Deflectometric Measurement of the Topography of Reflecting Freeform Surfaces in Motion

Hanning Liang

Dissertation zur Erlangung des Doktorgrades
der Naturwissenschaften (Dr. rer. nat.)
eingereicht an der Fakultät für Informatik und Mathematik
der Universität Passau

Dissertation submitted to
the Faculty of Computer Science and Mathematics
of the University of Passau
in Partial Fulfillment of Obtaining
the Degree of a Doctor of Natural Sciences

1. Gutachter:	Prof. Dr. Tomas Sauer
2. Gutachter:	Prof. Dr. Christian Faber
3. Gutachter:	Prof. Dr. Johannes Wallner

Passau, 06.09.2021

Tag der mündlichen Prüfung: 24.02.2022

Abstract

A new deflectometric method for the measurement of specular free-form surfaces with strong surface structures in motion is presented. It is based on an existing method - single-shot phase-measuring deflectometry (SSPMD [TIK82][HNA11] [Liu+14]) - which has been modified and extended in several ways. In single-shot PMD, a pattern of additively superimposed sinusoidal fringes aligned perpendicularly to each other is reflected from the test object and recorded by a camera. The phases of the distorted cross-fringe-pattern observed by the camera are separated and evaluated by single-sideband demodulation. The three-dimensional shape of the surface is then obtained from the phase-coded correspondence between the observed pattern points and the associated camera points via ray tracing and subsequently via numerical integration of the normal vectors.

So far, the existing method could not be applied for objects with strong surface structures, since these generate - due to the strongly varying image scale - a globally broadband signal in the camera image, which cannot be evaluated by single-sideband demodulation. Furthermore, due to the height ambiguity ("height problem of deflectometry"), the surface normal vectors from the ray tracing could only be obtained after adding suitable additional measurements, e.g. a second camera image from another view (so-called "stereo deflectometry").

Both problems are solved in the present work. The problem of the global broadband signal - the missing localization of the spectrum of the distorted pattern in frequency space due to the locally too strongly varying fringe densities

- is solved by a local frequency evaluation. For this purpose, instead of the global frequency evaluation given by Fourier transform based single sideband demodulation, the localized ridgelet transform was used, which allows for a local frequency and phase determination for global broadband signals.

The height problem of PMD is solved by adapting and substantially extending an existing approach, used for isolated points and discrete patterns proposed in [SCP04] and [SCP05], for the case of phase-based deflectometric measurements. In contrast to conventional PMD evaluation, where only the screen-point-camera-point correspondence is used separately for each observed pixel, this new approach uses the local lateral correspondence in the screen plane and in the image plane, which is present in each PMD measurement but not considered in conventional PMD evaluation methods[1], as additional source of information. The adapted and extended algorithm proposed in this work allows to resolve the ambiguity from the spatial context of each measurement point without any further measurements (”monocular absolute deflectometry”).

Finally, to further increase flexibility, data should not only be obtained and merged from different camera views, but also from different object positions during the motion and even different measurement stations if needed. For this purpose, a multi-position and especially a multi-station deflectometry was developed, which allows for highly precise registration of data from different object positions within one measuring station and even of data from different measuring stations with different patterns using different fringe frequencies. This makes it possible to obtain a specific optimal fringe resolution in the camera image for different object areas, if required.

The combination of all these approaches allows for the measurement of specular objects even with strongly curved surface structures in motion, which has not been possible so far. Corresponding measurement examples on mirrors, smartphones and petrol cap with movement speeds of up to 200 mm/s will be shown.

---

[1] except for the numerical integration step required to obtain the object shape from the measured slopes.

Contents

Acronyms

2D	2 dimensional
3D	3 dimensional
STFT	Short time Fourier transform
WFT	Windowed Fourier transform
FT	Fourier transform
CWT	Continuous Wavelet transform
PMD	Phase measuring deflectometry
SNR	Signal-to-noise ratio
SSPMD	Single-shot PMD
SSB	Single-sideband modulation
CoC	Circle of confusion
PSF	Point spread function
GBT	Gabor transform
WVT	Wigner-Ville transform
WVD	Wigner-Ville distribution

ACF	Autocorrelation function
PWCT	Pseudo Wigner-Ville transformn
ST	Stockwell transform
RT	Ridgelet transform
FWHM	Full width at half maximum
CT	Chirplet transform
RBF	Radial basis function

