pattern and let $\Phi_R^{(H)}(u, v)$ be the resulting

phase map. $\Phi_P^{(V)}(\xi,\eta)$ and $\Phi_P^{(H)}(\xi,\eta)$ can be expressed using the following equation:

$$\begin{cases} \Phi_P^{(V)}(\xi,\eta) = c_V \cdot \xi \\ \Phi_P^{(H)}(\xi,\eta) = c_H \cdot \eta \end{cases} \tag{5.9}$$

where c_V and c_H are constants.

Let $\Phi_P^{(A)}(\xi,\eta)$ denote the phase distribution of the generated adaptive projection pattern, and let $\Phi_R^{(A)}(u,v)$ denote the corresponding reference phase map to be constructed. When the sensor and the surface to be measured is fixed, each point (u,v) in the phase map has a unique corresponding point in the projection pattern, (ξ,η), which always has the same phase value as the point (u,v), no matter which projection pattern has been used. Therefore, in order to construct the phase map $\Phi_R^{(A)}$ from the known projection pattern $\Phi_P^{(A)}$, one needs to resolve the mapping from space (ξ,η) to space (u,v) for the reference plane. This can be done by using the prepared phase maps of the reference plane, $\Phi_R^{(V)}$ and $\Phi_R^{(H)}$, and the associated fringe patterns.

The procedure to compute the new reference phase map, $\Phi_R^{(A)}$, is as the following:

1. For each pixel (u_1,v_1) in $\Phi_R^{(A)}(u,v)$, compute its corresponding point, (ξ_1,η_1), in the projection pattern by using the following equation:

$$\begin{cases} \xi_1 = (1/c_V) \cdot \Phi_R^{(V)}(u_1,v_1) \\ \eta_1 = (1/c_H) \cdot \Phi_R^{(H)}(u_1,v_1) \end{cases} \tag{5.10}$$

   where c_V and c_H are constants defined in Eqn. 5.9.

2. Using the calculated coordinates (ξ_1,η_1), compute the phase value $\Phi_P^{(A)}(\xi_1,\eta_1)$ by interpolation over the phase distribution of the generated adaptive pattern, $\Phi_P^{(A)}(\xi,\eta)$. The reason for involving interpolation is that, $\Phi_P^{(A)}(\xi,\eta)$ is a 2-D array on the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane and has phase values only on a finite number of grid

points.

3. The calculated phase value $\Phi_P^{(A)}(\xi_1, \eta_1)$ is then assigned to $\Phi_R^{(A)}(u_1, v_1)$.

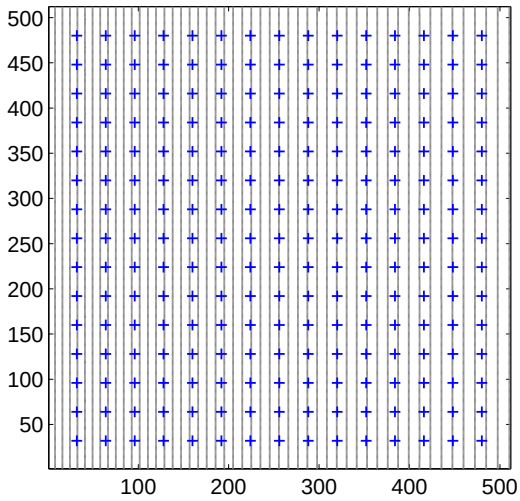
An example of constructing the new reference phase map for an adaptive projection pattern is shown in Fig. 5.11. Figure 5.11(a) and 5.11(b) are the contour plots of the phase maps $\Phi_R^{(V)}$ and $\Phi_R^{(H)}$, respectively. The blue crosses drawn in the figures are example pixels in the $\boldsymbol{u}$-$\boldsymbol{v}$ space, i.e. pixels (u_1, v_1) as referred in the description of the algorithm above. Figure 5.11(c) shows the phase distribution of the adaptive pattern, $\Phi_P^{(A)}$, in contours. The red crosses drawn in the figure are points in the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ space that correspond to the example pixels. Figure 5.11(d) shows the contour plot of the new reference phase map that is constructed, $\Phi_R^{(A)}$.

5.3 Implementation and Results

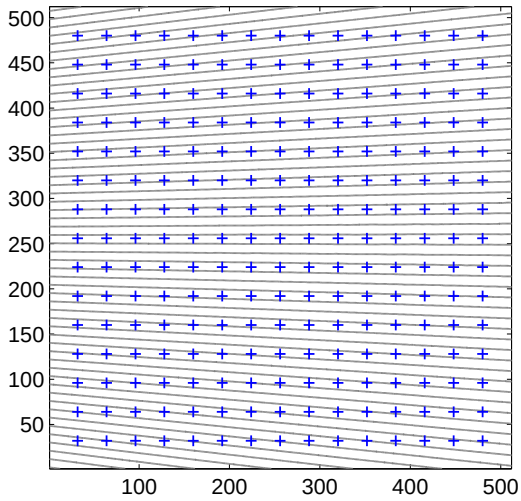
5.3.1 Implementation

The proposed algorithms for generating adaptive projection patterns and for constructing new reference phase map were implemented in Matlab. By using common data files as interface, the developed Matlab program is able to work with the point cloud construction software as described in Section 3.4.1. By doing this, the SMDFP system we have developed is able to conduct shape measurements using adaptive projection patterns.

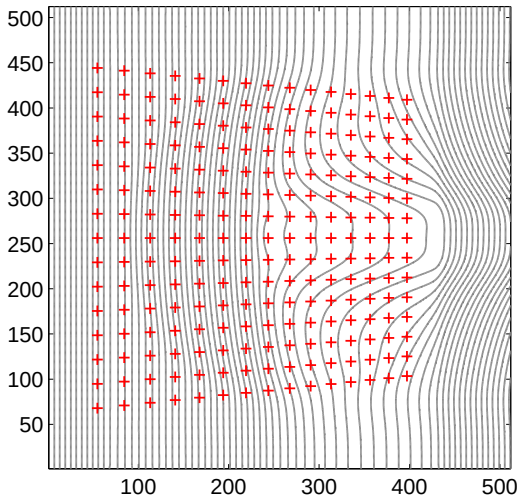
In order to verify the soundness of the adaptive pattern idea as well as the performance of the algorithms developed, a number of tests were conducted in the manner of both simulated measurements and physical measurements. The accuracy and coverage of measurements achieved with the use of adaptive patterns were compared against the performance benchmarks achieved by fixed-pitch fringe patterns.



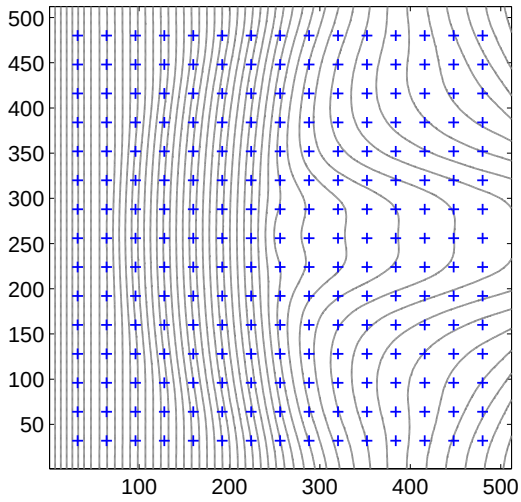
(a) Contours of the regular reference phase map, $\Phi_R^{(V)}$ (blue crosses are example pixels in the (u, v) space)



(b) Contours of the phase map $\Phi_R^{(H)}$ (example pixels drawn in blue crosses)



(c) Contours of the phase distribution of the adaptive pattern, $\Phi_P^{(A)}$ (red crosses are points in the (ξ, η) space that correspond to the example pixels shown in (a), (b) and (d))



(d) Contours of the new reference phase map computed, $\Phi_R^{(A)}$ (example pixels drawn in blue crosses)

Figure 5.11: Construction of the new reference phase map for an adaptive pattern

5.3.2 Performance Tests Using Measurements on CAD Models

By using the developed simulator for SMDFP systems (see Chapter 4 for details), a number of simulated measurements were carried out with objects of varied shapes. The measurement setup is shown in Fig. 5.12. The object being measured is placed in the world coordinate frame, $X^{(W)}Y^{(W)}Z^{(W)}$, with its center position close to the origin. The measurement sensor (i.e. the projector and the camera) is place above the object (along $Z^{(W)}$-axis) with its measurement volume covering the entire object or the area of interest. For all simulation runs, the same set of sensor parameters is used, and the position of the sensor w.r.t. the world coordinate frame remains fixed. In other words, only the projection pattern and the object being measured were changed.

Sensor parameters used in the simulations are as the following: The optical axis of the projector and the optical axis of the camera intersect at point $(0, 0, 0)$, and the angle between them, θ_{PC}, is 35^0; The z-coordinate of the projector's projection center, as well as the z-coordinate of the camera's projection center, is $476.544\,mm$; The field-of-view angles of the projector are 28.072^0 and the field-of-view angles of the camera are 21.239^0, in both horizontal and vertical directions; The projector has a resolution of 1024×1024 pixels and the camera has a resolution of 512×512 pixels; Both of them have a gray-depth of 8 bits.

Four parts were used in the simulated measurements: a part with a sawtooth profile (shown in Fig. 5.13(a)), a block with a cone-shaped hole (shown in Fig. 5.15(a)), a randomly generated spline surface (shown in Fig. 5.17(a)) and a spherical part (shown in Fig. 5.12). The X-Y dimensions of the parts are around $200\,mm \times 200\,mm$ or bigger. The Z-dimensions of the parts are as the following: The part with sawtooth profile, $27\,mm$; the cone-shaped hole, $160\,mm$; the spline surface, $55.554\,mm$; and the

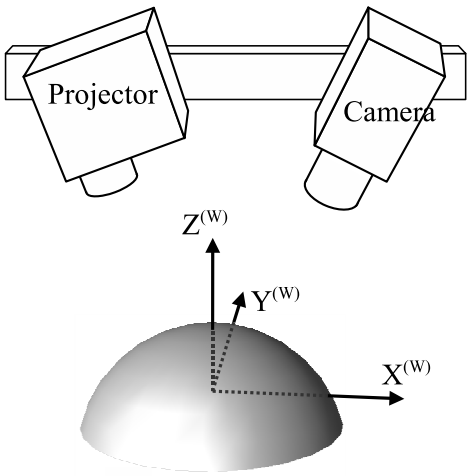


Figure 5.12: Schematic diagram of the measurement setup used in simulations

spherical part, 70 mm. All parts have diffuse surfaces.

To simulate the influence of error sources, random noise was added to the images of objects which were obtained under the projection of fringe patterns. For all simulation runs, the random noise added was generated with a uniform distribution and a magnitude of 2.5% (plus and minus) of the maximum image intensity.

The automatically generated adaptive projection patterns for the measurements of different parts are shown in Fig. 5.13(c), 5.15(c), 5.17(c) and 5.6(c). The images of the parts obtained under the projection of adaptive patterns are shown in Fig. 5.13(d), 5.15(d), 5.17(d) and 5.6(d). For comparison, images obtained under the projection of a fixed-pitch fringe pattern ($n_F = 100$) are also presented (Fig. 5.13(b), 5.15(b), 5.17(b) and 5.6(b)). As can be seen from the images, when the fixed-pitch fringe pattern ($n_F = 100$) was used, the corresponding images of the parts all have certain degree of fringe overcrowding in some part of the image. As a result, the construction of point clouds failed at these regions. Take the measurement of the cone-shaped hole as an example. As can be seen from Fig. 5.15(b), there is a severe fringe overcrowding in the region that corresponds to the center-right part of the cone. In the constructed

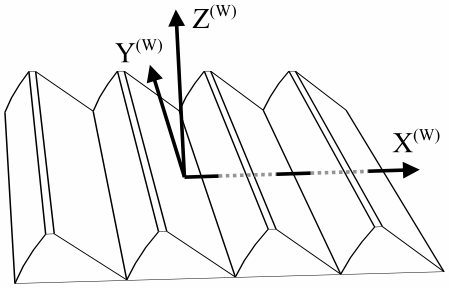
point cloud as shown in Fig.5.16(a), that part of the cone is missing. The same problem was observed in the measurements of other parts as well. However, when adaptive projection patterns were used, there was no such problem.

The measurement performance achieved by using adaptive and fixed-pitch fringe patterns are shown in Fig. 5.14(c), 5.16(c), 5.18(c) and 5.7. The horizontal axes of the plots represent the number of pixels that are unable to be resolved in a shape measurement. As mentioned earlier, the total number of pixels is $512 \times 512 = 262,144$. The vertical axes represent the RMS value of measurement error. n_F is the number of fringes in the fixed-pitch fringe patterns. As can be seen from the plots, when fixed-pitch fringe patterns (with varied fringe numbers) were used, any single measurement was unable to achieve good accuracy and good coverage at the same time. A high fringe number (e.g. 121) is able to achieve very good measurement accuracy but the surface area it fails to measure is quite large; On the opposite, a low fringe number (e.g. 49) has excellent measurement coverage but the accuracy of measurement is fairly poor. With the use of adaptive patterns, this compromise issue can be solved satisfactorily. In all measurements conducted, the adaptive patterns achieved an accuracy that is comparable to a fixed-pitch fringe pattern with a fringe number of around 100 to 121, while at the same time achieved a measurement coverage that is comparable to (a fixed-pitch fringe pattern with) a fringe number of around 49 to 64.

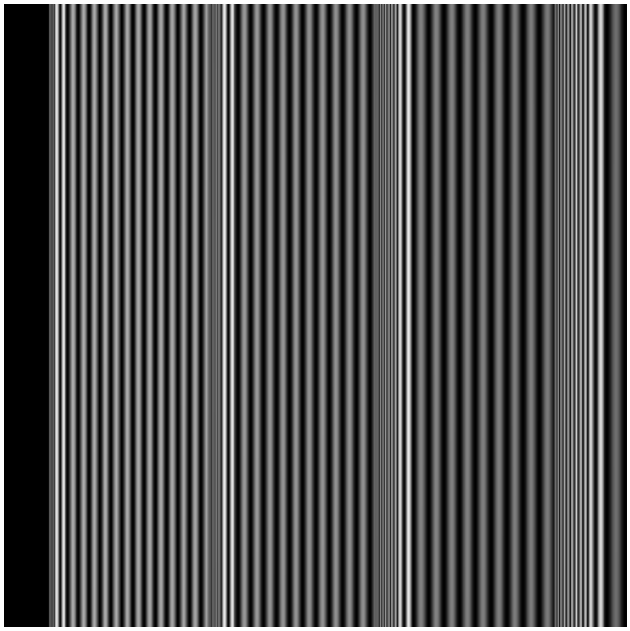
In the measurements of the cone-shaped hole by using fixed-pitch fringe patterns, the part was place at a few slightly different tilt angles around $Y^{(W)}$-axis. Figure 5.16(c) shows the results of the part at two different tilt angles, $\theta_{Part} = 0^0$ and $\theta_{Part} = 7^0$, where $\theta_{Part} = 0^0$ corresponds to the orientation that the part's top surface is parallel to the $X^{(W)} - Y^{(W)}$ plane. The measurement results indicate that, different tilt angles of the part yield different measurement accuracies and coverages for the same fringe pattern. However, it only changes the favor of the measurement performance between accuracy and coverage but makes no improvement to the mea-

surement performance as a whole.

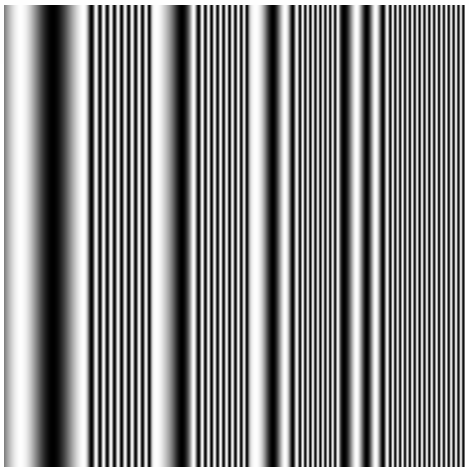
The proposed algorithm for automated generation of adaptive patterns has a few parameters that can be adjusted, e.g. $p_{fc}^{(P)}$ and $p_{fc}^{(I)}$ (see Section 5.2.4.2 for a detailed description of the algorithm and associated parameters). The values of the parameters could affect the exact pattern generated in the end. Figure 5.18(c) shows the measurement performance of two difference adaptive patterns that were generated for the measurement of the spline surface. Both adaptive patterns achieved better overall performance than fixed-pitch fringe patterns.



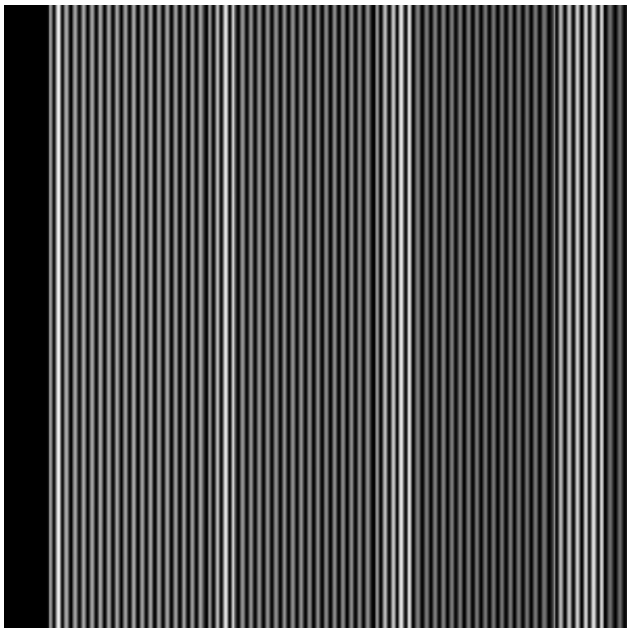
(a) Part with a sawtooth profile



(b) One of the images acquired by using fixed-pitch fringe pattern, $n_F = 100$

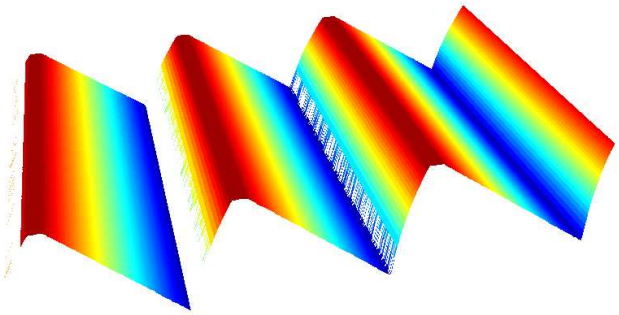


(c) Adaptive fringe pattern generated (shown in larger fringe pitch than actual)

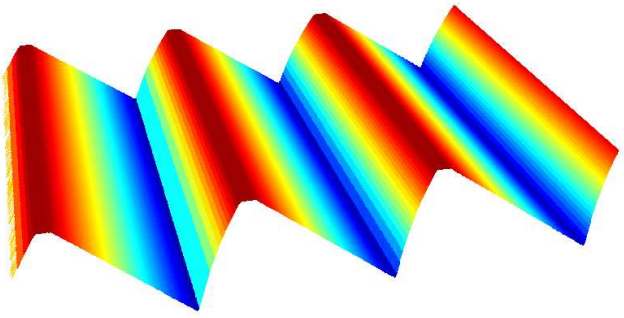


(d) One of the images acquired by using the adaptive fringe pattern as shown in (c)

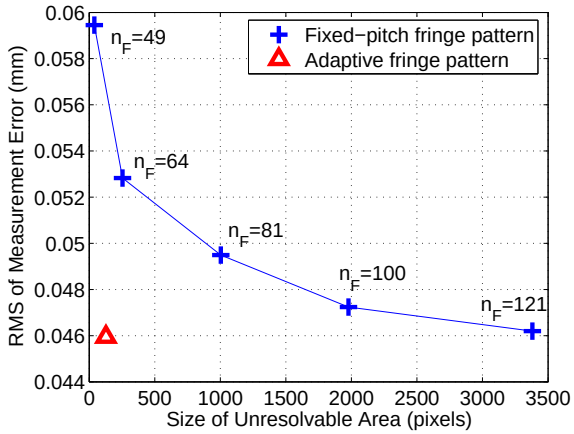
Figure 5.13: Simulated measurements of a part with a sawtooth profile (1)



(a) Point cloud obtained by using fixed-pitch fringe pattern, $n_F = 100$ (pseudo-color represents z-coordinate)

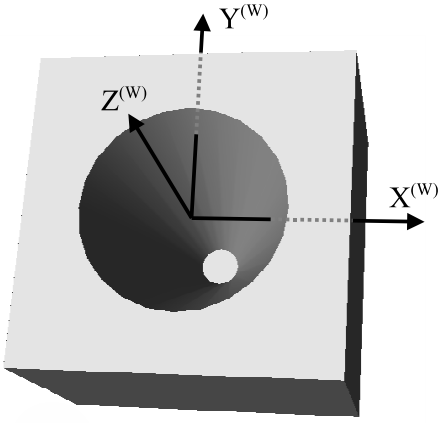


(b) Point cloud obtained by using the adaptive fringe pattern (pseudo-color represents z-coordinate)

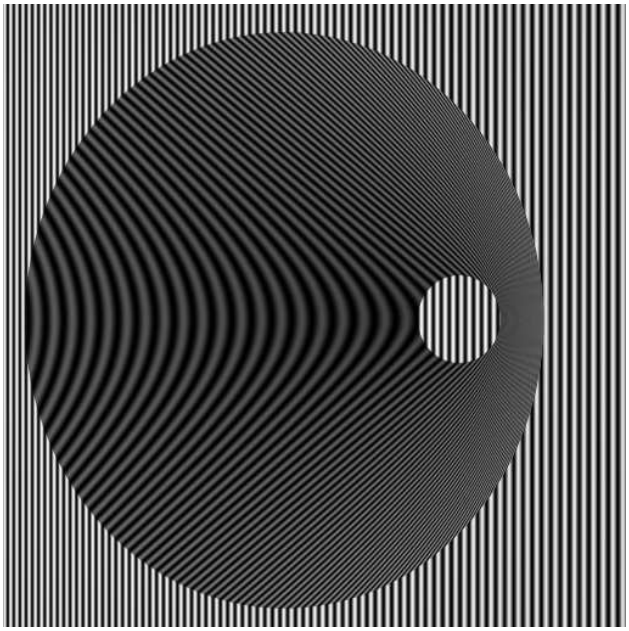


(c) Measurement performance: adaptive fringe pattern vs. fixed-pitch fringe patterns (n_F is the fringe number)

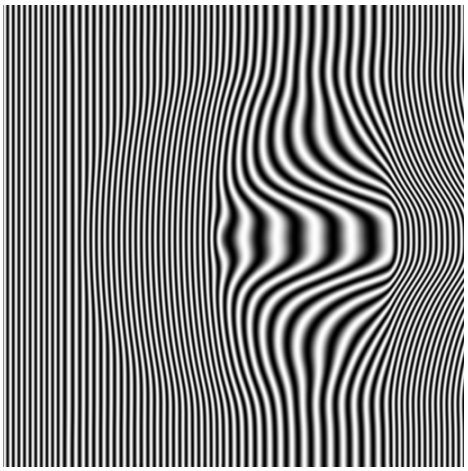
Figure 5.14: Simulated measurements of a part with a sawtooth profile (2)



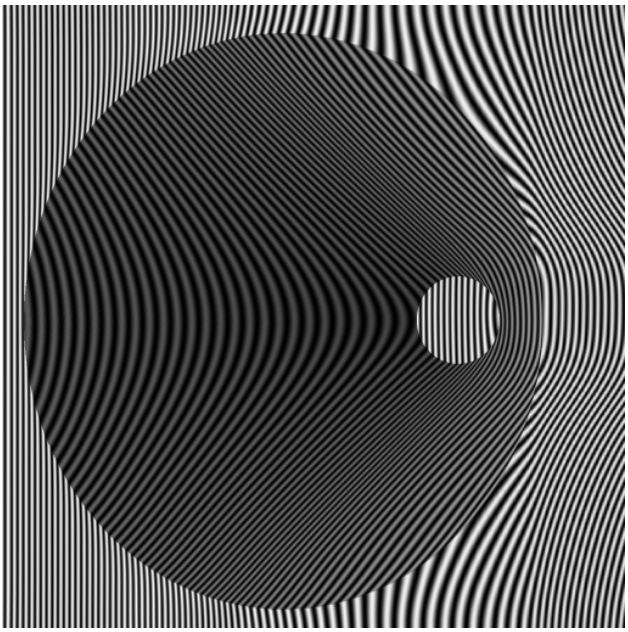
(a) Part with a cone-shaped hole



(b) One of the images acquired by using fixed-pitch fringe pattern, $n_F = 100$

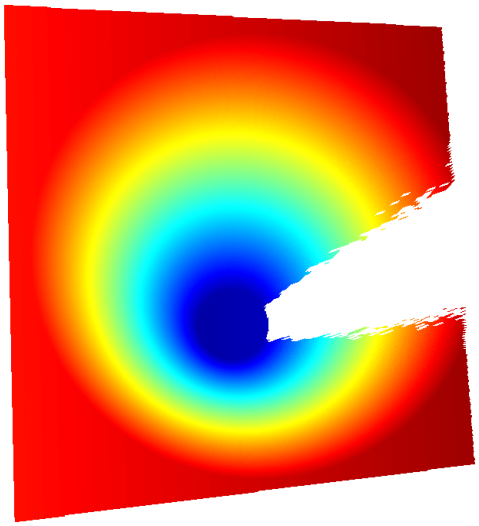


(c) Adaptive fringe pattern generated (shown in larger fringe pitch than actual)

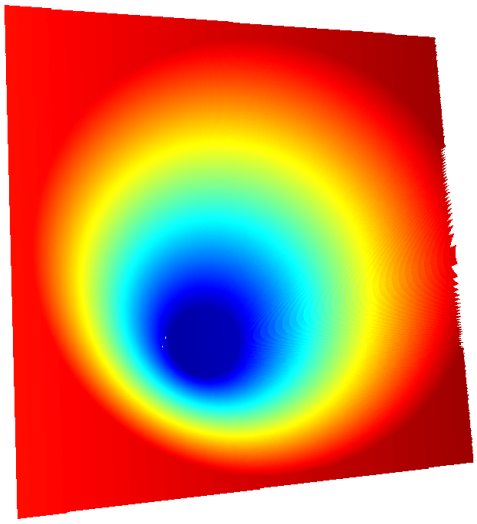


(d) One of the images acquired by using the adaptive fringe pattern as shown in (c)

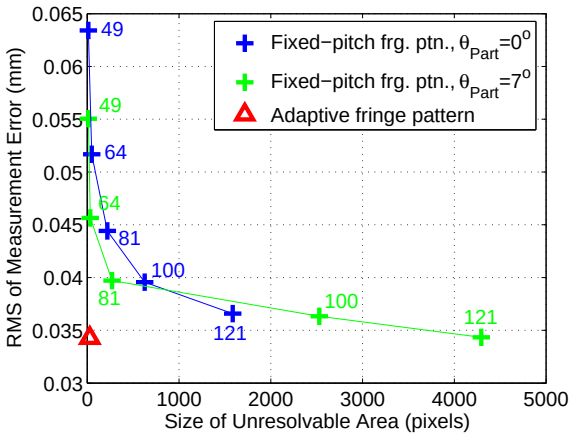
Figure 5.15: Simulated measurements of a cone-shaped hole (1)



(a) Point cloud obtained by using fixed-pitch fringe pattern, $n_F = 100$ (pseudo-color represents z-coordinate)

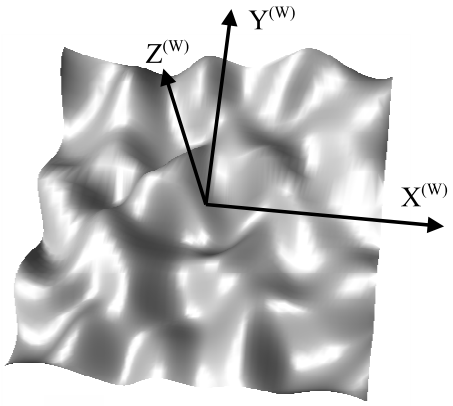


(b) Point cloud obtained by using the adaptive fringe pattern (pseudo-color represents z-coordinate)

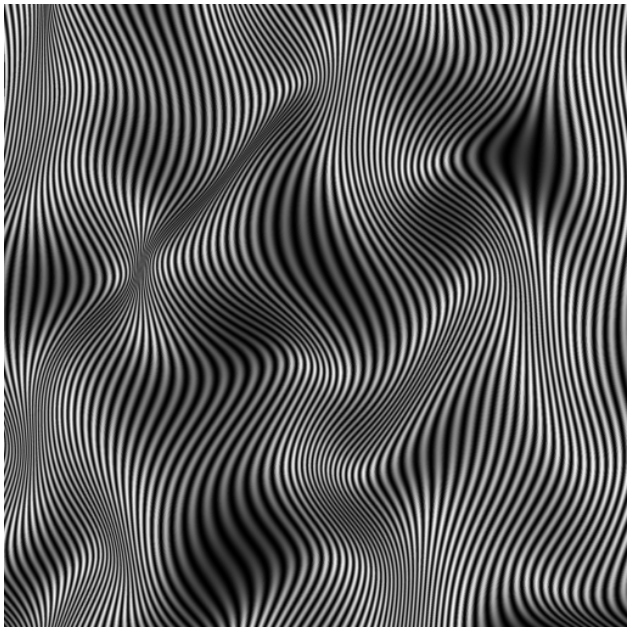


(c) Measurement performance: adaptive fringe pattern vs. fixed-pitch fringe patterns

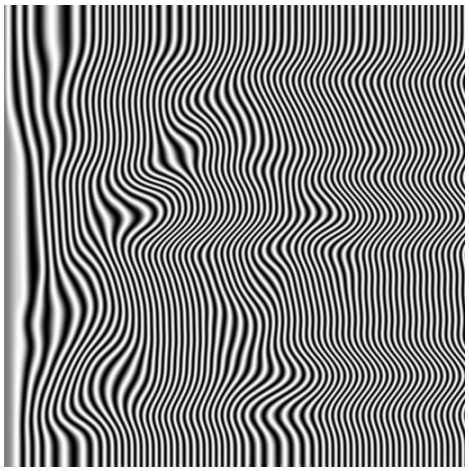
Figure 5.16: Simulated measurements of a cone-shaped hole (2)



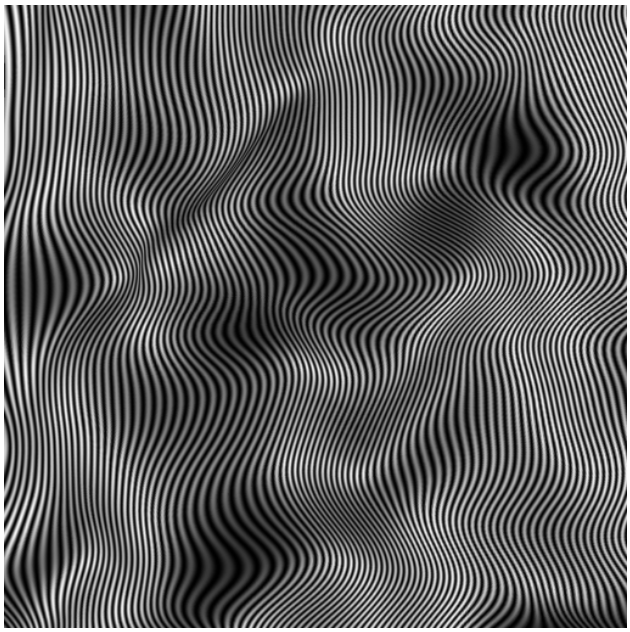
(a) Randomly generated spline surface



(b) One of the images acquired by using fixed-pitch fringe pattern, $n_F = 100$

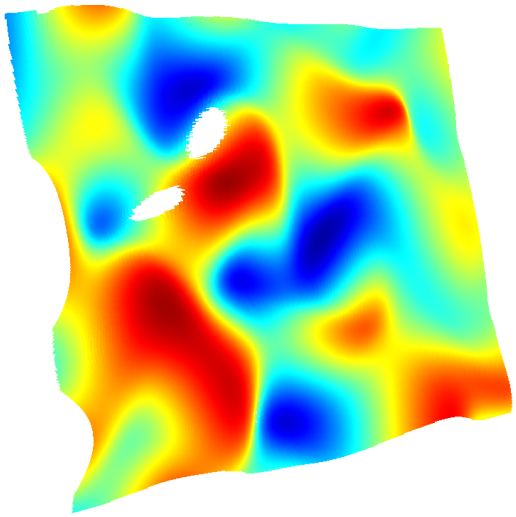


(c) Adaptive fringe pattern generated (shown in larger fringe pitch than actual)

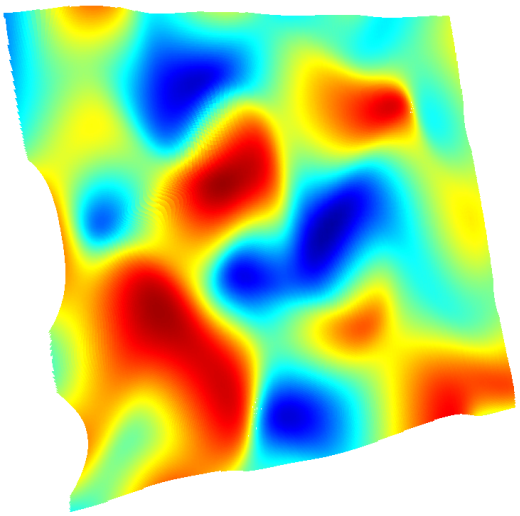


(d) One of the images obtained by using the adaptive fringe pattern as shown in (c)

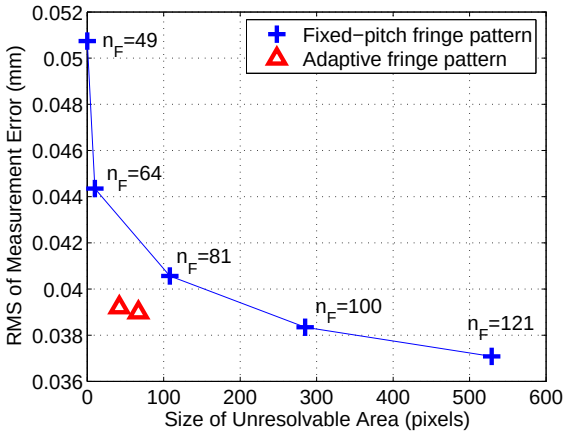
Figure 5.17: Simulated measurements of a spline surface (1)



(a) Point cloud obtained by using fixed-pitch fringe pattern, $n_F = 100$ (pseudo-color represents z-coordinate)



(b) Point cloud obtained by using the adaptive fringe pattern (pseudo-color represents z-coordinate)



(c) Measurement performance: adaptive fringe pattern vs. fixed-pitch fringe patterns (n_F is the fringe number)

Figure 5.18: Simulated measurements of a spline surface (2)

5.3.3 Performance Tests Using Measurements on Physical Parts

The performance test of adaptive projection patterns was also conducted on physical parts. Two artifacts were used in the test, one is a plastic flowerpot (see Fig. 5.19(a)) and the other is a plastic tube (see Fig. 5.20(a)). For the measurements of each artifact, both adaptive projection pattern and fixed-pitch fringe patterns were used, while keeping the other measurement settings fixed. The performances of different projection patterns, in terms of measurement coverage and measurement accuracy, were analyzed.

The measurements of the two artifacts were performed using the SMDFP sensor we have built, as described in Section 3.4.1. The major parameters of the sensor are as the following: The angle between the optical axis of the projector and the optical axis of the camera is 27^0; The distance between the sensor and the center of the measurement volume is around $600\,mm$; The projector's field-of-view angles are 43.6^0 (Horizontal) and 33.4^0 (Vertical); The camera's field-of-view angles are 22.28^0 (Horizontal) and 16.77^0 (Vertical); The resolution of the projector is 1024×768 pixels; The resolution of the camera is 640×480 pixels; Both the projector and the camera have a gray-depth of 8 bits. All measurements were conducted under regular indoor lighting condition.

The plastic flowerpot used in the test has a maximum diameter of $198\,mm$ and a depth of $149\,mm$. Figure 5.19(b) shows an image of the pot acquired by using a fixed-pitch fringe pattern with 100 fringes ($n_F = 100$). In the area that corresponds to the right inner wall of the pot, the fringes are overly crowded and hardly distinguishable. As a result, that part of the surface is unresolvable in the measurement. However, when the adaptive projection pattern was used (as shown in Fig. 5.19(c)[2]), the prob-

[2]The adaptive projection pattern is shown with uniformly increased fringe pitch, for the purpose of clear display

lem of fringe overcrowding is resolved (see Fig. 5.19(d)). This improvement can also be observed from Fig. 5.19(e), which shows the measurement coverages achieved by using adaptive projection pattern and fixed-pitch fringe patterns respectively. The horizontal axis of the plot represents the number of fringes in the projection pattern (n_F), and the vertical axis represents the number of pixels that were unable to be resolved in the measurement. For fixed-pitch fringe patterns, the size of unresolvable area increases as the fringe number increases. This is caused by fringe overcrowding as the fringe pitch in the projection pattern gets smaller. In the case of adaptive pattern, a measurement coverage comparable to a (fixed-pitch fringe pattern with a) fringe number of 50 was achieved, although the adaptive pattern has an equivalent fringe number of 118.

As a summary of the above, when the number of fringes in the projection pattern is fixed, e.g. in order to maintain a certain measurement accuracy, adaptive projection pattern is able to achieve better measurement coverage than fixed-pitch fringe pattern. Because the accurate shape of the flowerpot is not known, the study on measurement accuracy was skipped.

The plastic tube used in the test has a diameter of $127.34\,mm$ and a height of $95\,mm$. The adaptive pattern generated for the measurement of the tube is shown in Fig. 5.20(c). Images of the tube under the projection of a fixed-pitch fringe pattern ($n_F = 125$) and the adaptive pattern are presented in Fig. 5.20(b) and 5.20(d) respectively. The problem of fringe overcrowding was discovered at the left edge area of the tube when fixed-pitch fringe patterns with large fringe numbers (greater than 81) were used. This problem did not occur in the case of adaptive pattern. In order to evaluate the measurement accuracy, the point cloud acquired in a measurement was fitted to a cylinder and the residual deviation was analyzed. Figure 5.20(e) shows the measurement coverage and measurement accuracy achieved by using different projection patterns. The horizontal axis of the plot represents the number of pixels that

were unable to be resolved in the measurement, and the vertical axis represents the RMS value of the divergence of the point cloud from a perfect cylinder. As can be seen from the plot, the adaptive projection pattern achieved a better overall measurement performance than fixed-pitch fringe patterns.

5.4 Summary

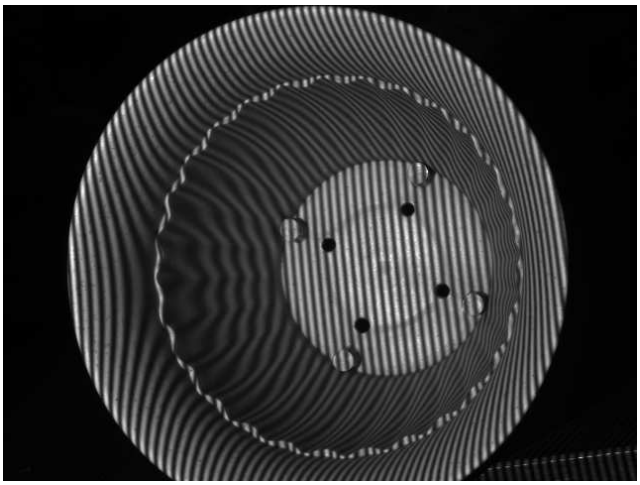
Most existing SMDFP systems use straight fringe patterns with fixed fringe pitch for projections. When measuring objects with a large range of surface normal directions, with the use of fixed-pitch fringe patterns it is hard to achieve full measurement coverage and good measurement accuracy at the same time without increasing the number of patterns required. The idea of adaptive projection pattern was proposed to solve this problem.

A detailed description of using adaptive projection patterns in SMDFP is given in this chapter, including the measurement procedure, potential applications and the requirements on system model and algorithms. A number of tests, including measurements on CAD models and measurements on physical parts, were conducted to verify the soundness and effectiveness of the adaptive pattern idea as well as the algorithms developed to enable this capability.

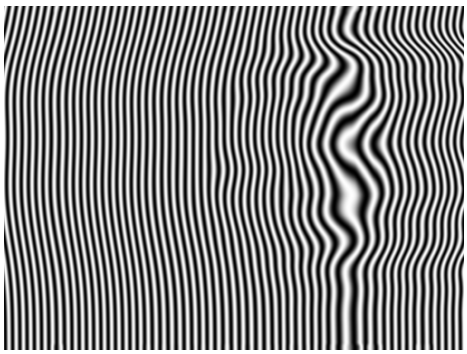
The results of the tests show, adaptive projection patterns provide better overall measurement performance (coverage and accuracy) than fixed-pitch fringe patterns, especially when the object being measured has a large range of surface normal directions. For applications such as on-line parts inspection, the overhead of using adaptive patterns is negligible. Hence, adaptive pattern is an ideal solution for such applications when measurement speed and accuracy are crucial.

The developed framework of using adaptive patterns integrates seamlessly with the mathematical system model and algorithms proposed in Chapter 3. The two

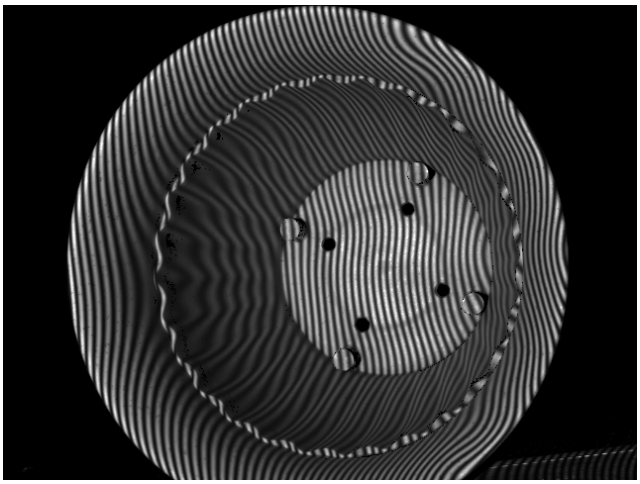
(a) Photograph of the flowerpot



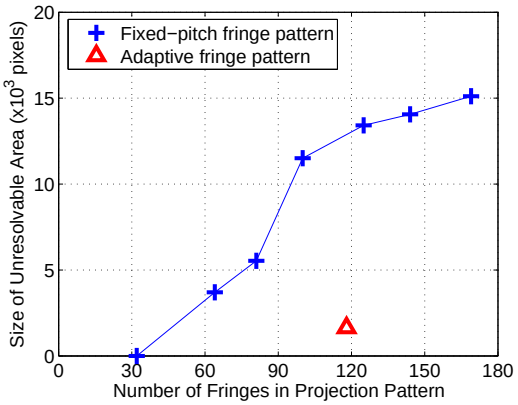
(b) Image acquired using fixed-pitch fringe pattern, $n_F = 100$



(c) The adaptive fringe pattern generated



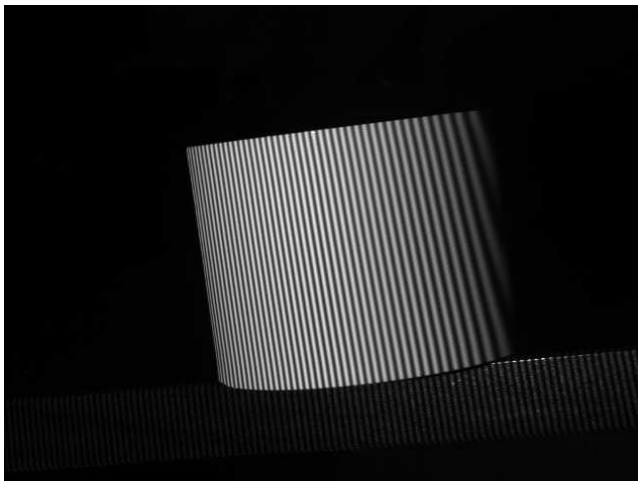
(d) Image acquired using the adaptive fringe pattern



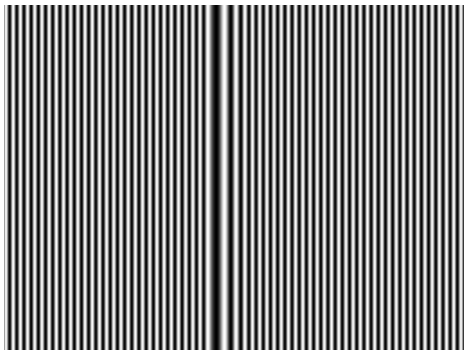
(e) Measurement coverage: adaptive fringe pattern vs. fixed-pitch fringe patterns

Figure 5.19: Measurements of a plastic flowerpot

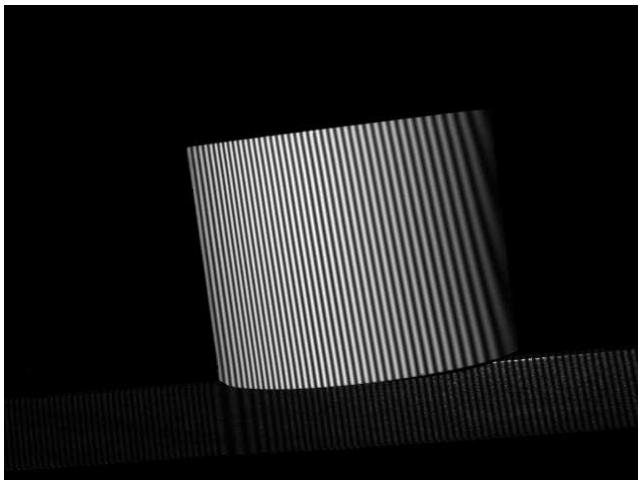
(a) Photograph of the plastic tube



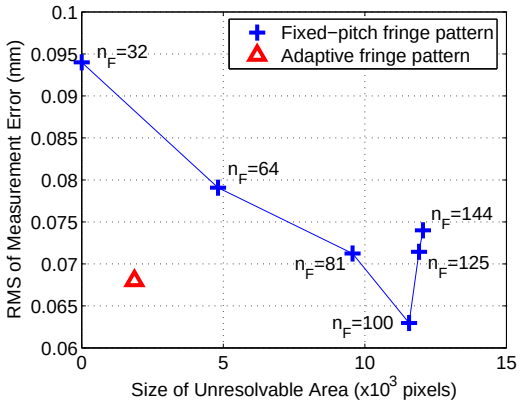
(b) Image acquired using fixed-pitch fringe pattern, $n_F = 125$



(c) The adaptive fringe pattern



(d) Image acquired using the adaptive fringe pattern



(e) Measurement performance: adaptive fringe pattern vs. fixed-pitch fringe patterns

Figure 5.20: Measurements of a plastic tube

algorithms presented in this chapter, i.e. the algorithm for automated generation of adaptive patterns and the algorithm for computation of new reference phase map, worked very well to support the use of adaptive patterns.

Chapter 6

Estimation of Measurement Uncertainties

This chapter describes the model and algorithms proposed for the estimation of uncertainties in shape measurement process conducted using SMDFP. Section 6.1 gives a brief description of the related work on the estimation of measurement uncertainties, the existing problems and the desired improvements. Section 6.2 gives an overview of the framework we have developed for estimation of measurement uncertainties. Section 6.3 explains the error sources considered in the model. Section 6.4 presents the definitions of uncertainty components as well as the algorithms for estimation of their values. Section 6.5 describes the algorithm for computing the uncertainties in the point clouds. Section 6.6 presents the results of an experiment conducted for verification of the model and algorithms. A summary of this chapter is given in Section 6.7.

6.1 Introduction

It is a desired feature for metrology devices to give the uncertainty of measurements along with measurement results. For SMDFP systems, this means giving the position uncertainties of the points in the point cloud created in a measurement.

In most existing commercial SMFP/SMDFP systems, measurement uncertainty is given as a fixed specification of the system, which is generally acquired by measuring gauge parts and analyzing the deviations of measurements w.r.t. the gauge parts. Unfortunately, the measurement uncertainty of a SMFP/SMDFP system varies across the measurement volume and can be influenced notably by many factors, e.g. the environmental lighting condition, the shape of the object being measured, etc. Therefore, a fixed rating of the measurement uncertainty may not correctly reflect the actual error magnitude of a measurement.

Another approach to estimate measurement uncertainty is to start from evaluating the error level in the acquired phase map of an object. Theories on the evaluation of phase errors caused by varied sources, such as image noise, intensity quantization, non-sinusoidal waveform of the fringe pattern and the phase-shifting method used, have been well developed [78, 38, 39, 34, 35]. However, the modeling of the relation between phase uncertainty and the uncertainty in shape measurement is much less studied. In many research literature, it is assumed that the position uncertainty of the measured points is proportional to the phase uncertainty. This is not true in general for the following reasons: Firstly, due to perspective projection and the angle between the projector and the camera, the ratio between phase error and the error it causes on the position of a measured point is not a constant throughout the measurement volume. This is particularly true when the camera-to-projector angle and/or the field-of-view of the projector and the camera are large. Secondly, there are other sources besides phase errors that contribute to the position errors in the

measured points, e.g. errors in the sensor parameters. Depending on the system configuration and the accuracy of calibration, these error components may be similar in magnitude as the phase errors, or even larger, and hence cannot be neglected.

Due to the complex influence of various sources on the shape measurement uncertainties, it is necessary to have a comprehensive model to describe the relations between the many error sources and the eventual measurement uncertainties caused by them. This model should be able to estimate the measurement uncertainties based on the image data acquired in a measurement as well as the apriori knowledge on all major error sources. However, currently a model consistent with the computational framework developed in this dissertation does not exist. In this chapter, a new model will be developed to meet the requirements discussed above.

6.2 Overview of Approach

The uncertainty model is defined based on the mathematical system model and the related point cloud construction algorithm described in Chapter 3. A schematic diagram of the infrastructure of the uncertainty model is shown in Fig. 6.1. Firstly, a number of error sources that contribute to shape measurement uncertainty are identified from the system's perspective. Examples of the error sources include: inaccurate system calibration, imperfect light projection, fluctuation in environmental light, and image noise. However, direct modeling of the error sources and the estimation of their magnitudes are difficult. To avoid this problem, a number of uncertainty components are defined based on the mathematical system model. These uncertainty components are reflections of the error sources in the system model and they are relatively easy to evaluate and utilize. The components defined include, uncertainties in the sensor parameters, uncertainties in the reference phase map and uncertainties in the object's phase map. One thing worth mentioning is, the mapping from the identi-

fied error sources to the uncertainty components defined does not have a one-to-one correspondence.

The magnitudes of the uncertainty components are estimated using algorithms. The uncertainties in sensor parameters are estimated from calibration data, which include geometric data and images of a calibration artifact. The outcome of the estimation is the uncertainties in each individual sensor parameter. The uncertainties in a phase map are estimated from the raw images that were used to compute the phase map, i.e. images acquired from phase-shifting. The outcome of the estimation is the uncertainty of each phase value in the phase map. The estimation processes need to be performed as part of a system calibration or for every measurement. In Fig. 6.1, the processes to be completed during system calibration are enclosed in the dashed box.

Once the magnitudes of all uncertainty components were estimated, their influences to the uncertainties of shape measurement are evaluated and combined. As the final result, for each point in the point cloud (acquired from a measurement), a 3-D uncertainty region is given, which indicates the possible range of the true position of that point.

6.3 Error Sources Considered

Error sources are the disagreements between a real SMDFP system and the virtual system defined by the mathematical model and the associated parameters. They are either neglected in the model, imprecisely represented or indefinite in quantity (e.g. random noise). The error sources that are considered in the proposed uncertainty model are enumerated in the following list with a short description given for each.

- **Error in the system model**: The proposed mathematical model for SMDFP systems uses the pinhole camera model with lens distortion model for describing

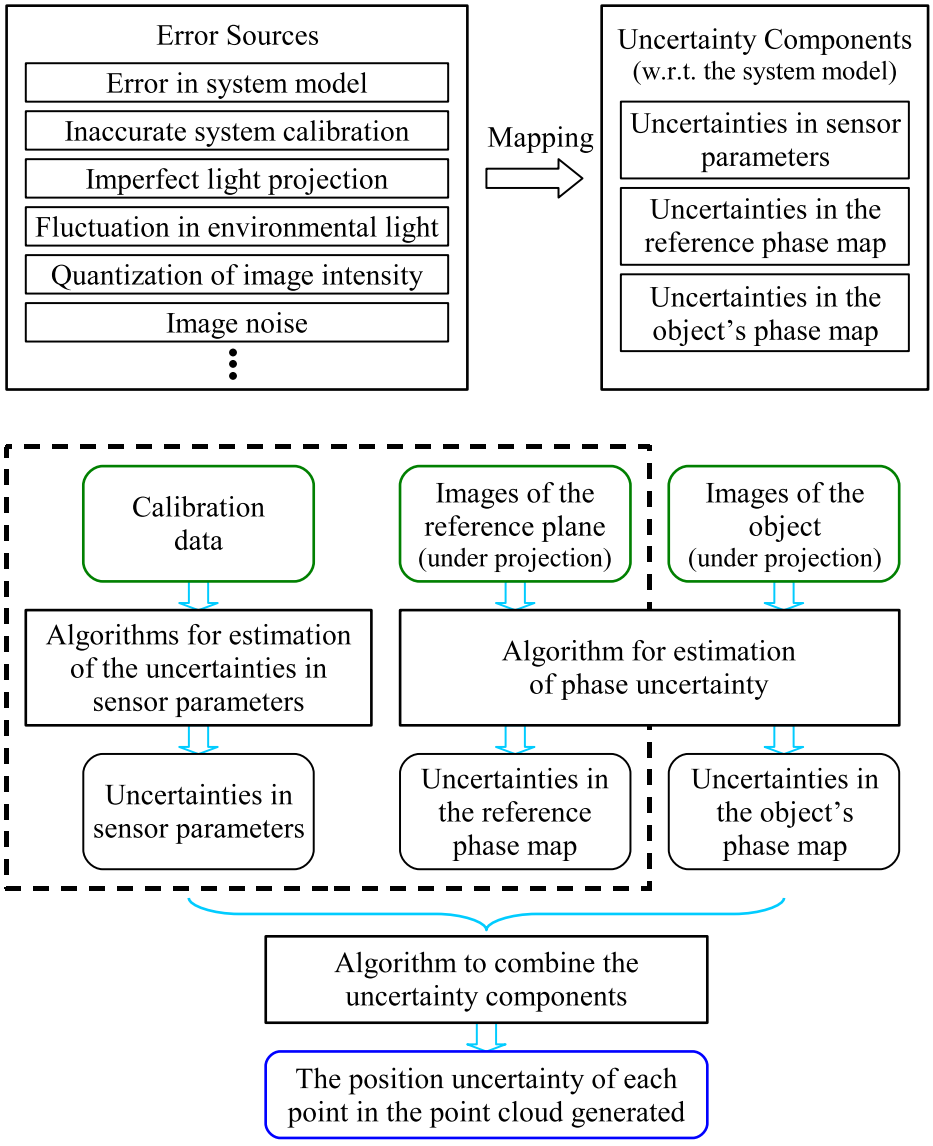


Figure 6.1: Schematic diagram of the infrastructure of the uncertainty model

the camera and the projector. Although the disagreement between this model and the physical laws of the camera and projector is very small, it does exist and hence will introduce errors in shape measurements.

- **Inaccurate system calibration**: The sensor parameters in the system model, which include parameters for describing the camera, projector and the reference plane, are obtained from calibrations. Depending on the design of calibration procedures and the accuracy of calibration artifacts, the estimated values of these parameters may diverge from their real values.

- **Imperfect light projection**: SMDFP technique assumes a perfect projector which can generate light projections precisely as designated. Real computer projectors have limited spatial resolution (e.g. 1024×768 pixels) and color depth (e.g. 8-bit gray-scale), and hence can generate digitized light projections only. Additionally, the light modulation unit of projectors, DMD or LCD, introduces noise as well.