Nomenclature

$.^*$	Complex conjugate
$.^T$	Matrix transpose
α	Chirp rate
$\hat{\mathbf{d}}$	Viewing direction
$\hat{\mathbf{n}}_r$	Surface normal
$\hat{\mathbf{u}}\hat{\mathbf{v}}\hat{\mathbf{w}}$	Local coordinate system on object surface
$\hat{\mathbf{x}}$	Normalized vectors in *2D* or *3D* space
$\hat{\mathbf{x}}_c\hat{\mathbf{y}}_c\hat{\mathbf{z}}_c$	Camera coordinate system
$\hat{\mathbf{x}}_i$	Incident ray
$\hat{\mathbf{x}}_r$	Reflection ray
$\hat{\mathbf{x}}_s\hat{\mathbf{y}}_s\hat{\mathbf{z}}_s$	Screen coordinate system
∇	Gradient operator
$\Phi_c(.)$	The unwrapped phase on the camera plane
$\phi_c(.)$	The wrapped phase on the camera plane
$\Phi_s(.)$	The unwrapped phase on the screen plane

$\phi_s(.)$	The wrapped phase on the screen plane
$\mathbf{A}$	Mapping matrix
$\mathbf{c}$	Projection center of the camera
$\mathbf{O}$	Patch
$\mathbf{r}_0$	Object point
$\mathbf{R}_c$	Camera rotation matrix
$\mathbf{r}_c$	Camera point
$\mathbf{R}_s$	Screen rotation matrix
$\mathbf{r}_s$	Screen point
$\mathbf{X}$	Tensors, matrices and *4D* vectors
$\mathbf{x}$	Points and vector in *2D* or *3D* space
θ	Angle of incidence
a	Wavelet dilation factor
b	Wavelet translation factor
c	Ridgelet translation factor in perpendicular direction
d	Determinant
f_u, f_v	Frequency variable in u, v direction
$g(.)$	Surface function in Monge form
I	The unique vector
i	Complex number $i = \sqrt{-1}$
I_c	Intensity function of the camera image

I_s	Intensity function of the screen pattern
I_{0s}	Background intensity of the screen pattern
I_{c0}	Background intensity of the camera image
K_c	Contrast of the camera image
K_s	Contrast of the screen pattern
l	Camera principal distance
M_c	Modulation of the camera image
M_s	Modulation of the screen pattern
x_c, y_c	Camera coordinate
x_s, y_s	Screen coordinate

Chapter 1

Introduction

1.1 Motivation

The topography measurement of specular free-form surfaces, such as varnished car bodies, eyeglass lenses or smartphone displays, is an important topic in the field of optical metrology. However, there is not yet a method that is capable of measuring specular objects with strong surface structures in motion quickly, robustly and in full-field.

Phase measuring deflectometry (PMD) has been an established method for the contactless optical measurement of the topography of specular surfaces for more than 20 years and is notable for its robustness, flexibility and high measurement dynamics. In principle, phase-measuring deflectometry has the potential to meet the above requirements. A disadvantage of this method, however, is that several images of a sinusoidal fringe pattern in different phase positions must be recorded for one measurement. During the measurement, the test object is not allowed to move. For many industrial applications - for example in the production with short cycle times - the necessity of stopping the object for inspection is a major disadvantage, even though the very few images required for evaluation can be captured with modern camera technology within the short cycle time of only a few seconds. Due to the large relative overhead caused by handling times for stopping, fixing and re-accelerating the object, however, the use of this measurement technique is not practical for many important applications. Therefore, the possibility to measure specular objects with strong surface structures in motion and in full-field is strongly desired.

1.2 Problem Statement

According to the state of the art, it is possible to perform a PMD measurement for "benign" object surfaces using only one single camera acquisition [TIK82][HNA11] [Liu+14]. Two sinusoidal fringe patterns aligned perpendicular to each other are additively superimposed and displayed on a screen. This pattern is reflected from the object surface and recorded by a camera. The pointwise correspondence between the screen plane and the camera plane is encoded in the phases of the fringe pattern. These are evaluated by single-sideband demodulation. The normal vectors of the surface are then obtained by ray tracing and the three-dimensional shape is determined by integrating the normal vectors obtained. However, so far this method does not work for objects with strong surface structures, since these generate a globally broadband signal that cannot be evaluated in the Fourier domain by single-sideband demodulation. To solve this problem, a new single-shot evaluation method for the phase determination using the localized ridgelet transform, which analyses the signal locally both in spatial and frequency domain, is presented.