- **Fluctuation in environmental light**: SMDFP technique requires stable environmental light. Unfortunately, most commonly used light sources, incandescent or fluorescent, have some degree of fluctuation in intensity due to the use of Alternating-Current power.

- **Noise in image and quantization of image intensity**: When an image is captured, light intensities are first converted to voltage signals by the imaging sensor (CCD/CMOS sensor) and then quantized to grayscales by A/D converter. During this process, electric noises are picked up and added to the image intensities. Meanwhile, the intensity quantization of image introduces truncation errors.

- **Finite pixel size of imaging sensor**: In the construction of point clouds,

the position of a 3-D point is calculated from the sequence of image intensities associated with a pixel. Since a pixel in an imaging sensor has a finite size, it collects light from a small area on the object's surface instead of from an infinitesimal point. As a result, the calculated point position is an averaging of the positions of many points on a surface facet. The generated point may not lie on the object's surface precisely even if no other errors are present.

- **Imperfection of the reference plane**: The algorithm for point cloud construction requires a plane that is perfectly flat for the acquisition of a reference phase map. In practice, a flat plane can be manufactured to achieve a flatness of a few microns, in which case only negligible errors (comparing to other sources) would be caused in shape measurements. However, if the flatness of the reference plane is poor, this term needs to be considered.

6.4 Estimation of Uncertainty Components

6.4.1 Definitions of Uncertainty Components

The uncertainty components are defined based on the mathematical system model and the point cloud construction algorithm described in Chapter 3. Recall that the point cloud construction algorithm takes three sets of inputs: the sensor parameters, phase map of the object, and the reference phase map. The uncertainty components defined below are explained following this classification.

- **Uncertainties in sensor parameters**:

  The sensor parameters of a SMDFP system are estimated from calibration procedures. The uncertainties in sensor parameters are defined as the standard errors in the *estimated* values of the parameters. They reflect the possible errors introduced in the calibration procedures as well as the inaccuracy of the

system model. A detailed list of the uncertainties is given at below. The algorithm for estimation of the uncertainties will be explained in Section 6.4.2.

σ_{f_x}, σ_{f_y}, σ_{u_0} and σ_{v_0}: Standard deviations of the estimated parameters f_x, f_y, u_0 and v_0 respectively, which are coefficients in the intrinsic matrix of the camera.

σ_{k_s} $(s = 1, \cdots, 5)$: Standard deviations of the estimated lens distortion coefficients of the camera, k_s $(s = 1, \cdots, 5)$.

σ_P: The standard deviation of the estimated position of the projector's projection center, point P. It is assumed for simplicity that the uncertainty region of point P is a spherical volume, and the scalar value σ_P indicates the radius of the sphere.

σ_{O_R} and $\sigma_{\boldsymbol{n}_R}$: The position and orientation of the reference plane are represented by $(O_R, \boldsymbol{n}_R)$, where O_R is a point on the reference plane and $\boldsymbol{n}_R$ is the plane's normal direction. σ_{O_R} is defined as the standard deviation of the position of O_R along the direction of $\boldsymbol{n}_R$.

$\sigma_{\boldsymbol{n}_R}$ is defined as the standard deviation of $\arccos\big(\boldsymbol{n}_R(i) \cdot \bar{\boldsymbol{n}_R}\big)$, where $\boldsymbol{n}_R(i)$ are the estimated values of $\boldsymbol{n}_R$ and $\bar{\boldsymbol{n}_R}$ is the mean of $\boldsymbol{n}_R(i)$.

Detailed descriptions of the sensor parameters mentioned above can be found in Section 3.2.1 and 3.3.2.

- **Uncertainties in the phase map of the object**, $\sigma_\phi(i, j)$:

  The uncertainties in a phase map $\phi(i, j)$ are defined as an array $\sigma_\phi(i, j)$, which has the same dimension as $\phi(i, j)$. Each element in $\sigma_\phi(i, j)$ gives the estimated standard deviation of the corresponding phase value in $\phi(i, j)$.

Since a phase map is calculated from images of an object obtained under projection patterns, the uncertainties in a phase map are caused by error sources related to light projection and/or imaging, e.g. imperfections in light projection, fluctuation in environmental light, image noise and quantization of image intensity. The phase unwrapping algorithm may occasionally introduce additional errors in phase values, in the magnitude of multiples of 2π. However, such errors can be detected and are generally excluded as failed points. Hence, they are not considered in the uncertainties of the phase map.

The algorithm for estimation of phase uncertainty will be explained in Section 6.4.3.

- **Uncertainties in the reference phase map**, $\sigma_{\phi_R}(i,j)$:

  The uncertainties in the reference phase map, $\sigma_{\phi_R}(i,j)$, are defined in a similar manner as the uncertainties in the phase map of a generic object. The only difference is, $\sigma_{\phi_R}(i,j)$ may include phase uncertainties that are caused by the unevenness of the reference plane.

6.4.2 Estimation of Uncertainties in Sensor Parameters

Sensor parameters are estimated from calibration procedures. To be specific, camera parameters, as well as the position and orientation of reference plane, are estimated from camera calibration; and the projection center of the projector is estimated from a procedure described in Section 3.3.5.2. In both cases, the uncertainties in the parameter values are given as by-products of the calibration.

Take the estimation of uncertainties in camera parameters as an example. The camera parameters are estimated by solving a set of non-linear equations, which are established from the world coordinates and image coordinates of a number of 3-D points. Due to errors in the coordinates of the points, the equation set is solved

using a least-square procedure and the results do not satisfy all equations in general. From the magnitudes of the residual errors, the uncertainties in the parameter values obtained can be estimated. The algorithm for estimating the uncertainties in camera parameters was implemented in the camera calibration toolbox we have used [56].

6.4.3 Estimation of Phase Uncertainties

6.4.3.1 Relationship between Phase Uncertainty and Image Intensity Uncertainty

Phase maps of objects can be obtained by a number of different techniques. The most popular ones include Fourier Transform and phase-shifting. In the scope of this dissertation, only phase-shifting method is discussed. There are many different schemes for phase-shifting. The relationship between phase uncertainty and image intensity uncertainty depends on the specific scheme used. In the following, the standard N-step phase-shifting algorithm will be discussed. The analytical approach used, however, can be applied to other phase-shifting algorithms as well.

Images obtained from a standard N-step phase-shifting can be expressed using the following equation:

$$I_n(i,j) = A(i,j) + B(i,j)\sin\left(\phi(i,j) + 2\pi\frac{n-1}{N}\right) \tag{6.1}$$

where $I_n(i,j)$ is the intensity of pixel (i,j) in the n-th image ($n = 1, \cdots, N$), $\phi(i,j)$ is the phase value of pixel (i,j). $A(i,j)$ and $B(i,j)$ are two constant coefficients related to pixel (i,j), which correspond to the background light intensity and fringe contrast respectively. The phase map can be calculated from the images using the following equation:

$$\phi(i,j) = \arctan\left(\frac{\sum_{n=1}^{N}\left[I_n(i,j)\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n(i,j)\sin\left(2\pi\frac{n-1}{N}\right)\right]}\right) \tag{6.2}$$

In the above equation, each phase value $\phi(i,j)$ is given as a function of the image intensity sequence $I_n(i,j)$ $(n = 1, \cdots, N)$. The relationship between phase uncertainty and image intensity uncertainty can be derived from this function, using the principles of uncertainty propagation. Here, both the phase uncertainty and image intensity uncertainty are defined as the standard deviation of the calculated or sampled values.

Law 1 (Propogation of Variance) : *If* $y = f(x_1, x_2, \cdots, x_n)$ *is a continuous function and* x_1, x_2, $\cdots$, x_n *are uncorrelated variables, then*

$$\sigma_y^2 = \left(\frac{\partial y}{\partial x_1}\right)^2 \sigma_{x_1}^2 + \left(\frac{\partial y}{\partial x_2}\right)^2 \sigma_{x_2}^2 + \cdots + \left(\frac{\partial y}{\partial x_n}\right)^2 \sigma_{x_n}^2 + o\left(\max(\sigma_{x_1}^2, \sigma_{x_2}^2, \cdots, \sigma_{x_n}^2)\right)$$

where σ_y^2 *is the variance of* y *and* $\sigma_{x_k}^2$ $(k = 1, 2, \cdots, n)$ *is the variance of* x_k.

Let $\sigma_\phi(i,j)$ denote the standard deviation of $\phi(i,j)$ and let $\sigma_{I_n}(i,j)$ $(n = 1, \cdots, N)$ denote the standard deviation of I_n. By applying the Law of Variance Propagation to Eqn. 6.2, the relationship between $\sigma_\phi(i,j)$ and $\sigma_{I_n}(i,j)$ can be written as follows: (In the following context, the subscripts i and j of variables will be dropped for brevity)

$$\sigma_\phi^2 \approx \sum_{n=1}^{N} \left[\left(\frac{\partial \phi}{\partial I_n}\right)^2 \sigma_{I_n}^2\right] \tag{6.3}$$

The partial derivative of ϕ w.r.t. I_n can be derived from Eqn. 6.2:
(see Appendix C.1 for the detailed derivation procedures involved in this section)

$$\frac{\partial \phi}{\partial I_n} = \frac{2}{NB} \cos\left(\phi + 2\pi \frac{n-1}{N}\right) \tag{6.4}$$

Assume that the influence of error sources is the same on all the N images acquired

from phase-shifting, we have

$$\sigma_{I_1} = \sigma_{I_2} = \cdots = \sigma_{I_N} = \sigma_I \tag{6.5}$$

Utilizing this property and substituting Eqn. 6.4 into Eqn. 6.3, we can get the following equation for σ_ϕ:

$$\sigma_\phi(i,j) \approx \sqrt{\frac{2}{N}}\,\frac{\sigma_I(i,j)}{B(i,j)} \tag{6.6}$$

where N is the number of steps in the phase-shifting method and $B(i,j)$ is the fringe contrast coefficient as defined in Eqn. 6.1.

6.4.3.2 Estimation of Image Intensity Uncertainty

Due to the errors in image intensities, images obtained from a standard N-step phase-shifting generally do not fully comply with Eqn. 6.1, when N is greater than 3. In other words, $I_n(i,j)$ $(n = 1, \cdots, N)$ does not give exact solution to $\phi(i,j)$, $A(i,j)$ and $B(i,j)$ since Eqn. 6.1 is overdetermined. The degree of disagreement between the intensity sequence I_n and Eqn. 6.1 reflects the magnitude of errors in I_n. In the following, an algorithm for estimation of image intensity uncertainty is proposed. This algorithm is presented based on the standard N-step phase-shifting algorithm $(N > 3)$ but can be extended to other phase-shifting algorithms as well.

- First of all, the phase map $\phi(i,j)$ is calculated from the images $I_n(i,j)$ by using Eqn. 6.2. Since the errors in $I_n(i,j)$ are random errors, the phase map calculated is an unbiased estimation of the true phase map, meaning that the disagreement between I_n and Eqn. 6.1 reaches the minimum. A detailed proof of this is presented in Appendix C.2.

- Secondly, the estimated values of $A(i,j)$ and $B(i,j)$ are calculated using the

following equations for each pixel (i, j) (subscripts i and j dropped for brevity):

$$\tilde{A} = \frac{1}{N}\sum_{n=1}^{N} I_n \tag{6.7}$$

$$\tilde{B} = \frac{2}{N}\sqrt{\left\{\sum_{n=1}^{N}\left[I_n \cos\left(2\pi\frac{n-1}{N}\right)\right]\right\}^2 + \left\{\sum_{n=1}^{N}\left[I_n \sin\left(2\pi\frac{n-1}{N}\right)\right]\right\}^2} \tag{6.8}$$

- For each pixel (i, j), the calculated phase value $\phi(i, j)$ and coefficients $\tilde{A}(i, j)$ and $\tilde{B}(i, j)$ can be used to compute a sequence of "compensated" intensity values, $\tilde{I}_n(i, j)$ $(n = 1, \cdots, N)$, by using the following equation:

$$\tilde{I}_n(i, j) = \tilde{A}(i, j) + \tilde{B}(i, j) \sin\left(\phi(i, j) + 2\pi\frac{n-1}{N}\right) \tag{6.9}$$

  $\tilde{I}_n(i, j)$ are generally different from their measured counterparts, $I_n(i, j)$, and the degree of disagreement reflects the magnitude of errors in $I_n(i, j)$. The standard deviation of the image intensity error, $\sigma_I(i, j)$, can be calculated from $\tilde{I}_n(i, j)$ and $I_n(i, j)$ by using the following equation:

$$\sigma_I(i, j) \approx \sqrt{\frac{N}{(N-3)(N-1)}\sum_{n=1}^{N}\left[\tilde{I}_n(i, j) - I_n(i, j)\right]^2} \tag{6.10}$$

  A detailed proof of this equation is given in Appendix C.3.

- Although σ_I can be estimated individually for each pixel by using Eqn. 6.10, the estimated value is not accurate due to the limited number of I_n $(n = 1, \cdots, N)$ samples, since N is usually selected to be less than 7 for a fast measurement speed. As a solution, the values of σ_I can be averaged among pixels with a similar fringe contrast, i.e. B value. The detailed algorithm is the following:

  In the first step, all pixels in the phase map are divided into a number of groups

based on their B values. For example, for a phase map obtained from images of 8-bit gray-depth, the possible range of B is $(0, 127.5]$. Hence, the pixels can be put into 128 groups, with each group having a B range of 1 or less, which is a good resolution for estimating σ_I. In the second step, the σ_I values of all pixels in a group are averaged and the result is assigned to the whole group.

The averaging of σ_I among pixels with a similar fringe contrast (B value) will give more accurate estimations of σ_I. Here we assume that the image intensity error has no relevance (or only a weak relevance) to the location of pixels in the image or the pixels' A values, which is reasonably true for the error sources we have discussed (see Section 6.3).

The algorithm presented above for estimation of uncertainty in image intensity has a number of advantages. First of all, it operates on the images acquired in a real measurement instead of images from calibration. Hence it is able to take into account the error sources that are measurement-related. Secondly, it only requires the images from phase-shifting as inputs and hence adds no additional cost to measurement. Thirdly, it works with any phase-shifting algorithms as long as the number of phase-shift steps is more than 3.

6.4.3.3 Algorithm for Estimation of Phase Uncertainties

A schematic diagram of the algorithm for estimation of phase uncertainties is shown in Fig. 6.2. The major steps in the algorithm can be described as follows:

The estimation of phase uncertainties takes place after the acquisition of an phase map, $\phi(i, j)$. The intensity uncertainties in the images are estimated as the first step, using the algorithm described in Section 6.4.3.2. The result is given by an array $\sigma_I(i, j)$, which has the same dimension as the phase map $\phi(i, j)$. Each element in the array represents the estimated uncertainty in the image intensity values of the

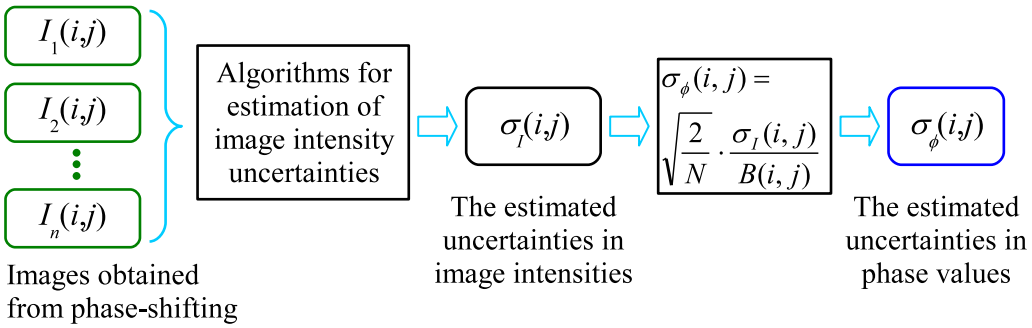


Figure 6.2: Schematic diagram of the algorithm for estimation of phase uncertainties

corresponding pixel. The estimated uncertainties in the phase map, $\sigma_\phi(i,j)$, can then be calculated using Eqn. 6.6.

6.4.3.4 Validation of the Algorithm by Simulations

In order to validate the proposed algorithm for estimation of phase uncertainties, a simulation procedure was developed. A detailed description of the simulation is as the following:

1. For an arbitrary phase value $\hat{\phi}$ in the range of $[-\pi, \pi]$ and selected values for A and B, compute the image intensity sequence corresponding to the N-step phase-shifting, i.e.

$$\hat{I}_n = A + B\sin\left(\hat{\phi} + 2\pi\frac{n-1}{N}\right) , \quad n = 1, \cdots, N \tag{6.11}$$

2. Add random noise to $\hat{I}_n$ using the equation at below and let I_n denote the result:

$$I_n = \hat{I}_n + L_N \,\text{rand}() , \quad n = 1, \cdots, N \tag{6.12}$$

The noise added have either a uniform distribution or a normal distribution.

In the former case, function rand() generates uniformly distributed random numbers in the range of $[-0.5, 0.5]$. In the latter case, the function generates normally distributed random numbers with a mean value of 0 and a standard deviation of 1. L_N is a scale coefficient to control the magnitude of the noise.

3. Compute the phase value from the erroneous image intensity sequence I_n by using Eqn. 6.2. The result is denoted by ϕ.

4. Repeat Step 2 and 3 for M times to acquire $I_n(m)$ and $\phi(m)$ $(m = 1, \cdots, M)$. Compute the standard deviation of $\left(\phi(m) - \hat{\phi}\right)$ and let $\hat{\sigma}_\phi$ denote the result. $\hat{\sigma}_\phi$ is the true value of the uncertainty in the calculated phase values. On the other hand, the estimated value of phase uncertainty, σ_ϕ, can be obtained from $I_n(m)$ by using the proposed algorithm. The difference between σ_ϕ and $\hat{\sigma}_\phi$ is an indicator of the effectiveness of the estimation algorithm.

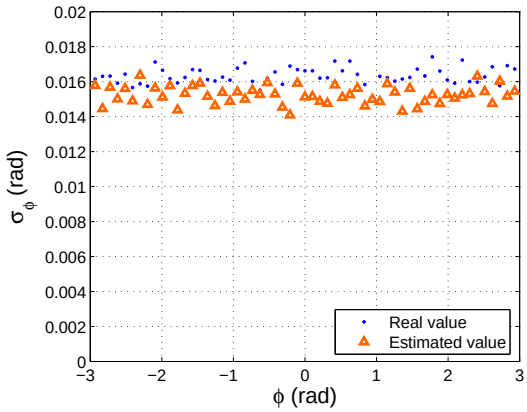
Using the procedure described above, a number of simulation tests were performed with different phase values, noise levels and distributions, and phase-shifting methods. The proposed algorithm for estimation of phase uncertainties worked well for all cases.

Figure 6.3(a) shows the true values and estimated values of σ_ϕ for a number of phase values ranging from $-\pi$ to π. The smaller blue dots represent the true values and the bigger orange triangles represent the estimated values. The result was obtained using the standard 4-step phase-shifting method, with a selected B value of 100, uniformly distributed noise whose magnitude $L_N = 8$, and 500 samples of I_n sequence for each phase value. In real measurements, the number of samples available are generally much larger than 500. As can be seen from the figure, the estimated values of σ_ϕ are pretty close to their true values although slightly smaller on average. The magnitudes of σ_ϕ are about the same for different phase values, which agrees with Eqn. 6.6. Figure 6.3(b) shows the simulation result under the same conditions but with the noise magnitude doubled ($L_N = 16$). The distribution pattern of the

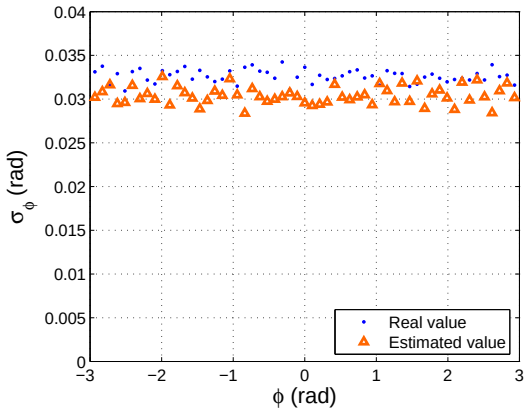
true values and estimated values is about the same, except that the magnitudes of σ_ϕ are doubled accordingly.

For the 4-step phase-shifting method, there is a fixed ratio difference between the estimated and the true values of σ_ϕ, which can be seen clearly from the results based on a large number of samples. Figure 6.3(c) shows a case with 10000 samples of I_n sequence. Other conditions of the simulation are: $B = 100$ and uniformly distributed noise with $L_N = 8$. The biased error in the estimated values is caused by the algorithm for estimation of image intensity uncertainty (see Section 6.4.3.2). Equation 6.10 would not give good estimations for the case of $N = 4$, since only 4 samples are available and only 2 of them are independent. However, as it will be shown next, this problem is almost eliminated for the 6-step phase-shifting method, in which case more samples are available to make individual estimations of the image intensity uncertainty. From the many simulations conducted with the 4-step phase-shifting method, it was found that the ratio between the estimated values and the true values is about 0.93, for all phase values and different noise levels. This constant ratio can be used to compensate the estimated values for the 4-step phase-shifting method.

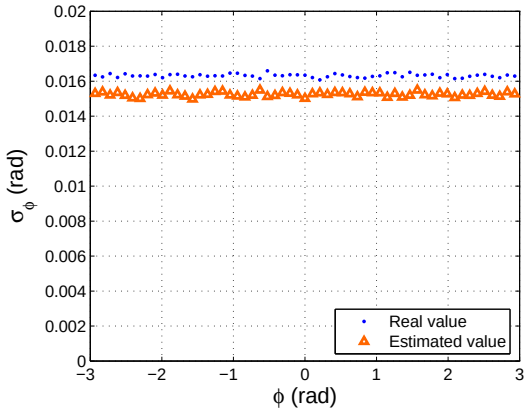
Three examples with the 6-step phase-shifting method are shown in Fig. 6.3. The result shown in Fig. 6.3(e) was obtained with the settings: $B = 100$, uniformly distributed noise with $L_N = 8$ and 10000 samples. As can be seen from the plot, the difference between the estimated and the true values of σ_ϕ is very small. The remaining disagreement is caused by the uniform distribution of the noise, since the proposed algorithm assumes normally distributed noise (see Eqn. 6.10). In Fig. 6.3(f), which was obtained with normally distributed noise, the estimated values and the true values of σ_ϕ match almost perfectly. Under the same conditions, the phase error yielded with the use of 6-step phase-shifting is about 82% in magnitude of its counterpart with the 4-step phase-shifting method, which can be seen from the



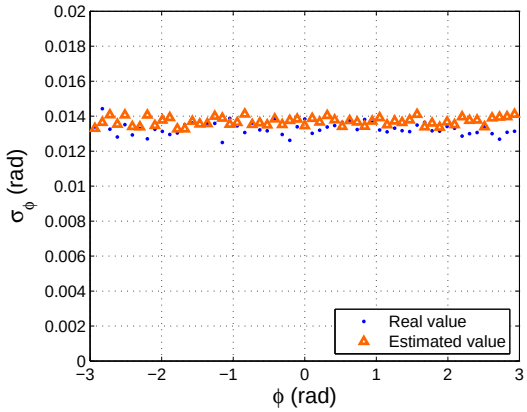
(a) $N = 4$, $B = 100$, $L_N = 8$ (uniform), $M = 500$



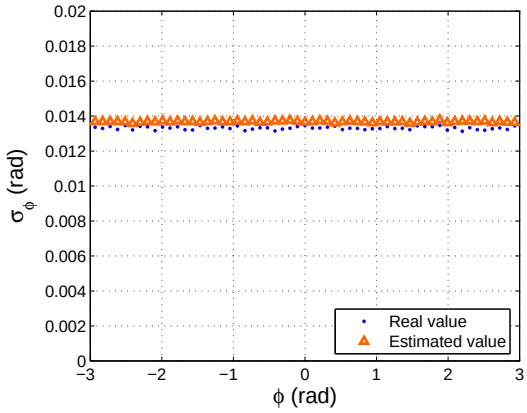
(b) $N = 4$, $B = 100$, $L_N = 16$ (uniform), $M = 500$



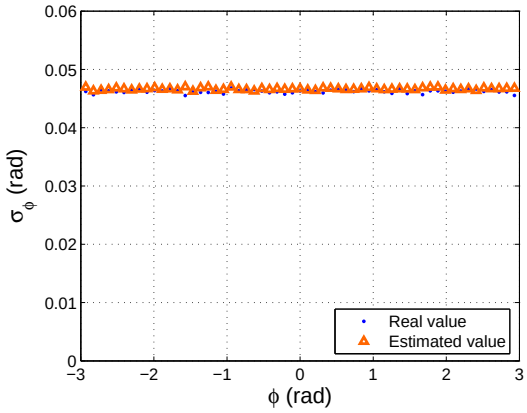
(c) $N = 4$, $B = 100$, $L_N = 8$ (uniform), $M = 10000$



(d) $N = 6$, $B = 100$, $L_N = 8$ (uniform), $M = 500$

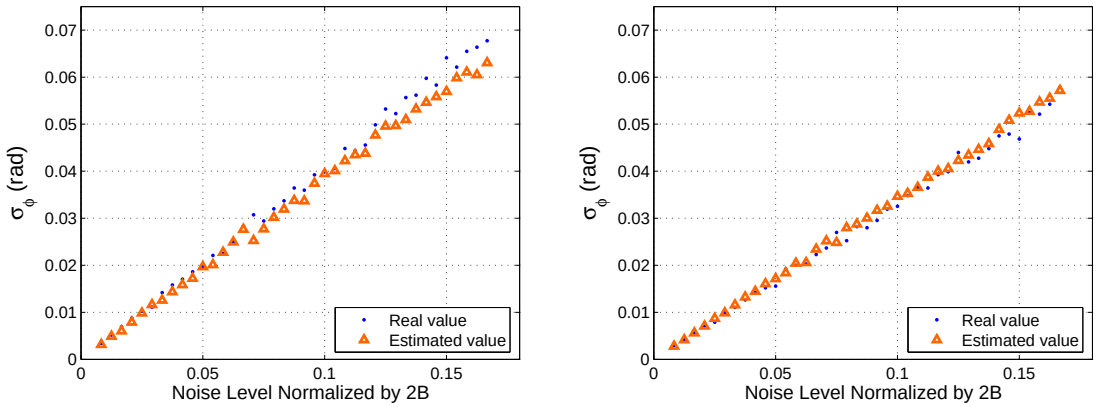


(e) $N = 6$, $B = 100$, $L_N = 8$ (uniform), $M = 10000$



(f) $N = 6$, $B = 100$, $L_N = 8$ (normal), $M = 10000$

Figure 6.3: Estimation of phase uncertainties, $\phi \in (-\pi, \pi)$



(a) $N = 4$, $\phi = \pi/3$, $M = 500$ (b) $N = 6$, $\phi = \pi/3$, $M = 500$

Figure 6.4: Estimation of phase uncertainties, varied noise levels

comparison of Fig. 6.3(c) and 6.3(e). This result agrees with Eqn. 6.6.

Figure 6.4 shows the changes of σ_ϕ as the magnitude of noise increases. The X-axis of the figures represents the normalized noise level, which is defined as $L_N/(2B)$. The common settings of the simulations are: $\phi = \pi/3$; 500 samples of I_n sequence were used for each noise level; and the noise were generated with uniform distribution. Figure 6.4(a) shows the case with the 4-step phase-shifting method and Fig. 6.4(b) for the 6-step phase-shifting method. The estimated values are close to the true values at all tested noise levels, which shows the effectiveness of the proposed algorithm for varied noise levels. Also, the value of σ_ϕ shows a linear relationship with the noise magnitude, which agrees with Eqn. 6.6.

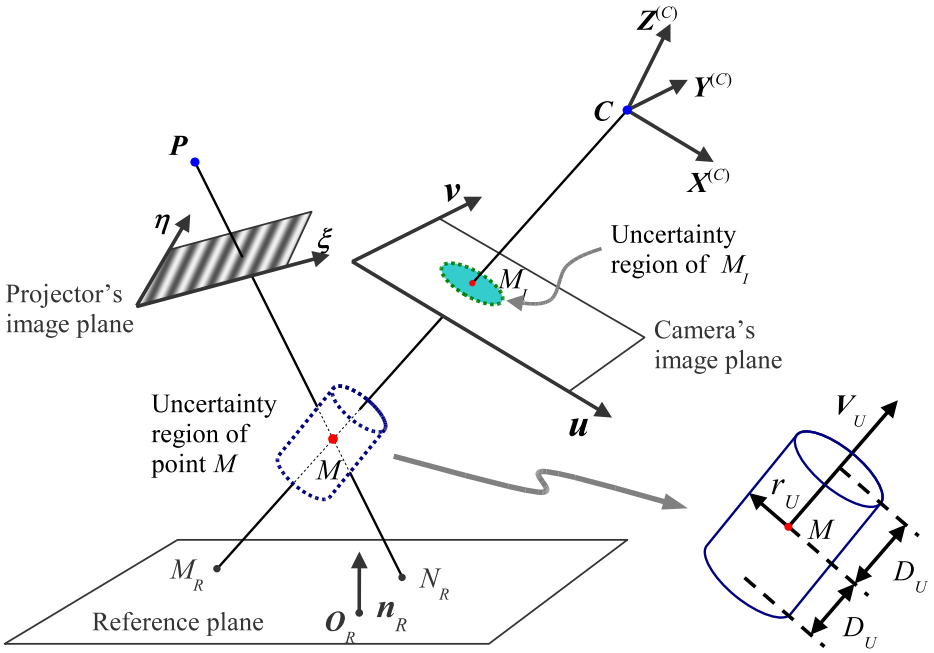


Figure 6.5: Cylindrical uncertainty region of a measured point

6.5 Estimation of Position Uncertainties of Measured Points

6.5.1 Cylindrical Uncertainty Region

In the proposed uncertainty model, the 3-D uncertainty region of each measured point is represented by a cylinder which takes the measured point as its center (see Fig. 6.5).

Recall the algorithm for the construction of point cloud (see Section 3.3.4). For each pixel in the object's phase map, a 3-D point can be generated and the calculation of its coordinates is done in two steps. First, a point on the camera's image plane, say point M_I, can be located by using the pixel's coordinates (i, j). A ray starting from the camera's projection center (point C) and passing through point M_I, can be determined. It is known from the pinhole camera model that, all points in the 3-D space that can be imaged to pixel (i, j) must lie on $\overrightarrow{CM_I}$. Therefore, two out of the

three dimensional information of the point, say M, are determined. The one degree of freedom left is the depth of point M w.r.t. the camera's projection center C, i.e. the distance $|CM|$. $|CM|$ can be decided from the corresponding phase value of M_I, $\Phi(i,j)$, which is given in the object's phase map.

Based on the above algorithm, the uncertainties in the 3-D coordinates of point M, or any other point in the point cloud, can be divided into two parts: the uncertainty of the orientation of $\overrightarrow{CM_I}$ and the uncertainty of the depth of M. Here, we consider the position uncertainty of M w.r.t. the camera coordinate frame. Therefore, point C is the absolute origin without any uncertainty in position.

The position of point M_I, and hence the orientation of $\overrightarrow{CM_I}$, are calculated from the pixel coordinates (i,j) and the camera parameters. They are subject to the uncertainties of the camera parameters. By following the principles of uncertainty propagation, the uncertainty of the position of M_I can be calculated from the known uncertainties in the camera parameters which were estimated from calibration. For simplicity, this uncertainty can be described by a circular region around M_I on the camera's image plane, whose radius is the standard deviation of the position of M_I. Following this definition, the resulting uncertainty space of $\overrightarrow{CM_I}$ is a cone which has point C as its apex and $\overrightarrow{CM_I}$ as its axis.

The depth of point M, i.e. $|CM|$, is calculated from a number of variables: the phase value of pixel (i,j), the camera parameters and projector parameters, the position and orientation of the reference plane, and the reference phase map. The uncertainty of $|CM|$ can be represented by the standard deviation of $|CM|$, denoted by $D_U(M)$.

By combining the uncertainties in $\overrightarrow{CM_I}$ and $|CM|$, the uncertainty region of the position of M can be determined. This 3-D region can be modeled as a cylinder, which is given by 4 parameters (see Fig. 6.5): the position of point M, the cylinder's axis $\boldsymbol{V}_U(M)$, radius $r_U(M)$, and half height $D_U(M)$. Among them, $D_U(M)$ is the

uncertainty of the depth of M, and $r_U(M)$ corresponds to the uncertainty of the orientation of $\overrightarrow{CM_I}$. $\boldsymbol{V}_U(M)$ is the unit vector that points to the opposite direction of $\overrightarrow{CM_I}$, i.e.

$$\boldsymbol{V}_U(M) = -\frac{\overrightarrow{CM_I}}{|CM_I|} \tag{6.13}$$

In the following sections, the evaluations of r_U and D_U are explained in details. For brevity, the uncertainty of the orientation of $\overrightarrow{CM_I}$ will be referred as the uncertainty in ray direction.

6.5.2 Estimation of Uncertainty in Ray Direction

6.5.2.1 Algorithm for Estimation of Uncertainty in Ray Direction

The ray direction $\overrightarrow{CM_I}$ is calculated from the pixel coordinates of M_I and the camera parameters. The uncertainty in the ray direction is determined by the uncertainties of the camera parameters as well as the camera parameters themselves. Specifically, the following inputs are required for the estimation:

- The camera's intrinsic parameters: f_x, f_y, u_0 and v_0.
- The lens distortion coefficients of the camera: k_s $(s = 1, \cdots, 5)$.
- The uncertainties in the camera parameters: σ_{f_x}, σ_{f_y}, σ_{u_0}, σ_{v_0} and σ_{k_s} $(s = 1, \cdots, 5)$, which are the standard deviations of f_x, f_y, u_0, v_0 and k_s $(s = 1, \cdots, 5)$, respectively.

Consider the uncertainty in the position of M_I. Recall that the position of M_I is calculated from a transformation of the pixel coordinates of M_I to its undistorted image coordinates. The transformation can be described in the following manner (see Section 3.2.1 for more details):

1. The first step is the transformation from the pixel coordinates, denoted by $(x^{(p)}, y^{(p)})$, to the distorted image coordinates, $(x^{(d)}, y^{(d)})$, which still carry the lens distortions. This transformation can be expressed using the following equation:

$$\begin{bmatrix} x^{(d)} \\ y^{(d)} \\ 1 \end{bmatrix} = \boldsymbol{A}^{-1} \cdot \begin{bmatrix} x^{(p)} \\ y^{(p)} \\ 1 \end{bmatrix}, \qquad \boldsymbol{A}^{-1} = \begin{bmatrix} 1/f_x & 0 & -u_0/f_x \\ 0 & 1/f_y & -v_0/f_y \\ 0 & 0 & 1 \end{bmatrix} \tag{6.14}$$

where $\boldsymbol{A}$ is the camera's intrinsic matrix and f_x, f_y, u_0 and v_0 are the camera's intrinsic parameters. Notice that the skewness coefficient α in matrix $\boldsymbol{A}$ has been assumed to be zero, which is true for cameras based on CCD/CMOS imaging sensors.

2. The second step is to remove the lens distortions in coordinates $(x^{(d)}, y^{(d)})$ and convert them to the undistorted image coordinates, $(x^{(n)}, y^{(n)})$. This conversion can be solved numerically from the following equation:

$$\begin{bmatrix} x^{(d)} \\ y^{(d)} \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix} + \begin{bmatrix} 2k_3 x^{(n)} y^{(n)} + k_4 \left(r^2 + 2\left(x^{(n)}\right)^2\right) \\ k_3 \left(r^2 + 2\left(y^{(n)}\right)^2\right) + 2k_4 x^{(n)} y^{(n)} \end{bmatrix} \tag{6.15}$$

where $r^2 = \left(x^{(n)}\right)^2 + \left(y^{(n)}\right)^2$ and k_s $(s = 1, \cdots, 5)$ are the lens distortion coefficients.

In the first step of the above transformation, the uncertainties of the camera parameters will be propagated to the calculated image coordinates $(x^{(d)}, y^{(d)})$. Let $\sigma_{x^{(d)}}$ and $\sigma_{y^{(d)}}$ be the standard deviations of $x^{(d)}$ and $y^{(d)}$ respectively. Rewrite Eqn. 6.14

as follows:

$$\begin{cases} x^{(d)} = \left(x^{(p)} - u_0\right)/f_x \\ y^{(d)} = \left(y^{(p)} - v_0\right)/f_y \end{cases} \tag{6.16}$$

from which the partial derivatives of $x^{(d)}$ and $y^{(d)}$ can be derived:

$$\begin{aligned} \frac{\partial x^{(d)}}{\partial u_0} &= -\frac{1}{f_x}\,, \qquad \frac{\partial x^{(d)}}{\partial f_x} = -\frac{(x^{(p)} - u_0)}{f_x^2} \\ \frac{\partial y^{(d)}}{\partial v_0} &= -\frac{1}{f_y}\,, \qquad \frac{\partial y^{(d)}}{\partial f_y} = -\frac{(y^{(p)} - v_0)}{f_y^2} \end{aligned} \tag{6.17}$$

Applying the Law of Variance Propagation to Eqn. 6.16 and substituting the partial derivatives listed above, the following equations about $\sigma_{x^{(d)}}$ and $\sigma_{y^{(d)}}$ can be obtained:

$$\begin{cases} \sigma_{x^{(d)}}^2 \approx \dfrac{\sigma_{u_0}^2}{f_x^2} + \dfrac{\sigma_{f_x}^2\left(x^{(p)} - u_0\right)^2}{f_x^4} \\ \sigma_{y^{(d)}}^2 \approx \dfrac{\sigma_{v_0}^2}{f_y^2} + \dfrac{\sigma_{f_y}^2\left(y^{(p)} - v_0\right)^2}{f_y^4} \end{cases} \tag{6.18}$$

A similar derivation can be applied to the second step of the transformation to resolve the uncertainty propagation from $(x^{(d)}, y^{(d)})$ to $(x^{(n)}, y^{(n)})$. Let $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ be the standard deviations of $x^{(n)}$ and $y^{(n)}$ respectively. The equations for computing $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ can be written as follows:

$$\begin{aligned} \sigma_{x^{(n)}}^2 \approx & \left[\left(\frac{\partial y^{(d)}}{\partial y^{(n)}}\right)^2 \sigma_{x^{(d)}}^2 + \left(\frac{\partial x^{(d)}}{\partial y^{(n)}}\right)^2 \sigma_{y^{(d)}}^2 + \sum_{s=1}^{5}\left(\frac{\partial y^{(d)}}{\partial y^{(n)}}\frac{\partial x^{(d)}}{\partial k_s} - \frac{\partial x^{(d)}}{\partial y^{(n)}}\frac{\partial y^{(d)}}{\partial k_s}\right)^2 \sigma_{k_s}^2\right] \\ & \Bigg/ \left(\frac{\partial x^{(d)}}{\partial x^{(n)}}\frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}}\frac{\partial x^{(d)}}{\partial y^{(n)}}\right)^2 \end{aligned} \tag{6.19}$$

$$\begin{aligned} \sigma_{y^{(n)}}^2 \approx & \left[\left(\frac{\partial y^{(d)}}{\partial x^{(n)}}\right)^2 \sigma_{x^{(d)}}^2 + \left(\frac{\partial x^{(d)}}{\partial x^{(n)}}\right)^2 \sigma_{y^{(d)}}^2 + \sum_{s=1}^{5}\left(\frac{\partial x^{(d)}}{\partial x^{(n)}}\frac{\partial y^{(d)}}{\partial k_s} - \frac{\partial y^{(d)}}{\partial x^{(n)}}\frac{\partial x^{(d)}}{\partial k_s}\right)^2 \sigma_{k_s}^2\right] \\ & \Bigg/ \left(\frac{\partial x^{(d)}}{\partial x^{(n)}}\frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}}\frac{\partial x^{(d)}}{\partial y^{(n)}}\right)^2 \end{aligned} \tag{6.20}$$

A detailed proof of the above equation, as well as the equations for computing the partial derivatives involved, is presented in Appendix C.4. In general, only the radial distortion needs to be considered in the lens distortions, i.e. coefficients k_2 to k_5, as well as their uncertainties, can be ignored. As a result, Eqn. 6.19 and 6.20 can be simplified to the following forms:

$$\begin{aligned}\sigma^2_{x^{(n)}} \approx \Bigg[&\left(1 + k_1 r^2 + 2k_1 \left(y^{(n)}\right)^2\right)^2 \sigma^2_{x^{(d)}} + \left(2k_1 x^{(n)} y^{(n)}\right)^2 \sigma^2_{y^{(d)}} + \\ &\left(\left(1 + k_1 r^2\right) r^2 x^{(n)}\right)^2 \sigma^2_{k_1} \Bigg] \Bigg/ \Big[\left(1 + k_1 r^2\right) \left(1 + 3k_1 r^2\right) \Big]^2 \end{aligned} \tag{6.21}$$

$$\begin{aligned}\sigma^2_{y^{(n)}} \approx \Bigg[&\left(2k_1 x^{(n)} y^{(n)}\right)^2 \sigma^2_{x^{(d)}} + \left(1 + k_1 r^2 + 2k_1 \left(x^{(n)}\right)^2\right)^2 \sigma^2_{y^{(d)}} + \\ &\left(\left(1 + k_1 r^2\right) r^2 y^{(n)}\right)^2 \sigma^2_{k_1} \Bigg] \Bigg/ \Big[\left(1 + k_1 r^2\right) \left(1 + 3k_1 r^2\right) \Big]^2 \end{aligned} \tag{6.22}$$

Let $r_{UI}(M_I)$ denote the radius of the uncertainty region of M_I (see Fig. 6.5). $r_{UI}(M_I)$ is defined in terms of $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ as the following:

$$r_{UI}(M_I) = \sqrt{\sigma^2_{x^{(n)}} + \sigma^2_{y^{(n)}}} \tag{6.23}$$

With this definition, the probability of the true position of M_I falling within the uncertainty region of the calculated position of M_I, is approximately 68%.

Finally, the radius of the cylindrical uncertainty region for point M, $r_U(M)$, can be calculated as follows:

$$r_U(M) = \frac{|CM|}{|CM_I|} r_{UI}(M_I) = \frac{|CM|}{|CM_I|} \sqrt{\sigma^2_{x^{(n)}} + \sigma^2_{y^{(n)}}} \tag{6.24}$$

6.5.2.2 Validation of the Algorithm by Simulations

To validate the developed algorithm for estimation of the uncertainty in ray direction, a simulation procedure was developed. A detailed description of the simulation is as

the following:

1. For a set of selected camera parameter values and an arbitrary pixel $(x^{(p)}, y^{(p)})$, add a certain amount of random errors to the camera parameters and compute the corresponding undistorted image coordinates, $(x^{(n)}, y^{(n)})$, by using Eqn. 6.14 and 6.15. Repeat this process to obtain a number of M samples of $(x^{(n)}, y^{(n)})$.

2. The standard deviations of the coordinates $x^{(n)}$ and $y^{(n)}$, denoted by $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ respectively, can be calculated from the M samples. On the other hand, $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ can be estimated using the algorithm described in the last section. The estimated values of $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ can be compared against their true values and the disagreement reflects the soundness of the estimation algorithm.

Using the procedure described above, a number of simulation tests were conducted with two different sets of camera parameters that were obtained from real SMDFP systems. The magnitudes of the random errors in simulation data were set to the estimated uncertainties in the camera parameters.

The results of two simulation runs are shown in Fig. 6.6 and 6.7. The camera parameters used in the simulation, as well as the magnitudes of random errors, are listed in Tab. 6.1. The data were obtained from a B/W digital camera (1/3" Sony CCD, 640×480 resolution) with a Computar$^{\text{TM}}$ M1214-MP lens (12mm focal length, F1.4). Fig. 6.6 shows the simulation results for the pixels whose $y^{(p)}$-coordinates are 240 and $x^{(p)}$-coordinates range from 0 to 640. The orange triangles in the figure represent the true values of $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$, and the blue dots represent the estimated values. The results were based on 1000 samples. As can be seen from the figure, the estimated values match with the true values very well. Fig. 6.7 shows the simulation results for pixels whose $x^{(p)}$-coordinates are 320 and $y^{(p)}$-coordinates range from 0 to 480.

In all simulation tests performed, the estimated values of $\sigma_{x^{(n)}}$ and $\sigma_{y^{(n)}}$ match with their true values, which shows that the algorithm developed for estimation of uncertainty in ray direction works well.

6.5.3 Estimation of Uncertainty in Depth

6.5.3.1 Overview of the Algorithm

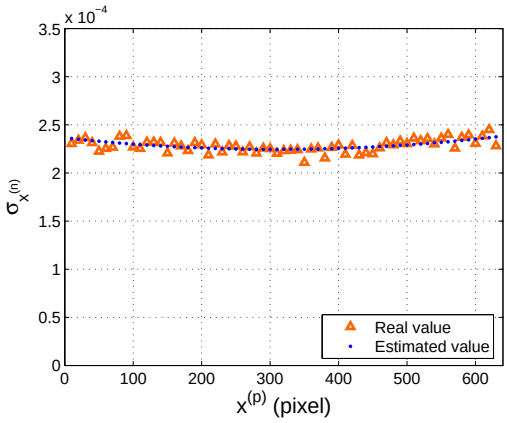
The computation of the depth of a point involves a number of variables, e.g. the absolute phase value of the point, the reference phase map, the position and orientation of the reference plane, and the position of the projector's projection center. The uncertainties in these variables could propagate to the uncertainty in the calculated depth value. The proposed algorithm begins with finding the relationships between the uncertainties of individual variables and the corresponding uncertainties they could cause in the depth. The overall depth uncertainty can then be estimated by combining the individual uncertainties.

Recall the process for computing the depth of a point, say M, during the construction of point cloud (see Section 3.3.4 for details). Before the process, the position and orientation of the ray $\overrightarrow{CM_I}$ have been fixed (see Fig. 6.8). The location of M along $\overrightarrow{CM_I}$ is determined through a triangulation process. As the first step, a point N_I is located on the camera's image plane ($\boldsymbol{u} - \boldsymbol{v}$) by searching along line $\overline{P_I M_I}$ for $\Phi_R(N_I) = \Phi(M_I)$, i.e. the phase value of point N_I in the reference phase map is equal to the phase value of point M_I in the object's phase map. Here, P_I is the projection of point P on the camera's image plane. Once N_I is located, its corresponding point on the reference plane, N_R, can be located accordingly. The depth of point M can then be determined by finding the intersection point of $\overline{CM_I}$ and $\overline{PN_R}$.

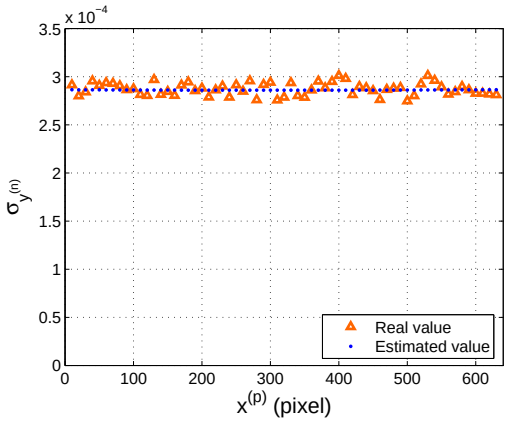
The components that could affect the depth of M are the positions of points P and N_R. The major factors that affect the position of N_R are the phase value of M, the

	f_x	f_y	u_0	v_0	k_1
Value	1633	1631.7	309.2	219.3	-0.049
STD of random errors	0.6	0.6	0.37	0.47	0.0017

Table 6.1: Camera parameters and error levels used in the simulation

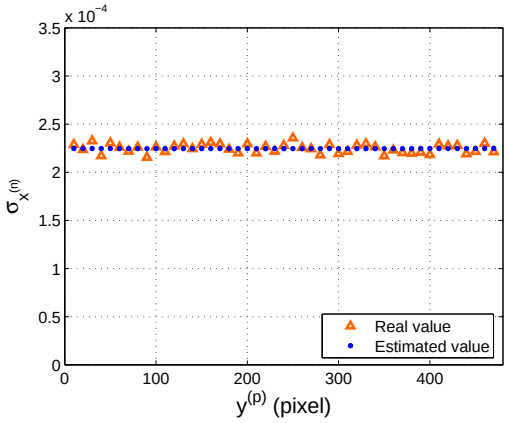


(a) Estimated and true values of $\sigma_{x^{(n)}}$

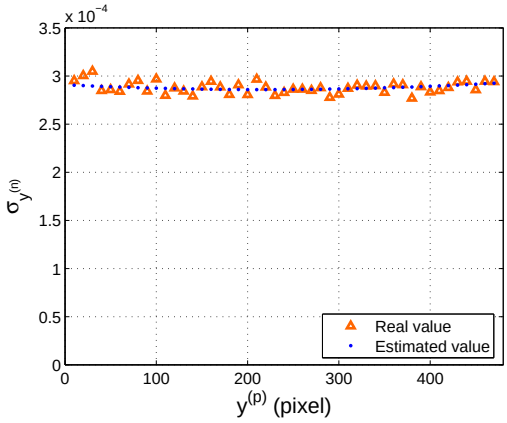


(b) Estimated and true values of $\sigma_{y^{(n)}}$

Figure 6.6: Estimation of uncertainty in ray direction: $y^{(p)} = 240$, $x^{(p)} \in [0, 640)$



(a) Estimated and true values of $\sigma_{x^{(n)}}$



(b) Estimated and true values of $\sigma_{y^{(n)}}$

Figure 6.7: Estimation of uncertainty in ray direction: $x^{(p)} = 320$, $y^{(p)} \in [0, 480)$

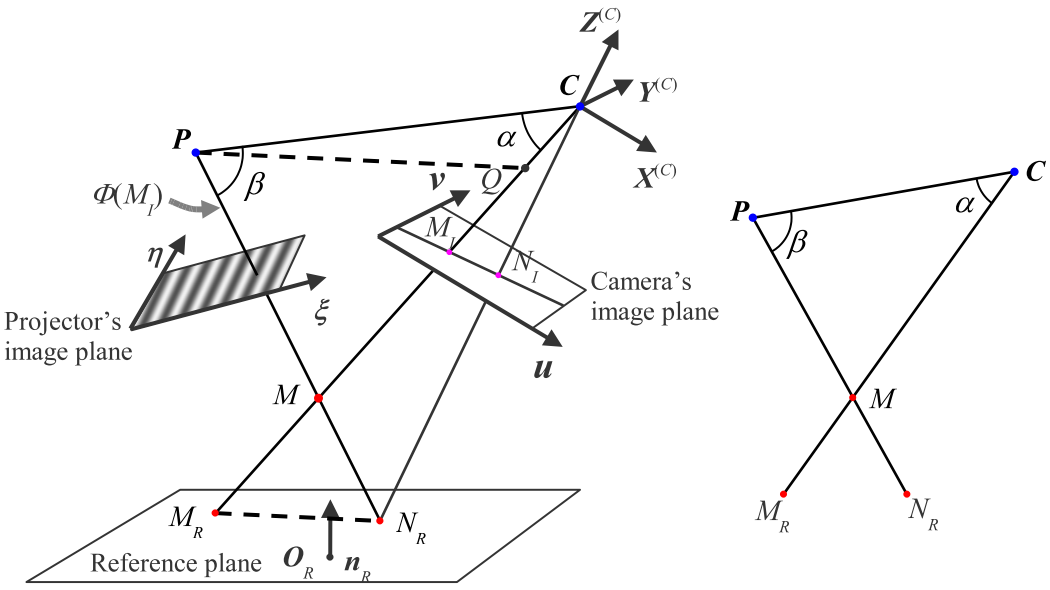


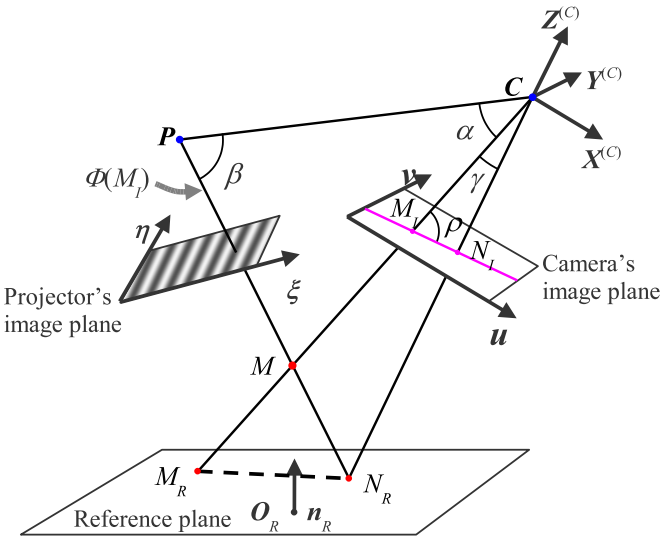
Figure 6.8: Triangulation process to determine the depth of point M

reference phase map, and the position and orientation of the reference plane. In the successive sections, the influence of the uncertainty components to the uncertainty in depth will be examined from the following aspects individually:

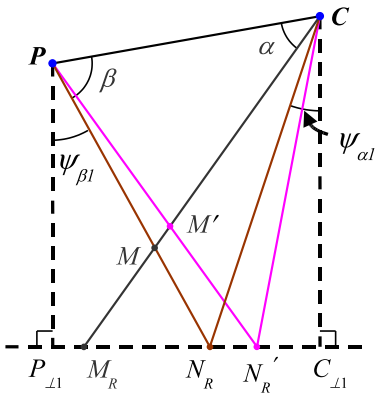
- The uncertainties in phase values, which include uncertainties in the object's phase map as well as the uncertainties in the reference phase map;
- The uncertainties in the position and orientation of the reference plane; and
- The uncertainty of the position of the projector's projection center.

6.5.3.2 Influence of Phase Uncertainties

In the computation of the depth of point M, the position of point N_I is located by searching along line $\overline{P_I M_I}$ for $\Phi_R(N_I) = \Phi(M_I)$, i.e. the phase value of N_I in the reference phase map is equal to the phase value of M_I in the object's phase map. If there are errors in $\Phi(M_I)$ and/or in the close neighborhood of N_I in the reference phase map, the calculated position of N_I would be affected. In turn, the position of



(a) Definition of angle γ and ρ



(b) Definition of angle $\psi_{\beta 1}$ and $\psi_{\alpha 1}$

Figure 6.9: Relationship between the position of N_I and angle β

point N_R would be affected, since it is the intersection of line $\overline{CN_I}$ with the reference plane.

Let line l_{FR} denote the intersection of the reference plane and the plane that is defined by points C, P and M_I. It is easy to see that points M_R and N_R are on this line. Let points $C_{\perp 1}$ and $P_{\perp 1}$ be the projections of points C and P, respectively, on line l_{FR} (see Fig. 6.9(b)). It can be seen that, as a result of possible errors in $\Phi(M_I)$ and in the reference phase map, point N_R would move along the line $\overline{P_{\perp 1}C_{\perp 1}}$.

The relationship between an error in the position of N_I and the error it causes in value of $|CM|$ (i.e. the depth of point M) can be derived as follows:

- **Relationship between $|CM|$ and angle β**: Recall the triangulation process for locating point M (see Fig. 6.9(b)). As point N_R moves along line $\overline{P_{\perp 1}C_{\perp 1}}$, angle β will change accordingly while angle α and the positions of points C and

P will not be affected. From the triangle $\triangle CPM$, we have

$$|CM| = \frac{\sin\beta |CP|}{\sin(\alpha+\beta)} \tag{6.25}$$

Taking the partial derivatives w.r.t. angle β on both sides of the equation and rearranging the terms, the following equation can be obtained:

$$\frac{\partial |CM|}{\partial \beta} = \frac{\sin\alpha |CP|}{\sin^2(\alpha+\beta)} \tag{6.26}$$

which describes the relationship between a small change in angle β and the corresponding change in $|CM|$.