The surface normal is determined using the correspondence between image plane and screen encoded by the observed fringe phases. However, since the emitted radiation of the screen is (and has to be) diffuse, the direction information of the incident ray from the screen is not available. Therefore, the surface normal, obtained by the law of reflection, can only be evaluated under a given height assumption. The (coupled) height and the normal of the surface can not be evaluated simultaneously and unambiguously (the so-called height problem of PMD). This problem can be solved with different approaches, but all of these approaches either require different types of mechanical adjustments to the PMD setup or are not suitable for practical use due to low signal-to-noise ratio. A new algorithm based on the adaption and substantial extension of the approach proposed in [SCP05][SCP04] for deflectometric measurements is presented, which applies not only the point to point correspondence, but also the local lateral correspondence information. This additional lateral correspondence information is present in each deflectometric measurement. This way, an absolute monocular PMD evaluation of both the slope and the height becomes possible without the necessity of any further measurements.

Finally, to improve the flexibility and also to extend the measuring field, which can be problematic for highly curved object surfaces, it should be possible to combine PMD data from different camera views and different measurement stations. Furthermore, for object surfaces with highly dynamical curvature variation, two fringe patterns with different frequencies could be required. However, switching the pattern during object movement is time consuming (much slower than camera shots). Therefore, it is technically preferable to have different patterns displayed in different measuring stations, if they exist. The data from different camera views and measuring stations must be fused. For this purpose, a new Multi-Position- and especially Multi-Station-Deflectometry was developed, which allows for a highly precise registration of data from different object positions within one measuring station and even of data from different measuring stations with different patterns in different fringe frequencies, appropriately combining the surface normal vectors obtained from all the measurements to determine the object surface.

1.3 Organization of Chapters

The principle and the state of art of conventional PMD and single-shot PMD will be briefly reviewed in Chapter 2.

In Chapter 3, the appropriate mathematical modeling of PMD signals and how the different models react differently to various physical influences will be presented. Specific limiting cases for single-shot PMD are discussed, which are of no concern for a phase shift evaluation but problematic for single-shot evaluation.

In Chapter 4, different time-frequency analyzing methods are discussed and a method suitable for PMD phase evaluation is presented. The algorithm used for the calculation of PMD phases is shown in detail and the limits of this evaluation are discussed.

In Chapter 5, the principle of the absolute evaluation for discrete pattern points and lines presented in [SCP05][SCP04] is briefly reviewed. After this, two important extensions of this evaluation for the case of PMD measurements are presented: from discrete evaluation to full-field lateral evaluation, and from line-dependent evaluation to line-free evaluation.

In Chapter 6, the algorithms used for Multi-View- and Multi-Station-PMD will be presented.

This thesis will be completed by Chapter 7, presenting a final summary, conclusion and outlook of this research.

Chapter 2

State of the Art

Phase-Measuring Deflectometry (PMD) [Häu99][PB97][KHL04b][Bot+04] is a frequently used method for the topographical measurement of specular free-form surfaces in industrial environments because of its low demands with respect to mechanical vibrations and temperature changes, and at the same time of its high accuracy and robustness.

In this chapter, the measuring principle, the existing problems and the drawbacks and limitations of PMD will be discussed.

2.1 Conventional PMD

A perfectly specular reflective surface is strictly speaking invisible. All we can see when looking at a specular surface is the (commonly distorted) image of the environment surrounding and illuminating the object - this is exactly the measuring principle exploited by PMD. For this purpose, a known (calibrated) diffuse environment in the form of a sinusoidal pattern as extended (and therefore spatially incoherent) light source is used. This differs from most of the other 3D metrology technologies, where a structured light source is *projected* onto the object surface (Fig. 2.1 on the left hand side). The necessity for this different approach is caused by the property of the object surface. Specular surfaces reflect the light only in a certain direction. If applying a projector, it is impossible to guarantee that all the reflected rays would enter the pupil of the observing camera, especially if the surface slope itself is unknown before the measurement (Fig. 2.1 in the middle). To ensure that all (or at least most of) the reflected rays can be captured by the camera, a large diffuse pattern should be utilized in PMD instead of a projector (Fig. 2.1 on the right hand side). The light emanating from the diffuse pattern is reflected by the (commonly curved) surface of the specular object and the image of the pattern is distorted according to the form of the object surface. The reflected pattern is recorded by a camera. From the information of these distorted pattern images and the known calibration data, the specular surface can be reconstructed in 3D.