- **Relationship between angle β and angle γ**: Let γ denote the angle between line $\overline{CM_I}$ and line $\overline{CN_I}$, as shown in Fig. 6.9. Let $\psi_{\beta 1}$ and $\psi_{\alpha 1}$ denote the angles $\angle N_R P P_{\perp 1}$ and $\angle N_R C C_{\perp 1}$ respectively. By using planar geometry, the following equation can be obtained, which describes the relationship between a small change in angle γ and the corresponding change in angle β:

$$\frac{\partial \beta}{\partial \gamma} = \frac{|CC_{\perp 1}| \cos^2\psi_{\beta 1}}{|PP_{\perp 1}| \cos^2\psi_{\alpha 1}} \tag{6.27}$$

In the above equation, dγ (the change of angle γ) is defined to be positive if as a result point N_I moves towards point P_I. P_I is the projection of point P on the camera's image plane. A detailed derivation of Eqn. 6.27 is presented in Appendix C.5.

- **Relationship between angle γ and $|M_I N_I|_+$**: Let ρ denote the angle between line $\overline{M_I C}$ and $\overline{M_I N_I}$ (see Fig. 6.9(a)). From the triangle $\triangle C M_I N_I$, we have

$$\frac{\sin\gamma}{\sin(\pi-\gamma-\rho)} = \frac{|M_I N_I|_+}{|CM_I|} \tag{6.28}$$

where $|M_I N_I|_+$ is the signed distance between points M_I and N_I. The definition of $|M_I N_I|_+$ is the following:

$$|M_I N_I|_+ = \overrightarrow{M_I N_I} \cdot \frac{\overrightarrow{M_I P_I}}{|M_I P_I|} \tag{6.29}$$

Based on this definition, the sign of $\mathrm{d}|M_I N_I|_+$ is consistent with the sign of $\mathrm{d}\gamma$, i.e. $\mathrm{d}|M_I N_I|_+$ is positive if as a result N_I moves towards P_I. Taking the partial derivatives w.r.t. $|M_I N_I|_+$ on both sides of Eqn. 6.28 and rearranging the terms, the following equation can be obtained:

$$\frac{\partial \gamma}{\partial |M_I N_I|_+} = \frac{\sin^2(\gamma + \rho)}{\sin \rho |C M_I|} \tag{6.30}$$

- **Relationship between $|CM|$ and $|M_I N_I|_+$**: Since point N_I is restrained on line $\overline{P_I M_I}$, its position can be described solely using $|M_I N_I|_+$, whose definition is given in Eqn. 6.29. Combining Eqn. 6.26, 6.27 and 6.30, the relationship between a change in the position of N_I and the corresponding change in the calculated depth, $|CM|$, can be expressed using the following equation:

$$\frac{\partial |CM|}{\partial |M_I N_I|_+} = \frac{\sin \alpha \sin^2(\gamma + \rho) \cos^2 \psi_{\beta 1} |CP||CC_{\perp 1}|}{\sin^2(\alpha + \beta) \sin \rho \cos^2 \psi_{\alpha 1} |CM_I||PP_{\perp 1}|} \tag{6.31}$$

The uncertainty of $|M_I N_I|_+$, which is caused by the phase uncertainties, can be estimated in the following manner:

- **Relationship between $|M_I N_I|_+$ and a phase error**: Assume that there is an error in $\Phi(M_I)$, denoted by $\mathrm{d}\Phi$, and as a result the calculated position of N_I moves to N_I', as shown in Fig. 6.10. The movement of point N_I can be

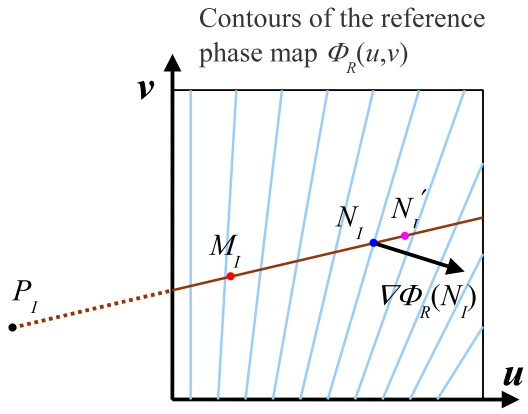


Figure 6.10: Influence of phase errors to the position of point N_I

expressed in terms of $\mathrm{d}\Phi$ using the following equation:

$$\mathrm{d}|M_I N_I|_+ = \frac{|M_I P_I|}{\overrightarrow{M_I P_I} \cdot \nabla\Phi_R(N_I)}\,\mathrm{d}\Phi \tag{6.32}$$

where $\nabla\Phi_R(N_I)$ is the phase gradient in the reference phase map $\Phi_R(u,v)$ at point N_I.

- **Estimation of the uncertainty of** $|M_I N_I|_+$: Let $\sigma_\Phi(M_I)$ denote the uncertainty of $\Phi(M_I)$, and let $\sigma_{\Phi_R}(N_I)$ denote the average uncertainty of the phase values in the close neighborhood of point N_I in the reference phase map. The corresponding uncertainty in $|M_I N_I|_+$, denoted by $\sigma_{|M_I N_I|}$, can be calculated using the following equation:

$$\sigma_{|M_I N_I|} = \frac{|M_I P_I|\sqrt{\sigma_\Phi^2(M) + \sigma_{\Phi_R}^2(N_I)}}{\left|\overrightarrow{M_I P_I} \cdot \nabla\Phi_R(N_I)\right|} \tag{6.33}$$

6.5.3.3 Influence of Uncertainties of the Position and Orientation of the Reference Plane

In the computation of the depth of point M, the position and orientation of the reference plane are used to locate point N_R along line $\overline{CN_I}$ (see Fig. 6.8). Errors in these two components would cause N_R to slide along $\overline{CN_I}$, and in turn cause an error in the calculated depth of M.

In the uncertainty model, the position and orientation of the reference plane are represented by $(O_R, \boldsymbol{n}_R)$, where O_R is a point on the plane and $\boldsymbol{n}_R$ is the normal direction of the plane. The uncertainties of the position and orientation are represented by $(\sigma_{O_R}, \sigma_{\boldsymbol{n}_R})$. σ_{O_R} is the uncertainty of the position of O_R along the direction of $\boldsymbol{n}_R$. $\sigma_{\boldsymbol{n}_R}$ is defined as the standard deviation of the angle between the estimated $\boldsymbol{n}_R$ and the true $\boldsymbol{n}_R$. Both σ_{O_R} and $\sigma_{\boldsymbol{n}_R}$ are known from system calibration.

To study the influence of $(\sigma_{O_R}, \sigma_{\boldsymbol{n}_R})$ to the uncertainty of depth, the relationship between an error in $|CN_R|$ and the corresponding error in $|CM|$ needs to be known. This relationship can be derived as follows:

- **Relationship between angle β and $|CN_R|$**: Recall the triangulation process for locating point M (see Fig. 6.11). Let point $P_{\perp 2}$ be the projection of point P on line $\overline{CN_I}$. Let $\Phi_{\beta 2}$ be the angle between line $\overline{PP_{\perp 2}}$ and line $\overline{PN_R}$. Assume that, due to errors in $(O_R, \boldsymbol{n}_R)$, point N_R moves to N_R' and accordingly point M moves to M'. From the right triangle $\triangle N_R P P_{\perp 2}$, the following equation can be obtained:

$$\tan \psi_{\beta 2} = \frac{|N_R P_{\perp 2}|}{|P P_{\perp 2}|} \tag{6.34}$$

  Differentiating the above equation, we have

$$\frac{\mathrm{d}\psi_{\beta 2}}{\mathrm{d}|N_R P_{\perp 2}|} = \frac{\cos^2 \psi_{\beta 2}}{|P P_{\perp 2}|} \tag{6.35}$$

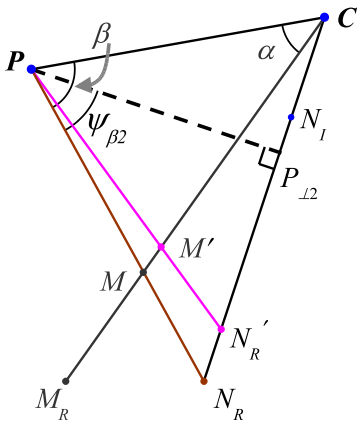


Figure 6.11: Relationship between $|CN_R|$ and angle β

Notice that $\mathrm{d}\beta = \pm\mathrm{d}\psi_{\beta 2}$ and $\mathrm{d}|CN_R| = \pm\mathrm{d}|N_R P_{\perp 2}|$, where the signs depend on whether N_R is in the middle of line segment $\overline{CP_{\perp 2}}$ or on the extensions. The relationship between a small change in $|CN_R|$ and the corresponding change in β can be expressed using the following equation:

$$\frac{\partial \beta}{\partial |CN_R|} = \frac{\cos^2 \psi_{\beta 2}}{|PP_{\perp 2}|} \tag{6.36}$$

- **Relationship between $|CM|$ and $|CN_R|$**: Combining Eqn. 6.26 and 6.36, the relationship between a change in $|CN_R|$ and the corresponding change in the depth, $|CM|$, can be expressed using the following equation:

$$\frac{\partial |CM|}{\partial |CN_R|} = \frac{\sin\alpha \cos^2 \psi_{\beta 2} |CP|}{\sin^2(\alpha+\beta)|PP_{\perp 2}|} \tag{6.37}$$

To study the influence of $\sigma_{\boldsymbol{n}_R}$ to the uncertainty of depth, the relationship between an error in $\boldsymbol{n}_R$ and the corresponding error in $|CN_R|$ needs to be known. Let $\boldsymbol{n}'_R$ denote the erroneous value of $\boldsymbol{n}_R$ (see Fig. 6.12). Let ϑ_R denote the angle between $\boldsymbol{n}_R$ and the projection of $\boldsymbol{n}'_R$ on the plane that is defined by $\overline{O_R N_R}$ and $\boldsymbol{n}_R$. The relationship between $|CN_R|$ and ϑ_R (without the consideration of signs) can be

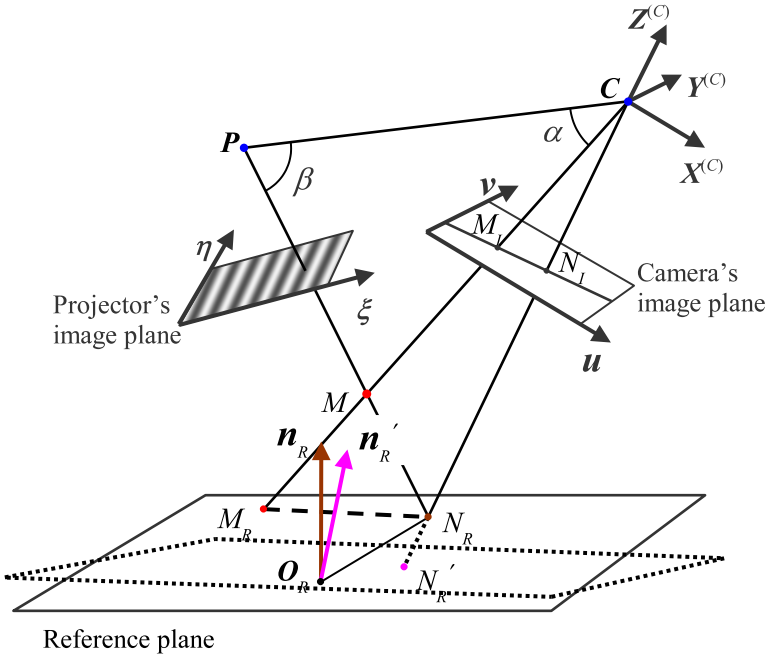


Figure 6.12: Relationship between $|CN_R|$ and ϑ_R

expressed using the following equation:

$$\frac{\partial |CN_R|}{\partial \vartheta_R} = \frac{|O_R N_R||CN_I|}{\left| \boldsymbol{n}_R \cdot \overrightarrow{CN_I} \right|} \tag{6.38}$$

Combining Eqn. 6.37 and 6.38, the final equation for describing the relationship between $|CM|$ and ϑ_R can be written as follows:

$$\frac{\partial |CM|}{\partial \vartheta_R} = \frac{\sin\alpha \cos^2\psi_{\beta 2}|CP||O_R N_R||CN_I|}{\sin^2(\alpha+\beta)|PP_{\perp 2}| \left| \boldsymbol{n}_R \cdot \overrightarrow{CN_I} \right|} \tag{6.39}$$

6.5.3.4 Influence of Uncertainty of the Projector's Projection Center

The uncertainty in the position of the projector's projection center (point P) has a complex influence on the uncertainty of depth. Assume that the estimated position

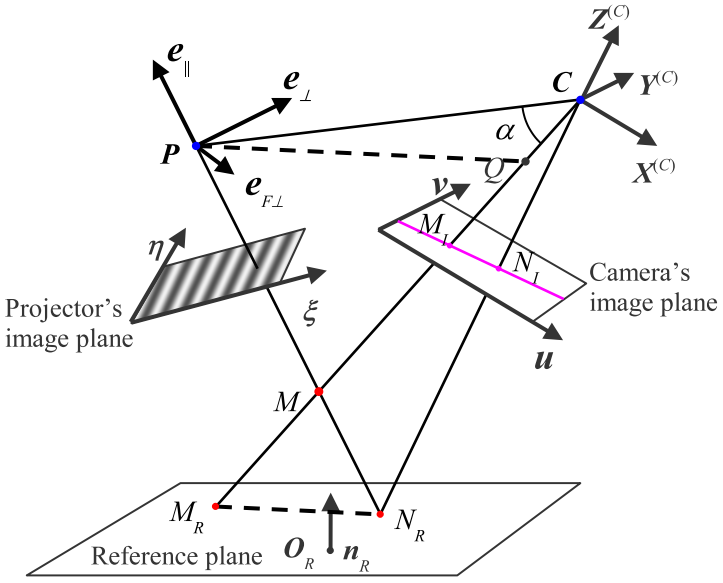


Figure 6.13: Decomposition of the position error of point P

of point P has an error of $\boldsymbol{\delta}P$, i.e. $\boldsymbol{\delta}P = P' - P$, where P' is the estimated position and P is the true position. $\boldsymbol{\delta}P$ can be decomposed along three orthogonal directions: $\boldsymbol{e}_{\parallel}$, $\boldsymbol{e}_{\perp}$ and $\boldsymbol{e}_{F\perp}$, as shown in Fig. 6.13. $\boldsymbol{e}_{\parallel}$ is along the direction of line $\overline{PN_R}$. $\boldsymbol{e}_{\perp}$ lies in plane CPN_R and is perpendicular to line $\overline{PN_R}$. $\boldsymbol{e}_{F\perp}$ is perpendicular to plane CPN_R. Let $\delta P_{\parallel}$, $\delta P_{\perp}$ and $\delta P_{F\perp}$ denote the three components of $\boldsymbol{\delta}P$ along vectors $\boldsymbol{e}_{\parallel}$, $\boldsymbol{e}_{\perp}$ and $\boldsymbol{e}_{F\perp}$, respectively. The influence of $\boldsymbol{\delta}P$ on $|CM|$ can be studied through these components individually. A detailed analysis is given below.

- **Relationship between $|CM|$ and $\delta P_{\parallel}$**: Recall that point M is located by the intersection of line $\overline{CM_R}$ and line $\overline{PN_R}$. The movement of point P along line $\overline{PN_R}$ would not change the position of M (see Fig. 6.13). Hence, $\delta P_{\parallel}$ does not affect $|CM|$.

- **Relationship between $|CM|$ and $\delta P_{\perp}$**: The relationship between $\delta P_{\perp}$ and $|CM|$ is illustrated in Fig. 6.14. As a result of $\delta P_{\perp}$, point P moves along $\boldsymbol{e}_{\perp}$ to P'. Accordingly, point M moves along $\overline{CM_I}$ to M'. By using planar geometry,

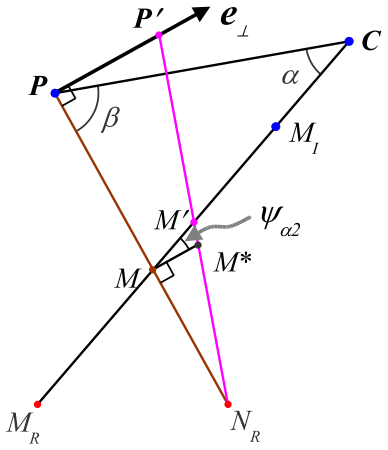


Figure 6.14: Relationship between $|CM|$ and $\delta P_\perp$

the following equation can be derived to describe the relationship between $\delta P_\perp$ and the corresponding change in $|CM|$:

$$\frac{\partial |CM|}{\partial (\delta P_\perp)} = \frac{|N_R M|}{\sin(\alpha+\beta)|N_R P|} \operatorname{sign}\left(\overrightarrow{N_R M} \cdot \overrightarrow{N_R P}\right) \operatorname{sign}\left(\overrightarrow{CP} \cdot \boldsymbol{e}_\perp\right) \tag{6.40}$$

where function sign() extracts the signs of variables, i.e. $\operatorname{sign}(x) = x/|x|$. A detailed derivation of the above equation is given in Appendix C.6.

- **Relationship between $|CM|$ and $\delta P_{F\perp}$**: As can be seen from Fig. 6.13, since $\boldsymbol{e}_{F\perp}$ is perpendicular to plane CPM_I, a small movement of P along $\boldsymbol{e}_{F\perp}$ would not change the distance $|CP|$ or angle α. Instead, it causes plane CPM_I to rotate around axis $\overline{CM_I}$, and this rotation in turn causes point N_R to move inside plane CPM_I (since N_R needs to remain on the reference plane). From the perspective of triangulation (see Fig. 6.8), the movement of point P along $\boldsymbol{e}_{F\perp}$ changes the position of N_R only. The positions of points C, P and M_R remain the same.

  The movement of point N_R can be further decomposed into two parts. Firstly,

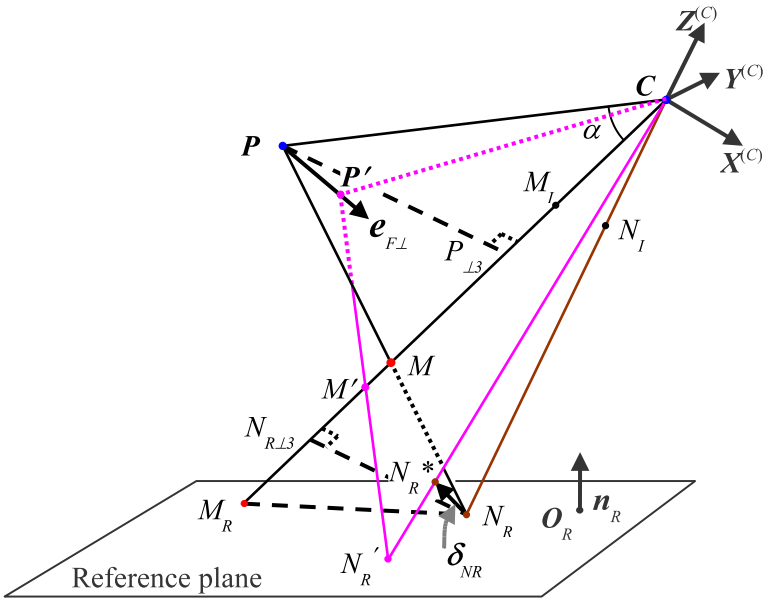


Figure 6.15: Relationship between $|CM|$ and $\delta P_{F\perp}$

the rotation of plane CPM_R causes vector $\overrightarrow{CN_I}$ to rotate around axis $\overline{CM_I}$ while the angle between $\overline{CM_I}$ and $\overline{CN_I}$ remains unaffected (see Fig. 6.15). The rotation of $\overrightarrow{CN_I}$ will change the length of $|CN_R|$ since in general $\overline{CN_R}$ is not strictly perpendicular to the reference plane. Secondly, the movement of P also changes the position of point P_I (P_I is the projection of P on the camera's image plane), and as a result the position of line $\overline{P_I M_I}$ in the image plane will change. Recall that the position of point N_I is located by searching along line $\overline{P_I M_I}$ for $\Phi_R(N_I) = \Phi(M)$ (see Section 6.5.3.1). Therefore, for different search path $\overline{P_I M_I}$, the resulting position of N_I is different and hence $|M_I N_I|$ may be different.

For the first part, the relationship between $\delta P_{F\perp}$ and the corresponding change in $|CN_R|$ can be expressed using the following equation (see Appendix C.7 for

a detailed derivation):

$$\frac{\partial|CN_R|}{\partial(\delta P_{F\perp})} = -\frac{(\boldsymbol{e}_{F\perp}\cdot\boldsymbol{n}_R)|CN_R||N_RM|}{(\overrightarrow{CN_R}\cdot\boldsymbol{n}_R)|PM|}\,\mathrm{sign}\Big(\overrightarrow{N_RM}\cdot\overrightarrow{PM}\Big) \tag{6.41}$$

For most SMDFP systems, the possible positions of plane CPM_I are nearly perpendicular to the reference plane, in which case the value of $\boldsymbol{e}_{F\perp}\cdot\boldsymbol{n}_R$ is small. As a result, the corresponding change in $|CN_R|$ is small when compared to influences from other uncertainty components, e.g. uncertainties in the position and orientation of the reference plane.

For the second part, the error in depth it may cause is in general one order of magnitude smaller than the errors caused by other components. Hence, it can also be ignored. A detailed proof is given in Appendix C.8.

As a summary of the above, the only component needs to be considered in the uncertainty of the position of point P is $\delta P_\perp$.

6.5.3.5 Overall Uncertainty in Depth

The major components that cause the uncertainty of depth, namely the phase uncertainties, the uncertainties in the position and orientation of the reference plane, and the position uncertainty of the projector's projection center, are uncorrelated. Therefore, the overall depth uncertainty can be estimated from these components by using the Law of Variance Propagation. The final equation for calculating the depth uncertainty of a point M is the following:

$$\begin{aligned}\big(D_U(M)\big)^2 \approx &\left(\frac{\partial|CM|}{\partial|M_IN_I|_+}\right)^2\sigma^2_{|M_IN_I|} + \left(\frac{\partial|CM|}{\partial|CN_R|}\right)^2\sigma^2_{O_R} + \\ &\left(\frac{\partial|CM|}{\partial\vartheta_R}\right)^2\sigma^2_{\boldsymbol{n}_R} + \left(\frac{\partial|CM|}{\partial(\delta P_\perp)}\right)^2\sigma^2_P\end{aligned} \tag{6.42}$$

where $D_U(M)$ is the depth uncertainty of point M; $\sigma_{|M_I N_I|}$ is the uncertainty of $|M_I N_I|_+$, which can be calculated from Eqn 6.33; σ_{O_R}, $\sigma_{\boldsymbol{n}_R}$ and σ_P are the uncertainty components as defined in Section 6.4.1; and the partial derivatives of $|CM|$ are given in Eqns 6.31, 6.37, 6.39 and 6.40, respectively.

6.6 Implementation and Results

6.6.1 Implementation

The proposed uncertainty model and the related algorithms were implemented in C++. The functionality of uncertainty estimation was integrated into the shape measurement software we have developed earlier (see Section 3.4.1 for details). At the end of each measurement, the software gives two outputs: 1) a point cloud that represents the surface being measured; and 2) the uncertainties in the point cloud generated. For each point (say point M) in the point cloud, the uncertainty in the position is given by a cylindrical region, which is defined by M (center position), $\boldsymbol{V}_U(M)$ (axis), $r_U(M)$ (radius) and $D_U(M)$ (half height). When assuming all the random error sources have normal distributions, the probability of the true position of M falling within the uncertainty region is approximately 68%.

A popularly used method for representing the accuracy of surface measurement is to give the distance of every measured point to the true surface position. The cylindrical-region representation of measurement uncertainties can be approximated to this distance representation by using the following equation:

$$\sigma_{ds} \approx \sqrt{(\boldsymbol{V}_U \cdot \boldsymbol{n}_S)^2 \, D_U^2 + \frac{1}{2}\left[1 - (\boldsymbol{V}_U \cdot \boldsymbol{n}_S)^2\right] r_U^2} \tag{6.43}$$

where σ_{ds} is the estimated uncertainty of the distance from a measured point to the true surface position; $\boldsymbol{V}_U$, r_U and D_U are the attributes of the cylindrical uncertainty

region; and $\boldsymbol{n}_S$ is the normal direction of the local surface, which can be calculated approximated from the point cloud obtained in the measurement.

6.6.2 Estimation of the Uncertainties in a Physical Measurement

To verify the soundness of the uncertainty model and algorithms developed, a gauge part was measured by using the SMDFP system we have built (see Section 3.4.1 for details). The estimated measurement uncertainties given by the shape measurement software were compared to the analysis results on the measurement errors, which were given by a commercial reverse engineering software [69].

The gauge part used in the measurement is the same one as described in Section 3.4.3.1. It is an aluminum part which has a curved cone shape with a dimension of $128mm$ (L) $\times$ $128mm$ (W) $\times$ $151mm$ (H). Figure 6.16(a) shows a picture of the part. The part was machined very accurately to a predefined CAD model. According to a CMM measurement of the part, the maximum deviation of the part from the CAD model is $35\mu m$ and the standard deviation is $11\mu m$. Before measurement, the part was applied a paint coating to avoid the specular reflection from the surface.

The measurement errors were evaluated by comparing the point cloud obtained from the measurement to the CAD model of the part (see Fig. 6.16(c)). The minimum distance from each point (in the point cloud) to the surface of the CAD model was calculated and taken as the measurement error of that point. This analysis was done by using the reverse engineering software. The result is shown in Fig. 6.16(b). The RMS value of the measurement errors is $94\mu m$.

The distribution of the estimated measurement uncertainties, in terms of point-to-surface distances, is shown in Fig. 6.16(d). The estimated uncertainties have a similar distribution pattern as the evaluated measurement errors, i.e. the central area of the

surface measured has smaller measurement errors and the surrounding area has larger errors. The RMS value of the estimated measurement uncertainties is $72\mu m$.

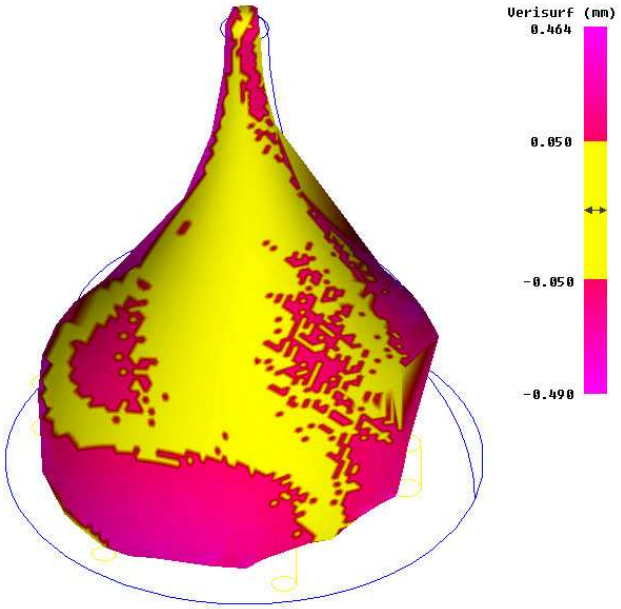
6.7 Summary

It is a desired feature for SMDFP systems to give the estimated uncertainties of measurement along with a measurement result. In most commercial SMFP/SMDFP systems, measurement uncertainty is given as a fixed specification, which may not correctly reflect the actual error magnitude of a particular measurement, since the magnitudes of uncertainties vary across the measurement volume and can be influenced notably by factors that are measurement-related.

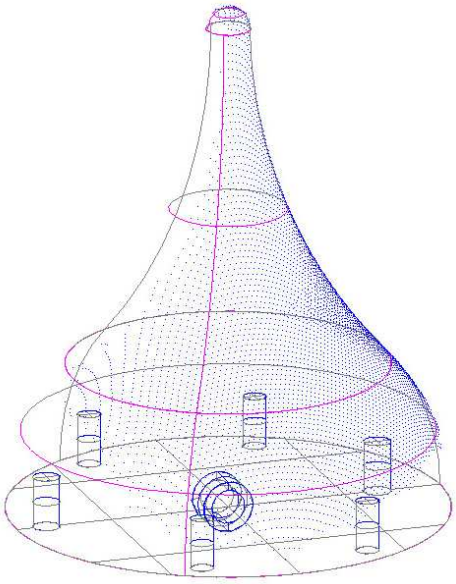
In this chapter, a comprehensive uncertainty model is proposed to describe the relations between the many error sources and the resulting uncertainties in a measurement. Based on this model, the uncertainties can be estimated by using the image data acquired in measurement and the apriori knowledge on other fixed error sources. The result obtained by this approach provides a good estimation of the uncertainties of a measurement.

The uncertainty model and the related algorithms were implemented in the shape measurement software we have developed. A gauge part was measured and the measurement errors were evaluated by using a commercial reverse engineering software. The estimated measurement uncertainties given by our software match with the results of measurement errors, in terms of distribution pattern and overall magnitude.

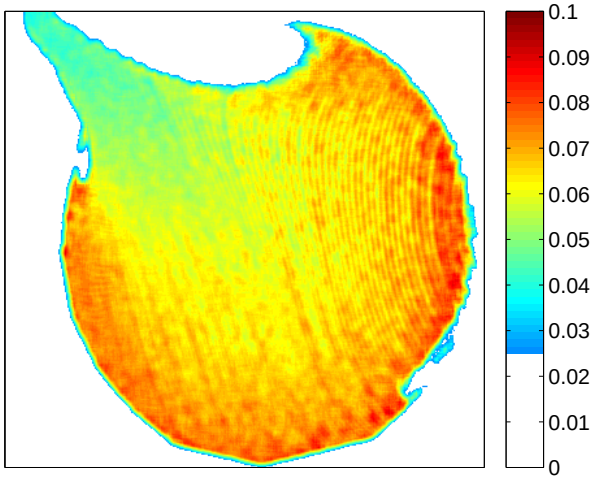
(a) Photograph of the gauge part



(b) Distribution of the measurement errors evaluated by a reverse engineering software



(c) The CAD model of the part and the point cloud acquired from measurement



(d) Distribution of the estimated measurement uncertainties

Figure 6.16: Estimated uncertainties of the measurement of a gauge part

Chapter 7

Conclusions

This chapter is organized in the following manner. Section 7.1 describes the main intellectual contributions of this dissertation. Section 7.2 identifies the anticipated industrial benefits resulting from this research. Section 7.3 discusses the future research directions.

7.1 Intellectual Contributions

This dissertation makes intellectual contributions in the following areas:

- **Mathematical model and shape measurement algorithms for SMDFP**: This dissertation presents a new mathematical model for SMDFP which gives an accurate modeling of the optical geometry of SMDFP systems. This model provides a mathematical basis that is required to achieve high-accuracy shape measurements. Based on this model, three related algorithms were developed, namely the algorithm for construction of absolute phase map, algorithm for construction of point cloud and algorithm for estimation of sensor parameters. By fully exploiting the information in the images obtained from phase-shifting, the algorithm for phase map construction uses fewer projection patterns than

existing algorithms without compromising performance. As a result, the measurement speed of SMDFP systems is improved. With the use of a reference phase map, the point cloud construction algorithm presented demands fewer sensor parameters than similar algorithms in its class and handles the lens distortions of projector automatically. As a consequence, the calibration process of SMDFP systems is simplified and the measurement accuracy can be potentially improved. We have shown that these algorithms work satisfactorily on a wide variety of parts.

- **Framework for using adaptive projection patterns in SMDFP**:
  This dissertation presents a framework for using adaptive projection patterns in SMDFP technique. Detailed issues such as the measurement procedure, prospected applications and the requirements on system model and algorithms, are discussed. A new algorithm for automatically generating adaptive patterns for measurements, was developed. Based on a number of tests, the use of adaptive projection patterns have shown to provide better overall performance, in terms of measurement accuracy and coverage, than using fixed-pitch fringe patterns, especially for measuring surfaces with a large dynamic range of normal directions.

- **Model and algorithms for estimation of measurement uncertainties**:
  This dissertation presents a comprehensive uncertainty model for describing the relations between the many error sources and the resulting uncertainties in shape measurements conducted using SMDFP. Based on the model and algorithms developed, the measurement uncertainties can be estimated from the image data acquired in a measurement as well as the knowledge on fixed error sources. We have shown that the measurement uncertainties computed by our approach match with the real measurement errors.

7.2 Anticipated Impacts on Industrial Practices

As a promising technique for 3-D shape measurements, SMDFP has been increasingly used in many industrial applications, e.g. mechanical reverse engineering, 3-D digital replication, parts inspection, and quality control. The research results reported in this dissertation can be used to improve the performance and features of existing SMDFP systems to ensure their wide spread use in industry. Specific improvements that can be made and the resulting benefits are outlined below:

- *Better measurement accuracy.* The mathematical model and shape measurement algorithms presented in this dissertation provides an accurate modeling of SMDFP systems. With an accurate calibration, the resulting SMDFP systems are able to achieve high measurement accuracy which could satisfy the requirements of many industrial applications.

- *Easier system calibration.* The model and shape measurement algorithms presented in this dissertation simplifies the calibration procedure of SMDFP systems. This could result in time saving and better user experience when system calibrations are required during usage, such as due to changes in system configuration and environmental conditions.

- *Faster measurement speed.* With the new algorithms for shape measurement and the use of adaptive projection pattern, the measurement speed of SMDFP systems can be increased. As a result, SMDFP technique could meet the speed requirements of more applications, such as 100% on-line parts inspection.

- *Better estimation of measurement uncertainties.* The uncertainty model presented in this dissertation is able to deliver better estimation of measurement uncertainties in a measurement-specific manner. The uncertainties in shape measurements carried out using SMDFP systems vary according to the parts

being measured and the environmental conditions. In applications such as parts inspection and quality control, various 3D features on a part need to be measured and analyzed, and judgments need to be made on whether a feature is within the tolerance specifications. A good estimation of measurement uncertainties can ensure the validity of such processes.

- *Lower system cost.* The research work conducted in this dissertation is targeted at SMDFP systems with low-cost system configurations, e.g. using commercial computer projectors and low-end industrial digital cameras. The hardware cost of such systems is around $3,000 (based on the market prices of the components at the time of writing). The low cost of the systems could ensure the wide deployment of this technique in industry.

7.3 Future Work

Following the research work described in this dissertation, new research can be conducted in the following areas:

1. *Improvement on the calibration procedure of SMDFP systems*: The calibration procedure presented in this dissertation uses a flat plate with a checkerboard pattern for calibrating the camera. However, a high-accuracy calibration plate like this is difficult to manufacture. Therefore, it is desired to seek for other calibration artifacts which are easier to make. As a result, the calibration process may require modifications.

2. *Improvement on the algorithm for automated generation of adaptive projection patterns*: The algorithm presented for generation of adaptive patterns works well and the generated patterns gave better overall performance than fixed-pitch fringe patterns. However, there is still room for improvement in this

algorithm to achieve better performance. An in-depth understanding of the relationship between the fringe pitch in projection pattern and the resulting measurement performance needs to be established, as well as the influence of other measurement factors need to be studied.

3. *Characterizing influence of system parameters on measurement uncertainties*: Based on the uncertainty model presented in this dissertation, a better understanding of the influence of system parameters on measurement uncertainties can be established. SMDFP systems can be configured within a large design space, e.g. the relative position and orientation between the projector and the camera, the specifications of the projector and the camera, the selection of calibration methods, etc. Different configurations of the system will result in different measurement accuracies. The characterization of the influence of system parameters on measurement uncertainties can be used in determining the system configuration to achieve specific design goals.

Appendix A

Proofs for System Model and Algorithms

A.1 Relationship between the phase-shifted projection patterns and the corresponding images

In a phase-shifting process, a set of phase-shifted fringe patterns are projected on an object and the corresponding images are captured. For each pixel in the camera's imaging sensor, the light intensity received depends on many factors, e.g. the intensity of light projected on the surface, the surface normal direction and optical property, the intensity of environmental light, etc. A phase-shifting process takes only a few seconds to finish, and during the process the positions of the camera, projector and object are fixed. Therefore, most of the factors mentioned above can be considered as constants w.r.t. time. The only variable is the light intensity projected on the surface. For most opaque materials, the intensity of reflected light is proportional to the intensity of incident light. As a result, the relationship between the light projection intensity and the image intensity is linear.

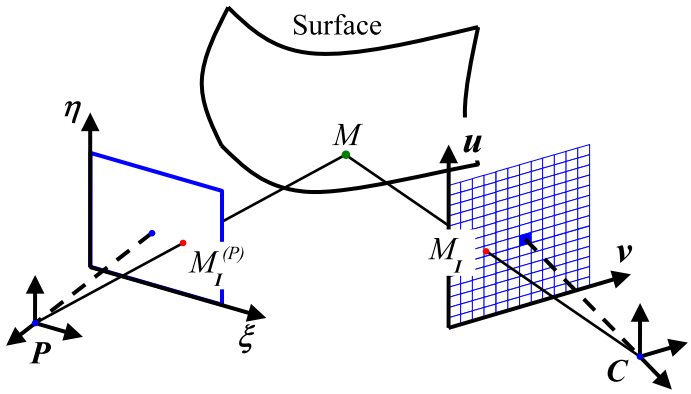


Figure A.1: Relationship between the light projection intensity and the image intensity

Consider a point M on the surface, as shown in Fig. A.1. Point M_I is the image of M in the camera and (i, j) are its pixel coordinates. According to the pinhole camera model, the light intensity received at M_I, and hence the image intensity of pixel (i, j) [1], is determined by the light intensity at point M. Similarly, $M_I^{(P)}$ is the projection of M on the image plane of the projector, and the light intensity at M is determined by the projection intensity at $M_I^{(P)}$. Let $I(i, j)$ denote the image intensity of pixel (i, j) and let $I_{i,j}^{(P)}$ denote the projection intensity at $M_I^{(P)}$. The relationship between $I(i, j)$ and $I_{i,j}^{(P)}$ can be described using the following equation:

$$I(i, j) = A'(i, j) + B'(i, j)\, I_{i,j}^{(P)} \tag{A.1}$$

where $A'(i, j)$ and $B'(i, j)$ are constants. Notice that the subscripts i and j of $I^{(P)}$ are for correspondence to the image pixel (i, j). They are not coordinates in the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane of the projector.

When a set of phase-shifted fringe patterns are projected, the intensity of any

[1] To be precise, each pixel in the camera's image plane corresponds to a tiny area on the surface, not a single point. However it can be considered as a single point in the context and will not change the conclusion

point in the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane varies sinusoidally. Take the N-step phase-shifting with uniform phase shifts as an example: For a point (ξ, η) in the $\boldsymbol{\xi}$-$\boldsymbol{\eta}$ plane, the corresponding projection intensity sequence, which consists of a number of N values, can be described using the following equation:

$$I_n^{(P)}(\xi,\eta) = \frac{I_{max}^{(P)}}{2}\left[1+\sin\left(\Phi^{(P)}(\xi,\eta)+\frac{2\pi(n-1)}{N}\right)\right], \quad n=1,\cdots,N \tag{A.2}$$

where $I_n^{(P)}(\xi,\eta)$ is the n-th projection intensity at (ξ,η), $I_{max}^{(P)}$ is the maximum projection intensity in the pattern and $\Phi^{(P)}(\xi,\eta)$ is the corresponding phase value at (ξ,η).

Assume that (ξ_1,η_1) are the coordinates of point $M_I^{(P)}$. When projecting the phase-shifted patterns as defined in Eqn. A.2, the corresponding image intensity sequence received at point M_I can be written as the following:

$$I_n(i,j) = A'(i,j) + B'(i,j)\, I_n^{(P)}(\xi_1,\eta_1), \quad n=1,\cdots,N \tag{A.3}$$

Substituting Eqn. A.2 in the equation above, we get

$$I_n(i,j) = \left[A'(i,j)+\frac{I_{max}^{(P)}\,B'(i,j)}{2}\right] + \frac{I_{max}^{(P)}\,B'(i,j)}{2}\,\sin\left(\Phi^{(P)}(\xi_1,\eta_1)+\frac{2\pi(n-1)}{N}\right) \tag{A.4}$$

which can be rewritten as the following by defining new constants $A(i,j)$ and $B(i,j)$:

$$I_n(i,j) = A(i,j) + B(i,j)\,\sin\left(\Phi^{(P)}(\xi_1,\eta_1)+\frac{2\pi(n-1)}{N}\right) \tag{A.5}$$

That is, the N intensity values associated with pixel (i,j) comply with a sinusoidal variation.

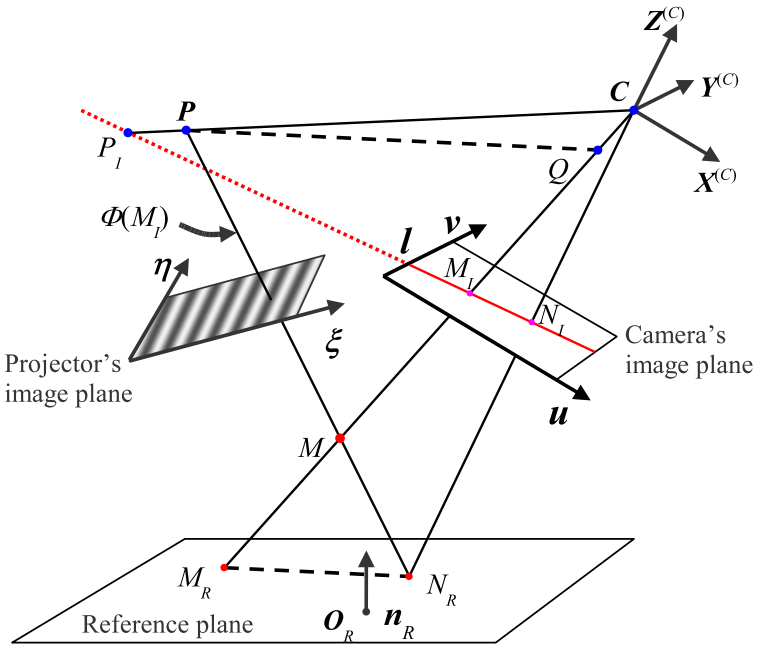


Figure A.2: Computing the 3-D coordinates of a point M from its absolute phase value

A.2 Computing the 3-D coordinates of a point from its absolute phase value

Notations and definitions

An illustration of the geometric elements involved in the computation is shown in Fig. A.2. $X^{(C)}Y^{(C)}Z^{(C)}$ is the camera coordinate frame and point C is the projection center of the camera. Plane $\boldsymbol{u}$-$\boldsymbol{v}$ is the image plane of the camera. Point P is the projection center of the projector. The reference plane passes through point O_R and has a normal vector of $\boldsymbol{n}_R$. M is the point on the object's surface that corresponds to pixel (u, v). All spatial coordinates referred in this section are with respect to the camera's coordinate frame if not otherwise declared.

Let point M_I denote the idealized position of the center of pixel (u, v) on the image plane of the camera. It can be known from the pinhole camera model that

point M lies on line $\overline{CM_I}$. Let point M_R denote the intersection of line $\overline{CM_I}$ with the reference plane. Similarly, let point N_R denote the intersection of line $\overline{PM}$ with the reference plane and let N_I denote the intersection of line $\overline{CN_R}$ with the image plane of the camera. Define point Q as the intersection of line $\overline{CM_I}$ with the plane that is parallel to the reference plane while passes through point P. It can be seen from the definitions that, points C, P, Q, M_I, N_I, M_R, N_R and M are coplanar.

Proof of the algorithm for locating N_I

Since point N_R is the intersection of line $\overline{PM}$ with the reference plane, we can get that $\Phi_R(N_I) = \Phi(M_I)$, which means the phase value of point N_R in the reference phase map is equal to the phase value of point M in the object's phase map. On the other hand, since points P, M and N_R are collinear in the 3-D space, their projections on the camera's image plane, $\boldsymbol{u}$–$\boldsymbol{v}$, are also collinear. In other words, points P_I, M_I and N_I are on the same line $\boldsymbol{l}$ (P_I is the projection of point P on plane $\boldsymbol{u}$–$\boldsymbol{v}$). Hence, the position of N_I can be located by searching in the reference phase map Φ_R along line $\overline{P_I M_I}$ for a point with phase value $\Phi(M_I)$.

Proof of Equation 3.23 for computing M_R

Since point M_R is on the reference plane, we have

$$\overrightarrow{CO_R} \cdot \mathbf{n}_R = \overrightarrow{CM_R} \cdot \mathbf{n}_R = \left|\overrightarrow{CM_R}\right| \frac{\overrightarrow{CM_I}}{\left|\overrightarrow{CM_I}\right|} \cdot \mathbf{n}_R \tag{A.6}$$

Rearranging the terms in the above equation, we can get

$$\left|\overrightarrow{CM_R}\right| = \left|\overrightarrow{CM_I}\right| \frac{\overrightarrow{CO_R} \cdot \mathbf{n}_R}{\overrightarrow{CM_I} \cdot \mathbf{n}_R} \tag{A.7}$$

Therefore the position of M_R can be acquired from the following equation:

$$M_R = C + \left|\overrightarrow{CM_R}\right| \frac{\overrightarrow{CM_I}}{\left|\overrightarrow{CM_I}\right|} = C + \left(\frac{\overrightarrow{CO_R} \cdot \mathbf{n}_R}{\overrightarrow{CM_I} \cdot \mathbf{n}_R}\right) \overrightarrow{CM_I} \tag{A.8}$$

Proof of Equation 3.25 for computing M

It is known from the definition of point Q that, line $\overline{PQ}$ and line $\overline{M_R N_R}$ are parallel. It is also known from the definitions that, points P, Q, M_R, N_R and M are coplanar. Utilizing these relations, the following equation can be obtained from similar triangles $\triangle MPQ$ and $\triangle MN_RM_R$:

$$\frac{\left|\overrightarrow{QM}\right|}{\left|\overrightarrow{QM_R}\right|} = \frac{\left|\overrightarrow{PQ}\right|}{\left|\overrightarrow{PQ} + \overrightarrow{M_R N_R}\right|} \tag{A.9}$$

Rearranging the terms in the above equation, we have

$$\left|\overrightarrow{QM}\right| = \frac{\left|\overrightarrow{PQ}\right| \left|\overrightarrow{QM_R}\right|}{\left|\overrightarrow{PQ} + \overrightarrow{M_R N_R}\right|} \tag{A.10}$$

The equation for computing the position of M can then be derived as follows:

$$M = Q + \left|\overrightarrow{QM}\right| \frac{\overrightarrow{CM_I}}{\left|\overrightarrow{CM_I}\right|} = Q + \left(\frac{\left|\overrightarrow{PQ}\right| \left|\overrightarrow{QM_R}\right|}{\left|\overrightarrow{PQ} + \overrightarrow{M_R N_R}\right| \left|\overrightarrow{CM_I}\right|}\right) \overrightarrow{CM_I} \tag{A.11}$$

Appendix B

Proofs for Adaptive Projection Patterns

B.1 Influence of the integration effect of CCD to the fringe contrast in image

The pixels on the CCD sensor of a camera have finite sizes. They collect light from small areas on the object's surface instead of sampling on ideal points. The integration effect of the CCD pixels could cause the fringe contrast in the image to drop. A proof of this is given at below.

When phase-shifted fringe patterns are projected on an object, the corresponding light intensity distributions on the camera's image plane can be expressed using the following equation:

$$I_n(u,v) = A(u,v) + B(u,v)\ \sin\left(\Phi(u,v) + 2\pi\frac{n-1}{N}\right)\ , \quad n = 1, \cdots, N \qquad \text{(B.1)}$$

where (u, v) are coordinates on the camera's image plane $\boldsymbol{u} - \boldsymbol{v}$ and n represents different phase-shift. In the following context, the phase-shift term $2\pi(n-1)/N$ will

be dropped for brevity.

For a single pixel on the CCD sensor, the light intensity received is the integration of light over a small area on the image plane, i.e.

$$I(i,j) = \iint \big[A(u,v) + B(u,v)\ \sin\big(\Phi(u,v)\big)\big]\,\mathrm{d}u\mathrm{d}v \tag{B.2}$$

The above integration can also be studied in a discrete manner, which is

$$I(i,j) = \sum_{m=1}^{M} \big[A_m + B_m \sin(\Phi_m)\big] S_p \tag{B.3}$$

where $M \cdot S_p$ is the size of pixel on the image plane. On the right-hand side of the equation, the subscripts i and j are dropped for brevity.

Consider the summation of the first two items in Eqn. B.3 and let $\tilde{I}_{1+2}$ denote the result, i.e.

$$\begin{aligned}\tilde{I}_{1+2} =& \big[A_1 + B_1 \sin(\Phi_1)\big] S_p + \big[A_2 + B_2 \sin(\Phi_2)\big] S_p \\ =& (A_1 + A_2) S_p + S_p \big[B_1 + B_2 \cos(\Phi_2 - \Phi_1)\big] \sin\Phi_1 + \\ & S_p \big[B_2 \sin(\Phi_2 - \Phi_1)\big] \cos\Phi_1\end{aligned} \tag{B.4}$$

Let us introduce variables B' and ϕ' and define their values using the following equation set

$$\begin{cases} 2B'\cos\phi' = B_1 + B_2\cos(\Phi_2 - \Phi_1) \\ 2B'\sin\phi' = B_2\sin(\Phi_2 - \Phi_1) \end{cases} \tag{B.5}$$

By substituting the above equation in Eqn. B.4, $\tilde{I}_{1+2}$ can hence be written in terms of B' and ϕ' as the following

$$\tilde{I}_{1+2} = \left[\frac{A_1 + A_2}{2} + B'\sin(\Phi_1 + \phi')\right](2S_p) \tag{B.6}$$

That is, $\tilde{I}_{1+2}$ is still in a sinusoidal form but with a different fringe contrast, B', and a different phase value, $\Phi_1 + \phi'$.

The value of B' can be solved from Eqn. B.5 as the following:

$$\begin{aligned} B' =& \sqrt{(B' \cos\phi')^2 + (B' \sin\phi')^2} \\ =& \frac{1}{2}\sqrt{B_1^2 + 2B_1B_2\cos(\Phi_2 - \Phi_1) + B_2^2} \end{aligned} \tag{B.7}$$

As we can see that, since Φ_2 is generally not equal to Φ_1, the value of B' is always smaller than the average of B_1 and B_2, and the decrement depends on the difference between Φ_1 and Φ_2. This means that the integration effect of the CCD pixels will cause the fringe contrast in the image to drop and the magnitude of dropping depends on the local fringe pitch in the image. The smaller the fringe pitch is, which means a larger variance of Φ within a pixel, the larger decrement in fringe contrast.

Appendix C

Proofs for Estimation of Measurement Uncertainties

C.1 Relationship between σ_ϕ and σ_I

Images obtained from a standard N-step phase shifting can be expressed using the following equation:

$$I_n(i,j) = A(i,j) + B(i,j)\sin\left(\phi(i,j) + 2\pi\frac{n-1}{N}\right) \tag{C.1}$$

The corresponding phase map can be calculated from

$$\phi(i,j) = \arctan\left(\frac{\sum_{n=1}^{N}\left[I_n(i,j)\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n(i,j)\sin\left(2\pi\frac{n-1}{N}\right)\right]}\right) \tag{C.2}$$

in which each phase value $\phi(i,j)$ is given as a function of the image intensity sequence $I_n(i,j)$ $(n = 1, \cdots, N)$.

By applying the Variance Propagation Law (see Page 173) to the above equation, the relationship between $\sigma_\phi(i,j)$, the standard deviation of $\phi(i,j)$, and $\sigma_{I_n}(i,j)$ $(n =$

$1, \cdots, N$), the standard deviation of I_n, can be written as

(in the following, the subscripts i and j of variables will be dropped for brevity)

$$\sigma_\phi^2 \approx \sum_{n=1}^{N} \left[\left(\frac{\partial \phi}{\partial I_n} \right)^2 \sigma_{I_n}^2 \right] \tag{C.3}$$

The partial derivatives of ϕ to I_n can be derived from Eqn. C.2 as the following:

$$\begin{aligned} \frac{\partial \phi}{\partial I_n} &= \frac{1}{1 + \left(\frac{\sin\phi}{\cos\phi} \right)^2} \left\{ \frac{\cos\left(2\pi \frac{n-1}{N}\right)}{\sum_{m=1}^{N} \left[I_m \sin\left(2\pi \frac{m-1}{N}\right) \right]} - \frac{\sum_{m=1}^{N} \left[I_m \cos\left(2\pi \frac{m-1}{N}\right) \right] \sin\left(2\pi \frac{n-1}{N}\right)}{\left(\sum_{m=1}^{N} \left[I_m \sin\left(2\pi \frac{m-1}{N}\right) \right] \right)^2} \right\} \\ &= \frac{1}{1 + \left(\frac{\sin\phi}{\cos\phi} \right)^2} \left\{ \frac{2\cos\left(2\pi \frac{n-1}{N}\right)}{NB\cos\phi} - \frac{2\sin\phi\sin\left(2\pi \frac{n-1}{N}\right)}{NB\cos^2\phi} \right\} \\ &= \frac{2}{NB} \cos\left(\phi + 2\pi \frac{n-1}{N} \right) \end{aligned} \tag{C.4}$$

in which the following relation between ϕ and the I_n sequence has been used:

$$\begin{cases} \sum_{n=1}^{N} \left[I_n \cos\left(2\pi \frac{n-1}{N}\right) \right] = \frac{NB}{2} \sin\phi \\ \sum_{n=1}^{N} \left[I_n \sin\left(2\pi \frac{n-1}{N}\right) \right] = \frac{NB}{2} \cos\phi \end{cases} \tag{C.5}$$

Since the influence of error sources to the N images is not different from one to another, we can get

$$\sigma_{I_1} = \sigma_{I_2} = \cdots = \sigma_{I_N} = \sigma_I \tag{C.6}$$

Utilizing this property and substituting Eqn. C.4 in Eqn. C.3, we have

$$\begin{aligned}\sigma_\phi^2 &\approx \sum_{n=1}^{N}\left[\left(\frac{2}{NB}\right)^2\cos^2\left(\Phi+2\pi\frac{n-1}{N}\right)\sigma_I^2\right] \\ &= \frac{4\sigma_I^2}{N^2B^2}\sum_{n=1}^{N}\left[\frac{1}{2}+\frac{1}{2}\cos\left(2\Phi+4\pi\frac{n-1}{N}\right)\right] \\ &= \frac{2\sigma_I^2}{NB^2}\end{aligned} \tag{C.7}$$

which can also be written as

$$\sigma_\phi \approx \sqrt{\frac{2}{N}}\cdot\frac{\sigma_I}{B} \tag{C.8}$$

where N is the number of steps in phase shifting and B is the pixel's image intensity contrast as defined in Eqn. C.1.

C.2 Evaluation of ϕ from I_n

Ideally, images obtained from a standard N-step phase shifting can be expressed using the following equation:

$$\hat{I}_n(i,j) = A(i,j) + B(i,j)\sin\left(\phi(i,j)+2\pi\frac{n-1}{N}\right) \tag{C.9}$$

However, due to the influence of varied error sources, the actual intensity sequence $I_n(i,j)$ contains errors and generally does not comply with Eqn. C.9 in cases where $N > 3$. That is, $I_n(i,j)$ $(n = 1, \cdots, N)$ give no solution to $A(i,j)$, $B(i,j)$ and $\phi(i,j)$ since Eqn. C.9 is overdetermined.

Under the assumption that the errors $I_n(i,j)$ are random, Eqn. C.9 can be solved in a least square manner, i.e. to make the variance of $(I_n - \hat{I}_n)$ to be minimum:

(subscript i and j dropped for brevity)

$$\sum_{n=1}^{N}\left[I_n - A - B\sin\left(\phi + 2\pi\frac{n-1}{N}\right)\right]^2 \tag{C.10}$$

Therefore, we can get

$$\sum_{n=1}^{N}\left\{2\left[I_n - A - B\sin\left(\phi + 2\pi\frac{n-1}{N}\right)\right](-B)\cos\left(\phi + 2\pi\frac{n-1}{N}\right)\right\} = 0 \tag{C.11}$$

which can be simplified to

$$\begin{aligned}
&\sum_{n=1}^{N}\left[I_n\cos\left(\phi + 2\pi\frac{n-1}{N}\right)\right] - B\sum_{n=1}^{N}\left[\sin\left(\phi + 2\pi\frac{n-1}{N}\right)\cos\left(\phi + 2\pi\frac{n-1}{N}\right)\right] = 0 \\
&\sum_{n=1}^{N}\left[I_n\cos\phi\cos\left(2\pi\frac{n-1}{N}\right) - I_n\sin\phi\sin\left(2\pi\frac{n-1}{N}\right)\right] = 0
\end{aligned} \tag{C.12}$$

The final form of the above equation suggests that phase value ϕ can be calculated from the following:

$$\frac{\sin\phi}{\cos\phi} = \frac{\sum_{n=1}^{N}\left[I_n\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n\sin\left(2\pi\frac{n-1}{N}\right)\right]} \tag{C.13}$$

in another form

$$\phi(i,j) = \arctan\left(\frac{\sum_{n=1}^{N}\left[I_n(i,j)\cos\left(2\pi\frac{n-1}{N}\right)\right]}{\sum_{n=1}^{N}\left[I_n(i,j)\sin\left(2\pi\frac{n-1}{N}\right)\right]}\right) \tag{C.14}$$

We can see that, Eqn. C.14 is the formula to compute the unbiased value of ϕ from erroneous I_n sequence under the assumption that the errors in I_n are random.