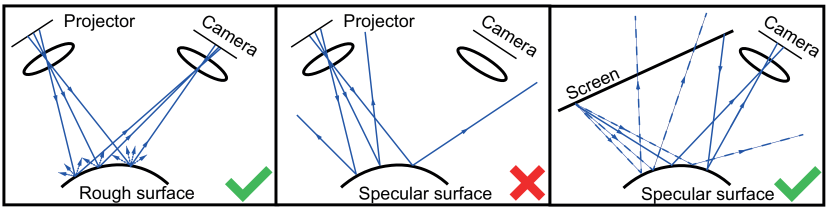


Fig. 2.1: The reflection property of rough and specular surface and the difference of applying a screen and a projector (only chief rays are shown).

2.1.1 Measuring Principle of PMD

A typical PMD setup is shown in Fig. 2.2. A PMD setup consists basically of a screen and a camera. A known sinusoidal fringe pattern in x- or in y-direction is displayed on the screen. The fringe patterns can be represented mathematically as:

$$I_{sx}(x_s, y_s) = I_{0s} + M_s(\cos(\Phi_{sx}(x_s))), with\ \Phi_{sx}(x_s) = 2\pi f_{sx} x_s + \phi_{x0} \qquad (2.1)$$

or

$$I_{sy}(x_s, y_s) = I_{0s} + M_s(\cos(\Phi_{sy}(y_s))), with\ \Phi_{sy}(y_s) = 2\pi f_{sy} y_s + \phi_{y0}, \qquad (2.2)$$

where (x_s, y_s) denotes the two dimensional coordinates of the plane defined by the screen, and $\Phi_{sx}(x_s), \Phi_{sy}(y_s)$ denote the phases of the fringe patterns in x_s and y_s direction respectively. f_{sx} and f_{sy} are the constant spatial frequencies of the fringes in x- and y-direction respectively. ϕ_{x0} and ϕ_{y0} are two constant phases that can be varied during a phase-shifting process, see Chapter 2.1.2. I_{0s} is the mean pattern intensity and M_s is the modulation. An example of the fringe pattern and the encoding phases are shown in Fig. 2.3. The phases $\Phi_{sx}(x_s)$ and $\Phi_{sy}(y_s)$ depend linearly on the screen coordinates and in this manner encode the screen position information. This pattern is reflected on the object surface and then captured by a camera. Analysing the phase of the resulting (distorted) fringe pattern in the camera image, the screen - camera (point to point) correspondence is determined. In conventional PMD, the phase of the reflected fringe is analyzed by applying the well-known phase-shifting method, which will be discussed in the next subsection.

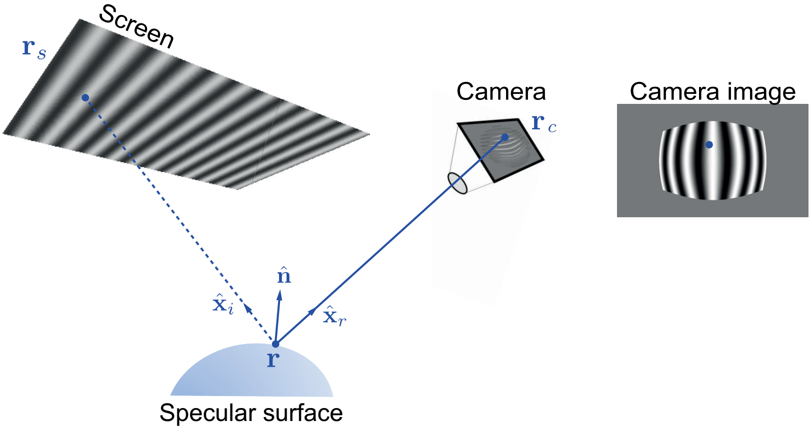


Fig. 2.2: Schematic representation of PMD. A sinusoidal fringe pattern is displayed on a screen. The camera records the pattern reflected by the specular object. The deformation of the pattern in the image is evaluated.