C.3 Evaluation of σ_I from I_n

Let ΔI denote the difference between the compensated image intensity$\tilde{I}$ and the raw image intensity I, i.e.

$$\Delta I_s(i,j) = \tilde{I}_s(i,j) - I_s(i,j)\ , \quad s = 1, \cdots, N \tag{C.15}$$

From Eqn. 6.9, 6.7, 6.8 and 6.2, we can see that each $\Delta I_s(i,j)$ can eventually be written as a function of the raw image intensities associated with pixel (i,j), i.e.

$$\Delta I_s(i,j) = f_s\Big(I_1(i,j),\, I_2(i,j),\, \cdots,\, I_N(i,j)\Big)\ , \quad s = 1, \cdots, N \tag{C.16}$$

By applying the Variance Propagation Law (see Page 173) to the above equation, we can get
(in the rest of this section, the subscripts i and j will be dropped for brevity)

$$\sigma^2_{\Delta I_s} \approx \sum_{n=1}^{N} \left[\left(\frac{\partial(\Delta I_s)}{\partial I_n} \right)^2 \sigma^2_{I_n} \right] \tag{C.17}$$

where $\sigma_{\Delta I_s}$ is the standard deviation of ΔI_s and σ_{I_n} is the standard deviation of I_n.

The partial derivatives of ΔI_s to I_n can be derived from Eqn. C.15 and 6.9 as the following:

$$\begin{aligned}
\frac{\partial(\Delta I_s)}{\partial I_n} &= \frac{\partial \tilde{A}}{\partial I_n} + \left[\tilde{B} \cos\left(\phi + 2\pi \frac{s-1}{N} \right) \frac{\partial \phi}{\partial I_n} + \frac{\partial \tilde{B}}{\partial I_n} \sin\left(\phi + 2\pi \frac{s-1}{N} \right) \right] - \frac{\partial I_s}{\partial I_n} \\
&= \frac{1}{N} + \left[\tilde{B} \cos\left(\phi + 2\pi \frac{s-1}{N} \right) \frac{2}{N\tilde{B}} \cos\left(\phi + 2\pi \frac{n-1}{N} \right) + \right. \\
&\qquad \left. \frac{2}{N} \sin\left(\phi + 2\pi \frac{n-1}{N} \right) \sin\left(\phi + 2\pi \frac{s-1}{N} \right) \right] - \delta(n-s) \\
&= \frac{1}{N} + \frac{2}{N} \cos\left(2\pi \frac{n-s}{N} \right) - \delta(n-s)
\end{aligned} \tag{C.18}$$

where $\delta(n-s)$ denotes the delta function, which is

$$\delta(n-s) = \begin{cases} 1 & \text{if} \quad n = s \\ 0 & \text{else} \end{cases} \tag{C.19}$$

In the above derivation, the following relations about $\tilde{A}$, $\tilde{B}$ and I_n were used, as well as Eqn. C.4:

$$\begin{cases} \dfrac{\partial \tilde{A}}{\partial I_n} = \dfrac{1}{N} \\ \dfrac{\partial \tilde{B}}{\partial I_n} = \dfrac{2}{N} \sin\left(\phi + 2\pi \dfrac{n-1}{N}\right) \end{cases} \tag{C.20}$$

Substituting Eqn. C.18 in Eqn. C.17 and utilizing the assumption made in Eqn. C.6, we get the relationship between $\sigma_{\Delta I_s}$ and σ_I as the following:

$$\begin{aligned} \sigma_{\Delta I_s}^2 &\approx \sigma_I^2 \sum_{n=1}^{N} \left[\frac{1}{N} + \frac{2}{N}\cos\left(2\pi\frac{n-s}{N}\right) - \delta(n-s)\right]^2 \\ &= \sigma_I^2 \left(1 - \frac{3}{N}\right) \end{aligned} \tag{C.21}$$

Assuming that the error in image intensity is random noise with a normal distribution, we can compute $\sigma_{\Delta I_s}$ by means of statistics as the following:

$$\sigma_{\Delta I} = \sqrt{\frac{1}{N-1}\sum_{s=1}^{N}(\tilde{I}_s - I_s)^2} \tag{C.22}$$

Combining Eqn. C.22 and C.21, we can get that

$$\sigma_I(i,j) \approx \sqrt{\frac{N}{(N-3)(N-1)}\sum_{n=1}^{N}\left[\tilde{I}_n(i,j) - I_n(i,j)\right]^2} \tag{C.23}$$

C.4 Uncertainty propagation from $(x^{(d)}, y^{(d)})$ to $(x^{(n)}, y^{(n)})$

The transformation from the undistorted image coordinates $(x^{(n)}, y^{(n)})$ to the distorted image coordinates $(x^{(d)}, y^{(d)})$ is given by the following equation:

$$\begin{bmatrix} x^{(d)} \\ y^{(d)} \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix} + \begin{bmatrix} 2k_3 x^{(n)} y^{(n)} + k_4 \left(r^2 + 2\left(x^{(n)}\right)^2\right) \\ k_3 \left(r^2 + 2\left(y^{(n)}\right)^2\right) + 2k_4 x^{(n)} y^{(n)} \end{bmatrix} \tag{C.24}$$

where $r^2 = \left(x^{(n)}\right)^2 + \left(y^{(n)}\right)^2$ and k_s $(s = 1, \cdots, 5)$ are the lens distortion coefficients. The partial derivatives of $x^{(d)}$ and $y^{(d)}$ to $x^{(n)}$, $y^{(n)}$ and k_s $(s = 1, \cdots, 5)$ can be derived from the equation above:

$$\begin{cases} \dfrac{\partial x^{(d)}}{\partial k_1} = r^2 x^{(n)}\ , \quad \dfrac{\partial x^{(d)}}{\partial k_2} = r^4 x^{(n)}\ , \quad \dfrac{\partial x^{(d)}}{\partial k_3} = 2x^{(n)} y^{(n)} \\ \dfrac{\partial x^{(d)}}{\partial k_4} = r^2 + 2\left(x^{(n)}\right)^2\ , \quad \dfrac{\partial x^{(d)}}{\partial k_5} = r^6 x^{(n)} \\ \dfrac{\partial x^{(d)}}{\partial x^{(n)}} = \left(2k_1 + 4k_2 r^2 + 6k_5 r^4\right)\left(x^{(n)}\right)^2 + 6k_4 x^{(n)} + 2k_3 y^{(n)} + \\ \qquad \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \\ \dfrac{\partial x^{(d)}}{\partial y^{(n)}} = \left(2k_1 + 4k_2 r^2 + 6k_5 r^4\right) x^{(n)} y^{(n)} + 2k_3 x^{(n)} + 2k_4 y^{(n)} \end{cases} \tag{C.25}$$

$$\begin{cases} \dfrac{\partial y^{(d)}}{\partial k_1} = r^2 y^{(n)}\ , \quad \dfrac{\partial y^{(d)}}{\partial k_2} = r^4 y^{(n)}\ , \quad \dfrac{\partial y^{(d)}}{\partial k_3} = r^2 + 2\left(y^{(n)}\right)^2 \\ \dfrac{\partial y^{(d)}}{\partial k_4} = 2x^{(n)} y^{(n)}\ , \quad \dfrac{\partial y^{(d)}}{\partial k_5} = r^6 y^{(n)} \\ \dfrac{\partial y^{(d)}}{\partial x^{(n)}} = \left(2k_1 + 4k_2 r^2 + 6k_5 r^4\right) x^{(n)} y^{(n)} + 2k_3 x^{(n)} + 2k_4 y^{(n)} \\ \dfrac{\partial y^{(d)}}{\partial y^{(n)}} = \left(2k_1 + 4k_2 r^2 + 6k_5 r^4\right)\left(y^{(n)}\right)^2 + 6k_3 y^{(n)} + 2k_4 x^{(n)} + \\ \qquad \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \end{cases} \tag{C.26}$$

Assuming that $\hat{x}^{(d)}$, $\hat{y}^{(d)}$, $\hat{x}^{(n)}$, $\hat{y}^{(n)}$ and $\hat{k_s}$ $(s = 1, \cdots, 5)$ are the true values of the corresponding parameters, the following equations can be obtained by a first order Taylor expansion at these true values:

$$\begin{aligned} x^{(d)} - \hat{x}^{(d)} &\approx \frac{\partial x^{(d)}}{\partial x^{(n)}}\left(x^{(n)} - \hat{x}^{(n)}\right) + \frac{\partial x^{(d)}}{\partial y^{(n)}}\left(y^{(n)} - \hat{y}^{(n)}\right) + \sum_{s=1}^{5}\left[\frac{\partial x^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right] \\ y^{(d)} - \hat{y}^{(d)} &\approx \frac{\partial y^{(d)}}{\partial x^{(n)}}\left(x^{(n)} - \hat{x}^{(n)}\right) + \frac{\partial y^{(d)}}{\partial y^{(n)}}\left(y^{(n)} - \hat{y}^{(n)}\right) + \sum_{s=1}^{5}\left[\frac{\partial y^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right] \end{aligned} \tag{C.27}$$

By a rearrangement of terms in Eqn. C.27, we can get

$$\begin{aligned} &\left(\frac{\partial x^{(d)}}{\partial x^{(n)}} \cdot \frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}} \cdot \frac{\partial x^{(d)}}{\partial y^{(n)}}\right) \cdot \left(x^{(n)} - \hat{x}^{(n)}\right) \approx \\ &\qquad \frac{\partial y^{(d)}}{\partial y^{(n)}} \cdot \left(\left(x^{(d)} - \hat{x}^{(d)}\right) - \sum_{s=1}^{5}\left[\frac{\partial x^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right]\right) - \\ &\qquad \frac{\partial x^{(d)}}{\partial y^{(n)}} \cdot \left(\left(y^{(d)} - \hat{y}^{(d)}\right) - \sum_{s=1}^{5}\left[\frac{\partial y^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right]\right) \end{aligned} \tag{C.28}$$

$$\begin{aligned} &\left(\frac{\partial x^{(d)}}{\partial x^{(n)}} \cdot \frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}} \cdot \frac{\partial x^{(d)}}{\partial y^{(n)}}\right) \cdot \left(y^{(n)} - \hat{y}^{(n)}\right) \approx \\ &\qquad \frac{\partial x^{(d)}}{\partial x^{(n)}} \cdot \left(\left(y^{(d)} - \hat{y}^{(d)}\right) - \sum_{s=1}^{5}\left[\frac{\partial y^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right]\right) - \\ &\qquad \frac{\partial y^{(d)}}{\partial x^{(n)}} \cdot \left(\left(x^{(d)} - \hat{x}^{(d)}\right) - \sum_{s=1}^{5}\left[\frac{\partial x^{(d)}}{\partial k_s}\left(k_s - \hat{k_s}\right)\right]\right) \end{aligned} \tag{C.29}$$

By applying the Variance Propagation Law (see Page 173) to Eqn. C.28 and C.29, the variances of $x^{(n)}$ and $y^{(n)}$ can be written as the following:

$$\begin{aligned} \sigma^2_{x^{(n)}} \approx &\left[\left(\frac{\partial y^{(d)}}{\partial y^{(n)}}\right)^2 \sigma^2_{x^{(d)}} + \left(\frac{\partial x^{(d)}}{\partial y^{(n)}}\right)^2 \sigma^2_{y^{(d)}} + \sum_{s=1}^{5}\left(\frac{\partial y^{(d)}}{\partial y^{(n)}}\frac{\partial x^{(d)}}{\partial k_s} - \frac{\partial x^{(d)}}{\partial y^{(n)}}\frac{\partial y^{(d)}}{\partial k_s}\right)^2 \sigma^2_{k_s}\right] \\ &\Big/ \left(\frac{\partial x^{(d)}}{\partial x^{(n)}}\frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}}\frac{\partial x^{(d)}}{\partial y^{(n)}}\right)^2 \end{aligned} \tag{C.30}$$

$$\sigma_{y^{(n)}}^2 \approx \left[\left(\frac{\partial y^{(d)}}{\partial x^{(n)}} \right)^2 \sigma_{x^{(d)}}^2 + \left(\frac{\partial x^{(d)}}{\partial x^{(n)}} \right)^2 \sigma_{y^{(d)}}^2 + \sum_{s=1}^{5} \left(\frac{\partial x^{(d)}}{\partial x^{(n)}} \frac{\partial y^{(d)}}{\partial k_s} - \frac{\partial y^{(d)}}{\partial x^{(n)}} \frac{\partial x^{(d)}}{\partial k_s} \right)^2 \sigma_{k_s}^2 \right] \Bigg/ \left(\frac{\partial x^{(d)}}{\partial x^{(n)}} \frac{\partial y^{(d)}}{\partial y^{(n)}} - \frac{\partial y^{(d)}}{\partial x^{(n)}} \frac{\partial x^{(d)}}{\partial y^{(n)}} \right)^2 \tag{C.31}$$

In many cases, only the effect of radial distortion needs to be considered in the lens distortion, i.e. k_2 to k_5 can be assumed to be zeros. As a result, the partial derivatives of $x^{(d)}$ and $y^{(d)}$ can be simplified as the following:

$$\begin{cases} \dfrac{\partial x^{(d)}}{\partial k_1} = r^2 x^{(n)} \\ \dfrac{\partial x^{(d)}}{\partial x^{(n)}} = 2k_1 \left(x^{(n)} \right)^2 + 1 + k_1 r^2 \\ \dfrac{\partial x^{(d)}}{\partial y^{(n)}} = 2k_1 x^{(n)} y^{(n)} \end{cases}, \quad \begin{cases} \dfrac{\partial y^{(d)}}{\partial k_1} = r^2 y^{(n)} \\ \dfrac{\partial y^{(d)}}{\partial x^{(n)}} = 2k_1 x^{(n)} y^{(n)} \\ \dfrac{\partial y^{(d)}}{\partial y^{(n)}} = 2k_1 \left(y^{(n)} \right)^2 + 1 + k_1 r^2 \end{cases} \tag{C.32}$$

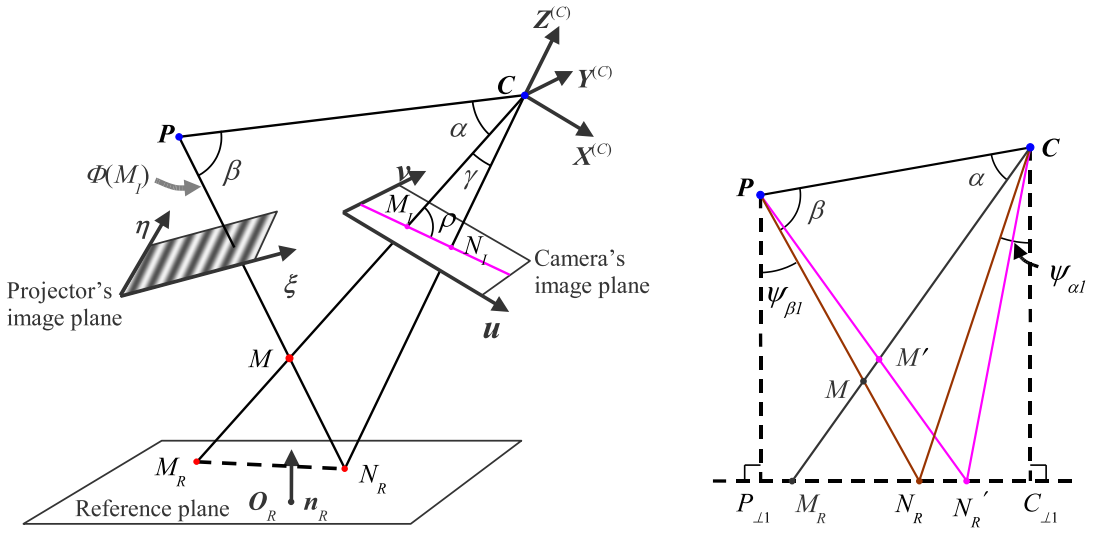
The variances of $x^{(n)}$ and $y^{(n)}$ can then be written as

$$\begin{aligned} \sigma_{x^{(n)}}^2 \approx & \Bigg[\left(1 + k_1 r^2 + 2k_1 \left(y^{(n)} \right)^2 \right)^2 \sigma_{x^{(d)}}^2 + \left(2k_1 x^{(n)} y^{(n)} \right)^2 \sigma_{y^{(d)}}^2 + \\ & \left(\left(1 + k_1 r^2 \right) r^2 x^{(n)} \right)^2 \sigma_{k_1}^2 \Bigg] \Bigg/ \Big[\left(1 + k_1 r^2 \right) \left(1 + 3k_1 r^2 \right) \Big]^2 \end{aligned} \tag{C.33}$$

$$\begin{aligned} \sigma_{y^{(n)}}^2 \approx & \Bigg[\left(2k_1 x^{(n)} y^{(n)} \right)^2 \sigma_{x^{(d)}}^2 + \left(1 + k_1 r^2 + 2k_1 \left(x^{(n)} \right)^2 \right)^2 \sigma_{y^{(d)}}^2 + \\ & \left(\left(1 + k_1 r^2 \right) r^2 y^{(n)} \right)^2 \sigma_{k_1}^2 \Bigg] \Bigg/ \Big[\left(1 + k_1 r^2 \right) \left(1 + 3k_1 r^2 \right) \Big]^2 \end{aligned} \tag{C.34}$$

C.5 Relationship between angle β and angle γ

In Fig. C.1, point $C_{\perp 1}$ is the projection of point C on the intersection of the reference plane and the plane defined by points C, P and M_I. Similarly, point $P_{\perp 1}$ is the projection of point P. Points $C_{\perp 1}$, $P_{\perp 1}$, M_R and N_R are collinear. As point N_I moves along line $\overline{P_I M_I}$, point N_R moves along line $\overline{P_{\perp 1} C_{\perp 1}}$ accordingly. Let $\psi_{\beta 1}$ and $\psi_{\alpha 1}$ denote angles $\angle N_R P P_{\perp 1}$ and $\angle N_R C C_{\perp 1}$ respectively.



(a) Definition of angle γ (b) Definition of angle $\psi_{\beta 1}$ and $\psi_{\alpha 1}$

Figure C.1: Relationship between angle β and γ

From right triangle $\triangle N_R P P_{\perp 1}$, we can get the following equation:

$$\tan \psi_{\beta 1} = \frac{|N_R P_{\perp 1}|}{|P P_{\perp 1}|} \tag{C.35}$$

Differentiating both sides of the equation w.r.t. $|N_R P_{\perp 1}|$, we get that

$$\frac{\mathrm{d}}{\mathrm{d}\psi_{\beta 1}}\big(\tan \psi_{\beta 1}\big) \cdot \frac{\mathrm{d}\psi_{\beta 1}}{\mathrm{d}|N_R P_{\perp 1}|} = \frac{1}{|P P_{\perp 1}|} \tag{C.36}$$

which can also be written as the following by a rearranging of terms:

$$\frac{\mathrm{d}\psi_{\beta 1}}{\mathrm{d}|N_R P_{\perp 1}|} = \frac{\cos^2 \psi_{\beta 1}}{|P P_{\perp 1}|} \tag{C.37}$$

Similarly, we can get the following equation about $\psi_{\alpha 1}$ from right triangle $\triangle N_R C C_{\perp 1}$:

$$\frac{\mathrm{d}\psi_{\alpha 1}}{\mathrm{d}|N_R C_{\perp 1}|} = \frac{\cos^2 \psi_{\alpha 1}}{|C C_{\perp 1}|} \tag{C.38}$$

Notice that $\mathrm{d}|N_R P_{\perp 1}| = \pm\, \mathrm{d}|N_R C_{\perp 1}|$, where the sign depends on whether point N_R is in the middle of $P_{\perp 1}$ and $C_{\perp 1}$ or outside. Therefore, Eqn. C.37 and C.38 can be merged to a single equation as the following:

$$\mathrm{d}\psi_{\beta 1} = \pm\, \frac{|CC_{\perp 1}| \cos^2 \psi_{\beta 1}}{|PP_{\perp 1}| \cos^2 \psi_{\alpha 1}}\, \mathrm{d}\psi_{\alpha 1} \tag{C.39}$$

From the definitions of the angles (see Fig. C.1), it is also known that

$$|\mathrm{d}\psi_{\beta 1}| = |\mathrm{d}\beta| \quad \text{and} \quad |\mathrm{d}\psi_{\alpha 1}| = |\mathrm{d}\gamma| \tag{C.40}$$

Define $\mathrm{d}\gamma$ to be positive if as a result N_I moves toward P_I (P_I is the projection of point P on the camera's image plane $\boldsymbol{u}-\boldsymbol{v}$). The relationship between a small change in γ and the corresponding change in β can be written as the following:

$$\mathrm{d}\beta = \frac{|CC_{\perp 1}| \cos^2 \psi_{\beta 1}}{|PP_{\perp 1}| \cos^2 \psi_{\alpha 1}}\, \mathrm{d}\gamma \tag{C.41}$$

C.6 Relationship between $|CM|$ and $\delta P_\perp$

The relationship between the error component $\delta P_\perp$ and the depth $|CM|$ is illustrated in Fig. C.2, where the left diagram shows the case that α is acute and the right diagram for α being obtuse. As a result of $\delta P_\perp$, point P moves along $\boldsymbol{e}_\perp$ to P', i.e. $\delta P_\perp \boldsymbol{e}_\perp = P' - P$, and accordingly point M moves along $\overline{CM_I}$ to M'.

Consider the case shown in the left diagram in Fig. C.2. From the similar triangles $\triangle N_R M M^*$ and $\triangle N_R P P'$, we can get the following equation based on the fact that $\delta P_\perp$ is very small comparing to $|PN_R|$:

$$\frac{|MM'| \cos \psi_{\alpha 2}}{\delta P_\perp} = \frac{|N_R M|}{|N_R P|} \tag{C.42}$$

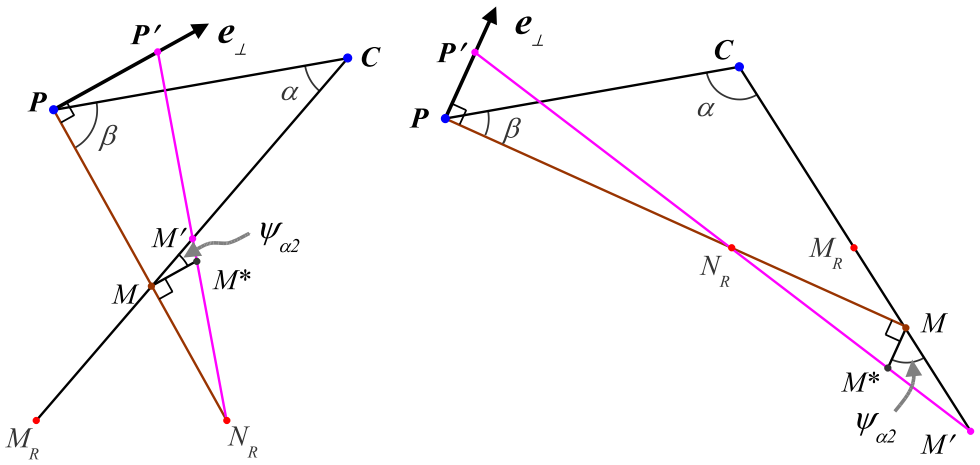


Figure C.2: Relationship between $|CM|$ and $\delta P_\perp$

where $\psi_{\alpha 2}$ is the angle between line $\overline{MM^*}$ and $\overline{MM'}$. Notice that $\psi_{\alpha 2} = \alpha + \beta - \pi/2$. Differentiating both sides of the equation w.r.t. $\delta P_\perp$ and utilizing the fact that $d|CM| = -|MM'|$, we can get the following equation:

$$\frac{\partial |CM|}{\partial (\delta P_\perp)} = -\frac{|N_R M|}{|N_R P| \cos(\alpha + \beta - \pi/2)} \tag{C.43}$$

Considering different cases in which β and α can either be acute or obtuse (but not both obtuse), the final equation for describing the relationship between $\delta P_\perp$ and the corresponding change in $|CM|$ can be written as the following:

$$\frac{\partial |CM|}{\partial (\delta P_\perp)} = \frac{|N_R M|}{\sin(\alpha + \beta)|N_R P|} \operatorname{sign}\left(\overrightarrow{N_R M} \cdot \overrightarrow{N_R P}\right) \operatorname{sign}\left(\overrightarrow{CP} \cdot \boldsymbol{e}_\perp\right) \tag{C.44}$$

where function sign() is used to extract the sign of variable, i.e. $\operatorname{sign}(x) = x/|x|$.

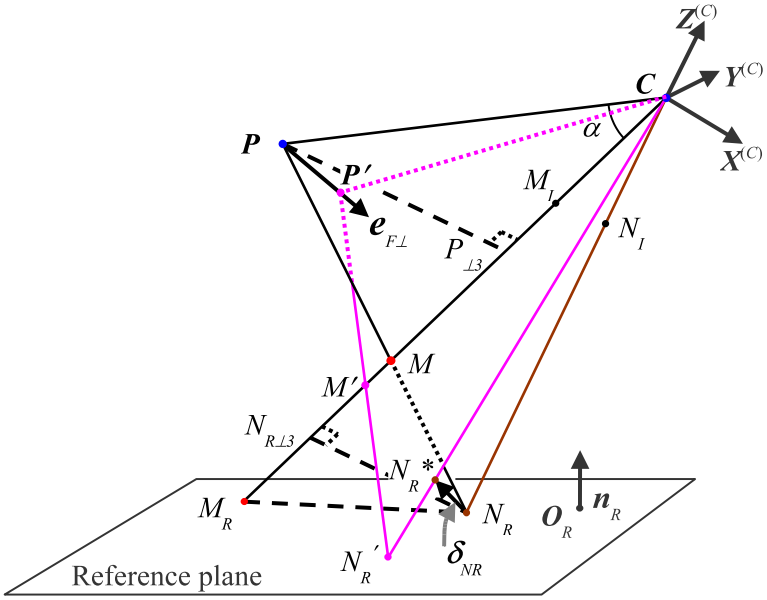


Figure C.3: Relationship between $|CN_R|$ and $\delta P_{F\perp}$

C.7 Relationship between $|CN_R|$ and $\delta P_{F\perp}$

Figure C.3 illustrates the change of $|CN_R|$ as point P moves along $\boldsymbol{e}_{F\perp}$ by $\delta P_{F\perp}$. Due to the movement of P, plane CPM_I rotates by a certain angle around axis $\overline{CM_I}$ and as a result $\overrightarrow{CN_R}$ moves from its original position to $\overrightarrow{CN_R^*}$. Utilizing the principles of rigid body rotation, we can get the following equation:

$$\frac{\boldsymbol{\delta}_{N_R}}{|N_R N_{R\perp 3}|} = \frac{\delta P_{F\perp}\, \boldsymbol{e}_{F\perp}}{|PP_{\perp 3}|} \operatorname{sign}\left(\overrightarrow{N_R N_{R\perp 3}} \cdot \overrightarrow{PP_{\perp 3}}\right) \tag{C.45}$$

where $P_{\perp 3}$ is the projection of point P on line $\overline{CM_R}$, $N_{R\perp 3}$ is the projection of N_R, and $\boldsymbol{\delta}_{N_R} = N_R^* - N_R$. Noticing that $|N_R N_{R\perp 3}|/|PP_{\perp 3}| = |N_R M|/|PM|$, we can write the above equation as the following:

$$\boldsymbol{\delta}_{N_R} = \frac{|N_R M|\, \delta P_{F\perp}\, \boldsymbol{e}_{F\perp}}{|PM|} \operatorname{sign}\left(\overrightarrow{N_R M} \cdot \overrightarrow{PM}\right) \tag{C.46}$$

Let N'_R denote the intersection of the reference plane with the extension of $\overrightarrow{CN^*_R}$, i.e. N'_R is the new position of N_R. Based on the fact that both N_R and N'_R are on the reference plane, the following equation can be obtained:

$$s\left(\overrightarrow{CN_R}+\boldsymbol{\delta}_{N_R}\right)\cdot\boldsymbol{n}_R=\overrightarrow{CN_R}\cdot\boldsymbol{n}_R \tag{C.47}$$

where the coefficient s is defined as $|CN'_R|/|CN_R|$. By rearranging the terms, we have

$$s=\frac{\overrightarrow{CN_R}\cdot\boldsymbol{n}_R}{\left(\overrightarrow{CN_R}+\boldsymbol{\delta}_{N_R}\right)\cdot\boldsymbol{n}_R} \tag{C.48}$$

Let $d|CN_R|$ denote the change of $|CN_R|$ caused by $\delta P_{F\perp}$. Utilizing the definition of s and the equation above, we can get

$$\mathrm{d}|CN_R|=(s-1)|CN_R|=-\frac{(\boldsymbol{\delta}_{N_R}\cdot\boldsymbol{n}_R)\,|CN_R|}{\left(\overrightarrow{CN_R}+\boldsymbol{\delta}_{N_R}\right)\cdot\boldsymbol{n}_R} \tag{C.49}$$

Since $(\boldsymbol{\delta}_{N_R}\cdot\boldsymbol{n}_R)$ is much smaller than $(\overrightarrow{CN_R}\cdot\boldsymbol{n}_R)$, the above equation can be simplified as the following:

$$\mathrm{d}|CN_R|\approx-\frac{(\boldsymbol{\delta}_{N_R}\cdot\boldsymbol{n}_R)\,|CN_R|}{\overrightarrow{CN_R}\cdot\boldsymbol{n}_R} \tag{C.50}$$

Substitute Eqn. C.46 into Eqn. C.50, we can get the following equation that describes the relationship between $\delta P_{F\perp}$ and the corresponding change in $|CN_R|$:

$$\frac{\partial|CN_R|}{\partial(\delta P_{F\perp})}=-\frac{(\boldsymbol{e}_{F\perp}\cdot\boldsymbol{n}_R)|CN_R||N_RM|}{(\overrightarrow{CN_R}\cdot\boldsymbol{n}_R)|PM|}\,\mathrm{sign}\left(\overrightarrow{N_RM}\cdot\overrightarrow{PM}\right) \tag{C.51}$$

C.8 Relationship between $|M_IN_I|_+$ and $\delta P_{F\perp}$

As point P moves along $\boldsymbol{e}_{F\perp}$ by $\delta P_\perp$, its projection on the camera's image plane $\boldsymbol{u}-\boldsymbol{v}$ moves accordingly from P_I to P'_I. As shown in Fig. C.4, as point P_I moves

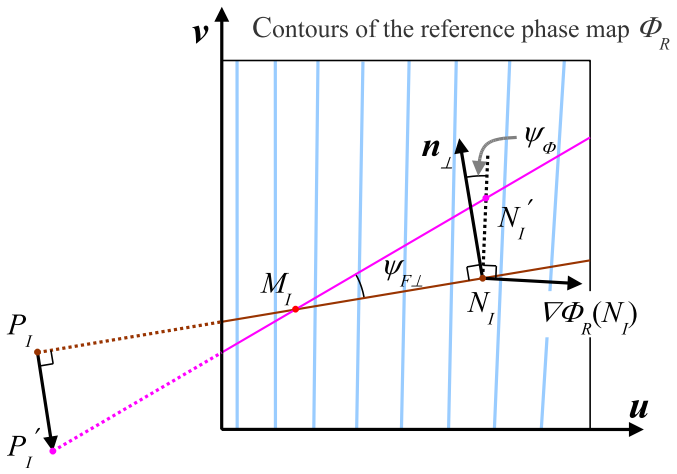


Figure C.4: Relationship between $|M_I N_I|_+$ and $|P_I P_I'|$

to P_I', line $\overline{P_I M_I}$ rotates by an angle of $\psi_{F\perp}$ to $\overline{P_I' M_I}$. As a result, point N_I also moves to a new position N_I'. Notice that N_I satisfies $\Phi_R(N_I) = \Phi(M_I)$ and the same constraint applies for N_I' as well, i.e. $\Phi_R(N_I') = \Phi(M_I)$. Therefore, for a small angle $\psi_{F\perp}$, $N_I N_I'$ is along the contour line (of the reference phase map) that passes through point N_I. In other words, $N_I N_I'$ is perpendicular to vector $\nabla\Phi_R(N_I)$, which is the phase gradient of the reference phase map at N_I. Let us introduce vector $\boldsymbol{n}_\perp$ and angle ψ_Φ, where $\boldsymbol{n}_\perp$ is perpendicular to $\overline{M_I N_I}$ and points towards N_I' and ψ_Φ is the angle between $\overline{N_I N_I'}$ and $\boldsymbol{n}_\perp$.

Let $|M_I N_I|_+$ denote the signed length of $\overline{M_I N_I}$, which can be described using the following equation:

$$|M_I N_I|_+ = \overrightarrow{M_I N_I} \cdot \frac{\overrightarrow{M_I P_I}}{|M_I P_I|} \tag{C.52}$$

Let $\mathrm{d}|M_I N_I|_+$ denote the change of $|M_I N_I|_+$ due to the movement of P_I. The absolute value of $\mathrm{d}|M_I N_I|_+$ can be calculated as follows:

$$\left|\mathrm{d}|M_I N_I|_+\right| \approx |M_I N_I|\ \psi_{F\perp} \tan\psi_\Phi \tag{C.53}$$

Notice that for a small angle $\psi_{F\perp}$, $\psi_{F\perp} \approx |P_I P_I'|/|P_I M_I|$. Therefore, the above equation can be written as

$$\left|\mathrm{d}|M_I N_I|_+\right| \approx \frac{|M_I N_I||P_I P_I'| \tan\psi_\Phi}{|P_I M_I|} \tag{C.54}$$

Considering different position arrangements of points P_I, M_I and N_I and different rotation directions of $\overline{P_I M_I}$, the following equation can be obtained to describe the relationship between $|P_I P_I'|$ and the corresponding change in $|M_I N_I|_+$:

$$\frac{\partial|M_I N_I|_+}{\partial|P_I P_I'|} = -\frac{|M_I N_I| \tan\psi_\Phi}{|P_I M_I|} \operatorname{sign}\left(\overrightarrow{P_I P_I'} \cdot \nabla\Phi_R(N_I)\right) \tag{C.55}$$

in which angle ψ_Φ can be solved from the following equation:

$$\cos\psi_\Phi = \frac{\overrightarrow{M_I N_I} \cdot \nabla\Phi_R(N_I)}{|M_I N_I||\nabla\Phi_R(N_I)|}, \quad \psi_\Phi \in (0, \pi) \tag{C.56}$$

In most cases, $|M_I N_I|$ is much smaller than $|P_I M_I|$, and angle ψ_Φ is small because the phase contours are nearly perpendicular to $\overline{P_I M_I}$. As a result, the change of $|M_I N_I|_+$ due to $|P_I P_I'|$ is generally one order of magnitude smaller than the changes (of $|M_I N_I|_+$) caused by other error components, such as the phase errors. Therefore, the depth error caused by $\delta P_{F\perp}$ can be neglected in general.

C.9 Uncertainty of the point-to-surface distance

Let M_E denote the expectation value of the position of point M. The probability distribution of M in the 3-D space can be described in the cylindrical coordinate frame defined by M_E and $\boldsymbol{V}_U$ (see Fig. C.5). Let $\boldsymbol{n}_S$ denote the surface normal direction (unit vector) at M_E. $\boldsymbol{n}_S$ can be decomposed into two parts: a component along $\boldsymbol{V}_U$, denoted by $\boldsymbol{n}_{S\parallel}$; and a component perpendicular to $\boldsymbol{V}_U$, denoted by $\boldsymbol{n}_{S\perp}$. Let $\boldsymbol{n}_{S\perp}$

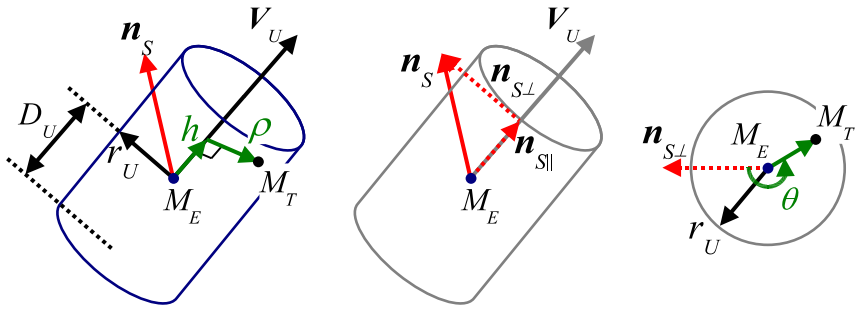


Figure C.5: Cylindrical coordinate frame for describing the probability distribution of M

be the radial direction of the coordinate frame where $\theta = 0$. With the cylindrical coordinate system defined, the position of an arbitrary 3-D point can be located by coordinates (h, ρ, θ), as shown in Fig. C.5. As assumed in the uncertainty model, the probability distributions of point M along the axial, radial, and tangential directions (of the cylindrical frame) are independent of each other. Therefore, the probability distribution function of M, $p_M(h, \rho, \theta)$, can be written as the following:

$$p_M(h, \rho, \theta) = p_M(h)\, p_M(\rho)\, p_M(\theta) \tag{C.57}$$

Based on the definition of the cylindrical uncertainty region, the following equations can also be obtained: (In the following, the subscript M will be dropped for brevity)

$$\int p(h)\, h^2 \mathrm{d}h = \sigma_h^2 = D_U^2\ , \quad \int p(\rho)\, \rho^2 \mathrm{d}\rho = \sigma_\rho^2 = r_U^2 \quad \text{and} \quad p(\theta) = \frac{1}{2\pi} \tag{C.58}$$

For a point $M_T(h, \rho, \theta)$, the point-to-surface distance, d_S, can be calculated from the following equation:

$$d_S = \boldsymbol{n}_S \cdot \overrightarrow{M_E M_T} = \boldsymbol{n}_S \cdot \left(h\boldsymbol{V}_U + \rho\, \overrightarrow{\theta} \right) \tag{C.59}$$

where the unit vector $\overrightarrow{\theta}$ represents the projection of $\overrightarrow{M_E M_T}$ on the plane that is

perpendicular to $\boldsymbol{V}_U$. Using the probability distribution functions, the variance of d_S, denoted by σ_{ds}^2, can be written as the following:

$$\sigma_{ds}^2 = \iiint p(h)\, p(\rho)\, p(\theta)\, d_S^2 \, \mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta \tag{C.60}$$

Substituting Eqn. C.59 in the above equation, we can get

$$\begin{aligned}\sigma_{ds}^2 = &\iiint p(h)\, p(\rho)\, p(\theta)\, h^2 \big(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\big)^2 \mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta \;+ \\ &\iiint p(h)\, p(\rho)\, p(\theta)\, 2h\rho \left(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\right)\left(\boldsymbol{n}_S \cdot \overrightarrow{\theta}\right) \mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta \;+ \\ &\iiint p(h)\, p(\rho)\, p(\theta)\, \rho^2 \left(\boldsymbol{n}_S \cdot \overrightarrow{\theta}\right)^2 \mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta\end{aligned} \tag{C.61}$$

As can be seen from above, σ_{ds}^2 is the summation of three integration terms. The calculation of each individual term is explained at below.

- Since the unit vectors $\boldsymbol{n}_S$ and $\boldsymbol{V}_U$ do not depend on the coordinates (h, ρ, θ), the first term in Eqn. C.61 can be simplified as follows by utilizing Eqn. C.58:

$$\begin{aligned}&\iiint p(h)\, p(\rho)\, p(\theta)\, h^2 \big(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\big)^2 \mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta \\ &= \big(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\big)^2 \int p(h) h^2 \mathrm{d}h \int p(\rho)\mathrm{d}\rho \int_0^{2\pi} p(\theta)\mathrm{d}\theta \\ &= \big(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\big)^2 D_U^2\end{aligned} \tag{C.62}$$

- By utilizing the decomposition of $\boldsymbol{n}_S$ and the definition of unit vector $\overrightarrow{\theta}$, the following equation can be derived:

$$\begin{aligned}\boldsymbol{n}_S \cdot \overrightarrow{\theta} &= \boldsymbol{n}_{S\perp} \cdot \overrightarrow{\theta} + \boldsymbol{n}_{S\parallel} \cdot \overrightarrow{\theta} \\ &= \sqrt{1 - \big(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\big)^2} \cos\theta\end{aligned} \tag{C.63}$$

Hence, the second term in Eqn. C.61 can be written as the following:

$$\begin{aligned}&\iiint p(h)\,p(\rho)\,p(\theta)\,2h\rho\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)\left(\boldsymbol{n}_S\cdot\overrightarrow{\theta}\right)\mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta\\&=2\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)\sqrt{1-\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)^2}\int p(h)\,h\mathrm{d}h\int p(\rho)\,\rho\mathrm{d}\rho\int_0^{2\pi}p(\theta)\cos\theta\mathrm{d}\theta\end{aligned}\tag{C.64}$$

Since M_E (the origin of the cylindrical coordinate frame) is the expectation value of the position of M, the following equations hold:

$$\int p(h)\,h\mathrm{d}h=0\quad\text{and}\quad\int p(\rho)\,\rho\mathrm{d}\rho=0\tag{C.65}$$

Substituting the above equations in Eqn. C.64, we can get

$$\iiint p(h)\,p(\rho)\,p(\theta)\,2h\rho\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)\left(\boldsymbol{n}_S\cdot\overrightarrow{\theta}\right)\mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta=0\tag{C.66}$$

- By utilizing Eqn. C.63, the last term in Eqn. C.61 can be written as the following:

$$\begin{aligned}&\iiint p(h)\,p(\rho)\,p(\theta)\,\rho^2\left(\boldsymbol{n}_S\cdot\overrightarrow{\theta}\right)^2\mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta\\&=\left[1-\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)^2\right]\int p(h)\,\mathrm{d}h\int p(\rho)\,\rho^2\mathrm{d}\rho\int_0^{2\pi}p(\theta)\cos^2\theta\mathrm{d}\theta\end{aligned}\tag{C.67}$$

Since $p(\theta)=1/(2\pi)$, we have

$$\int_0^{2\pi}p(\theta)\cos^2\theta\mathrm{d}\theta=\int_0^{2\pi}\frac{\cos^2\theta}{2\pi}\mathrm{d}\theta=\frac{1}{2}\tag{C.68}$$

Hence, Eqn. C.67 can be written as the following by utilizing Eqn. C.58:

$$\begin{aligned}&\iiint p(h)\,p(\rho)\,p(\theta)\,\rho^2\left(\boldsymbol{n}_S\cdot\overrightarrow{\theta}\right)^2\mathrm{d}h\mathrm{d}\rho\mathrm{d}\theta\\&=\frac{1}{2}\left[1-\left(\boldsymbol{n}_S\cdot\boldsymbol{V}_U\right)^2\right]r_U^2\end{aligned}\tag{C.69}$$

By combining Eqns. C.61, C.62, C.66, and C.69, the final equation for computing σ_{ds} can be written as the following:

$$\sigma_{ds} = \sqrt{\left(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\right)^2 D_U^2 + \frac{1}{2}\left[1 - \left(\boldsymbol{n}_S \cdot \boldsymbol{V}_U\right)^2\right] r_U^2} \tag{C.70}$$

Bibliography

[1] B. Breuckmann. Topometric sensors for prototyping and manufacturing. In *Proceedings of SPIE 2787*, pages 2–11, 1996.

[2] F. Chen, G. M. Brown, and M. Song. Overview of three-dimensional shape measurement using optical methods. *Optical Engineering*, 39(1):10–22, 2000.

[3] M. S. Mermelstein, D. L. Feldkhun, and L. G. Shirley. Video-rate surface profiling with acousto-optic accordion fringe interferometry. *Optical Engineering*, 39(1):106–113, 2000.

[4] M. A. Sutton, W. Zhao, S. R. McNeill, H. W. Schreier, and Y. J. Chao. Development and assessment of a single-image fringe projection method for dynamic applications. *Experimental Mechanics*, 41(3):205–217, 2001.

[5] S. R. McNeill, M. A. Sutton, Z. Miao, and J. Ma. Measurement of surface profile using digital image correlation. *Experimental Mechanics*, 37(1):13–20, 1997.

[6] C. P. Keferstein and M. Marxer. Testing bench for laser triangulation sensors. *Sensor Review*, 18(3):183–187, 1998.

[7] Z. Ji and M. C. Leu. Design of optical triangulation devices. *Optics and Laser Technology*, 21(5):335–338, 1989.

[8] C. S. Fraser and D. C. Brown. Industrial photogrammetry - new developments and recent applications. *Photogrammetric Record*, 12(68):197–217, 1986.

[9] C. S. Fraser. Photogrammetric measurement to one part in a million. *Photogrammetric Engineering and Remote Sensing*, 58(3):305–310, 1992.

[10] Photogrammetry information. http://www.geodetic.com/photogrammetry.htm.

[11] G. J. Swanson, M. P. Kavalauskas, and L. G. Shirley. High-precision surface profiling with broadband accordion fringe interferometry. In *Proceedings of SPIE 4189*, pages 161–169, 2001.

[12] X. Peng, J. D. Tian, P. Zhang, et al. Three-dimensional vision with dual acousto-optic deflection encoding. *Optics Letters*, 30(15):1965–1967, 2005.

[13] L. J. Hornbeck. Current status of the digital micromirror device (DMD) for projection television applications. In *Technical Digest - International Electron Devices Meeting*, pages 381–384, 1993.

[14] P. F. van Kessel, L. J. Hornbeck, R. E. Meier, and M. R. Douglass. A MEMS-based projection display. *Proceedings of the IEEE*, 86(8):1687–1704, 1998.

[15] E. U. Wagemann, M. Schönleber, and H. J. Tiziani. Grazing holographic projection of object-adapted fringes for shape measurements with enhanced sensitivity. *Optics Letters*, 23(20):1621–1623, 1998.

[16] L. J. Hornbeck. Digital Light Processing^{TM} for high-brightness, high-resolution applications. In *Proceedings of SPIE 3013*, pages 27–40, 1997.

[17] G. Sansoni, F. Docchio, U. Minoni, and C. Bussolati. Development and characterization of a liquid crystal projection unit for adaptive structured illumination. In *Proceedings of SPIE 1614*, pages 78–86, 1991.

[18] B. Breuckmann, F. Halbauer, E. Klaas, and M. Kube. 3D-metrologies for industrial applications. In *Proceedings of SPIE 3102*, pages 20–29, 1997.

[19] R. Legarda-Sáenz, T. Bothe, and W. P. Jüptner. Accurate procedure for the calibration of a structured light system. *Optical Engineering*, 43(2):464–471, 2004.

[20] V. Srinivasan, H. C. Liu, and M. Halioua. Automated phase-measuring profilometry of 3-D diffuse objects. *Applied Optics*, 23(18):3105–3108, 1984.

[21] B. A. Rajoub, D. R. Burton, and M. J. Lalor. A new phase-to-height model for measuring object shape using collimated projections of structured light. *Journal Of Optics A: Pure And Applied Optics*, 7(6):S368–S375, 2005.

[22] S. Toyooka and Y. Iwaasa. Automatic profilometry of 3-D diffuse objects by spatial phase detection. *Applied Optics*, 25(10):1630–1633, 1986.

[23] C. Quan, X. Y. He, C. F. Wang, C. J. Tay, and H. M. Shang. Shape measurement of small objects using LCD fringe projection with phase shifting. *Optics Communications*, 189(1–3):21–29, 2001.

[24] Q. Hu, P. S. Huang, Q. Fu, and F. Chiang. Calibration of a three-dimensional shape measurement system. *Optical Engineering*, 42(2):487–493, 2003.

[25] P. S. Huang, Q. Hu, and F. Chiang. Error compensation for a three-dimensional shape measurement system. *Optical Engineering*, 42(2):482–486, 2003.

[26] J. Heikkilä. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1066–1077, 2000.

[27] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[28] W. Nadeborn, P. Andrä, W. Jüptner, and W. Osten. Evaluation of optical shape measurement methods with respect to accuracy of data. In *Proceedings of SPIE 1983*, pages 928–930, 1993.

[29] R. Sitnik, M. Kujawińska, and J. Woźnicki. Digital fringe projection system for large-volume 360-deg shape measurement. *Optical Engineering*, 41(2):443–449, 2002.

[30] R. Sitnik and M. Kujawińska. Opto-numerical methods of data acquisition for computer graphics and animation systems. In *Proceedings of SPIE 3958*, pages 36–43, 2000.

[31] B. Han, D. Post, and P. Ifju. Moiré interferometry for engineering mechanics: current practices and future developments. *Journal of Strain Analysis for Engineering Design*, 36(1):101–117, 2001.

[32] B. Han, D. Columbus, Z. Wu, and J. Lu. Mechanical fringe shifting in moiré interferometry. *Experimental Techniques*, 23(1):16–19, 1999.

[33] I. Yamaguchi and T. Zhang. Phase-shifting digital holography. *Optics Letters*, 22(16):1268–1270, 1997.

[34] K. Hibino, B. F. Oreb, D. I. Farrant, and K. G. Larkin. Phase shifting for nonsinusoidal waveforms with phase-shift errors. *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, 12(4):761–768, 1995.

[35] Y. Surrel. Design of algorithms for phase measurements by the use of phase stepping. *Applied Optics*, 35(1):51–60, 1996.

[36] C. Joenathan. Phase-measuring interferometry: new methods and error analysis. *Applied Optics*, 33(19):4147–4155, 1994.

[37] C. R. Coggrave and J. M. Huntley. Optimization of a shape measurement system based on spatial light modulators. *Optical Engineering*, 39(1):91–98, 2000.

[38] B. Zhao. A statistical method for fringe intensity-correlated error in phase-shifting measurement: The effect of quantization error on the N-bucket algorithm. *Measurement Science and Technology*, 8(2):147–153, 1997.

[39] B. Zhao and Y. Surrel. Effect of quantization error on the computed phase of phase-shifting measurements. *Applied Optics*, 36(10):2070–2075, 1997.

[40] M. Kujawińska and W. Osten. Fringe pattern analysis methods: Up-to-date review. In *Proceedings of SPIE 3407*, pages 56–66, 1998.

[41] J. Batlle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: A survey. *Pattern Recognition*, 31(7):963–982, 1998.

[42] G. Sansoni, S. Corini, S. Lazzari, R. Rodella, and F. Docchio. Three-dimensional imaging based on gray-code light projection: characterization of the measuring algorithm and development of measuring system for industrial applications. *Applied Optics*, 36(19):4463–4472, 1997.

[43] H. Gärtner, P. Lehle, and H. J. Tiziani. New, highly efficient, binary codes for structured light methods. In *Proceedings of SPIE 2599*, pages 4–13, 1996.

[44] J. M. Huntley and H. O. Saldner. Temporal phase-unwrapping algorithm for automated interferogram analysis. *Applied Optics*, 32(17):3047–3052, 1993.

[45] H. O. Saldner and J. M. Huntley. Profilometry using temporal phase unwrapping and a spatial light modulator-based fringe projector. *Optical Engineering*, 36(2):610–615, 1997.

[46] H. O. Saldner and J. M. Huntley. Temporal phase unwrapping: application to surface profiling of discontinuous objects. *Applied Optics*, 36(13):2770–2775, 1997.

[47] H. Zhao, W. Chen, and Y. Tan. Phase-unwrapping algorithm for the measurement of three-dimensional object shapes. *Applied Optics*, 33(20):4497–4500, 1994.

[48] W. Nadeborn, P. Andrä, and W. Osten. A robust procedure for absolute phase measurement. *Optics and Lasers in Engineering*, 24(2–3):245–260, 1996.

[49] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[50] J. G. Fryer and D. C. Brown. Lens distortion for close-range photogrammetry. *Photogrammetric Engineering and Remote Sensing*, 52(1):51–58, 1986.

[51] C. Kolb, D. Mitchell, and P. Hanrahan. A realistic camera model for computer graphics. In *Proceedings of the 22nd annual conference on Computer Graphics and Interactive Techniques*, pages 317–324, 1995.

[52] T. A. Clarke and J. G. Fryer. The development of camera calibration methods and models. *Photogrammetric Record*, 16(91):51–66, 1998.

[53] J. Salvi, X. Armangué, and J. Batlle. A comparative review of camera calibrating methods with accuracy evaluation. *Pattern Recognition*, 35(7):1617–1635, 2002.

[54] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, RA–3(4):323–344, 1987.

[55] J. Heikkilä and O. Silvén. A four-step camera calibration procedure with implicit image correction. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1106–1112, 1997.

[56] Camera calibration toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/.

[57] Open Source Computer Vision Library. http://www.intel.com/research/mrl/research/opencv/.

[58] C. Reich, R. Ritter, and J. Thesing. 3-D shape measurement of complex objects by combining photogrammetry and fringe projection. *Optical Engineering*, 39(1):224–231, 2000.

[59] C. Reich. Photogrammetric matching of point clouds for 3D-measurement of complex objects. In *Proceedings of SPIE 3520*, pages 100–110, 1998.

[60] W. Schreiber and G. Notni. Theory and arrangements of self-calibrating whole-body three-dimensional measurement systems using fringe projection technique. *Optical Engineering*, 39(1):159169, 2000.

[61] W. Osten. Application of optical shape measurement for the nondestructive evaluation of complex objects. *Optical Engineering*, 39(1):232–243, 2000.

[62] R. Kowarschik, P. Kühmstedt, J. Gerber, W. Schreiber, and G. Notni. Adaptive optical three-dimensional measurement with structured light. *Optical Engineering*, 39(1):150–158, 2000.

[63] T. Peng, S. K. Gupta, and K. Lau. Algorithms for constructing 3-D point clouds using multiple digital fringe projection patterns. In *Computer-Aided Design and Applications*, volume 2, pages 737–746, 2005.

[64] T. Peng and S. K. Gupta. A computational framework for point cloud construction using digital projection patterns. In *Proceedings of IDETC/CIE 2006*, pages DETC2006–99553, 2006.

[65] P. S. Huang, C. Zhang, and F. P. Chiang. High-speed 3-d shape measurement based on digital fringe projection. *Optical Engineering*, 42(1):163–168, 2003.

[66] B. A. Barsky, D. R. Horn, S. A. Klein, J. Pang, and M. Yu. Camera models and optical systems used in computer graphics: Part I, object-based techniques. In *Lecture Notes in Computer Science 2669*, pages 246–255, 2003.

[67] P. S. Huang, Q. Hu, F. Jin, and F. P. Chiang. Color-encoded digital fringe projection technique for high-speed three-dimensional surface contouring. *Optical Engineering*, 38(6):1065–1071, 1999.

[68] N. Li, P. Cheng, M. A. Sutton, and S. R. McNeill. Three-dimensional point cloud registration by matching surface features with relaxation labeling method. *Experimental Mechanics*, 45(1):71–82, 2005.

[69] Verisurf®. CAD model based inspection software. http://www.verisurf.com/.

[70] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[71] Francis S. Hill Jr. *Computer Graphics Using OpenGL*. Prentice Hall, second edition, 2000.

[72] T. Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23(6):343–349, 1980.

[73] M. Subbarao and M. C. Lu. Image-sensing model and computer simulation for CCD camera systems. *Machine Vision and Applications*, 7(4):277–289, 1994.

[74] E. Krotkov. Focusing. *International Journal of Computer Vision*, 1(3):223–237, 1987.

[75] Jonathan Corney and Theodore Lim. *3D Modeling with ACIS*. Saxe-Coburg Publications, second edition, 2002.

[76] 3D ACIS modeler. http://www.spatial.com/components/acis/.

[77] Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall, second edition, 1995.

[78] C. P. Brophy. Effect of intensity error correlation on the computed phase of phase-shifting interferometry. *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, 7(4):537–541, 1990.