Taking this into account, PMD does not measure the height / the shape of the surface primarily. PMD actually measures predominantly the *normal* / the *slope* of the object surface. Utilizing the phase encoding, for each camera point $\mathbf{r}_c$ in the image plane (Fig. 2.2) with the associated reflection ray $\hat{\mathbf{x}}_r$ (known by camera calibration), the corresponding screen point $\mathbf{r}_s$ is determined. Applying the law of reflection, the surface normal $\hat{\mathbf{n}}$ at the reflecting surface point $\mathbf{r}$ is obtained[1]. Performing a 2D integration of the surface slopes recovered from the surface normals $\hat{\mathbf{n}}$, the 3D shape of the object surface is obtained. Performing

---

[1] This is a simplified description. Actually, to apply the law of reflection, the length s of the reflection ray $\hat{\mathbf{x}}_r$ or the direction of the incident ray $\hat{\mathbf{x}}_i$ must be given. However, neither s nor $\hat{\mathbf{x}}_i$ is available in PMD. This underdetermination is known as the "height problem" in PMD. This problem and its solution will be discussed in Chapter 2.1.4 and Chapter 5

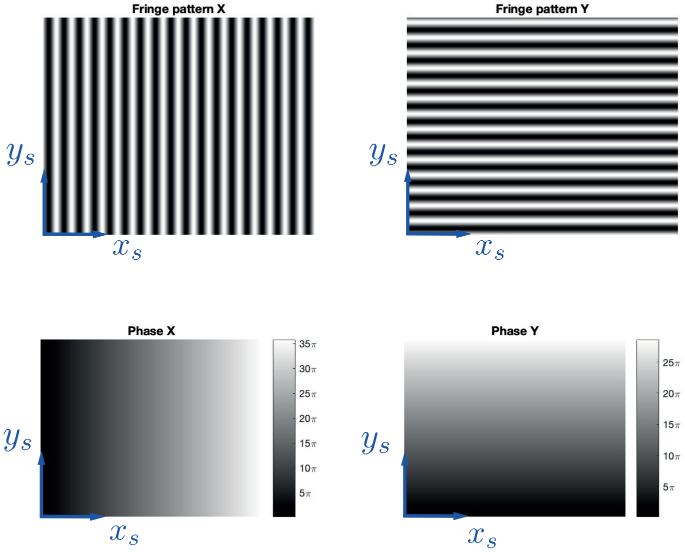


Fig. 2.3: Fringe pattern in x- and y- direction. Phase encoding in x- and y-direction.

a differentiation, the curvature map is produced. Measuring the surface slopes as primary measurand is in many ways advantageous compared to a direct measurement of the surface heights. For optically reflective surfaces, users are usually interested in their refractive power / the radius of curvature. To calculate the curvature, the height data must be differentiated twice, but the slope data only once. Note that each numeric differentiation increases the noise due to its high-pass characteristics and therefore worsens the measurement results.

2.1.2 Phase Shift

Conventionally, the phase of the observed fringe pattern in the camera image is obtained by sequential phase-shifting. The intensity distribution of the kth

($k = 1, 2, 3, 4...$) sequence can be written as [Kin+88][KHL04a]:

$$I_{ck}(x_c, y_c) = I_{c0}(x_c, y_c)(1 + K_c(x_c, y_c) \cos(\Phi_c(x_c, y_c) - \phi_{k_{shift}})), \tag{2.3}$$

where $I_{c0}(x_c, y_c)$ denotes the background intensity, $K_c(x_c, y_c)$ is the fringe contrast resp. visibility, $\phi_{k_{shift}}$ denotes the shifted phase in the kth sequence and $\Phi_c(x_c, y_c)$ is the encoded phase to be determined. Note that $I_{c0}(x_c, y_c)$ and $K_c(x_c, y_c)$ are only determined by the surface properties (roughness, reflectivity - see Chapter 3) and the illuminating situation. They do not change while shifting the phase. To solve for the phase $\Phi_c(x_c, y_c)$, at least three phase-shifts or more are needed since there are three unknowns $I_{c0}(x_c, y_c)$, $K_c(x_c, y_c)$ and $\Phi_c(x_c, y_c)$ in (2.3). By three-Phase-shifting (phase-shifting step: $\frac{2}{3}\pi$), the phase $\Phi_c(x_c, y_c)$ in wrapped form is determined by[KHL04a]:

$$\phi_c(x_c, y_c) = \text{atan2}(\sqrt{3}(I_{c2}(.) - I_{c3}(.)), 2I_{c1}(.) - I_{c2}(.) - I_{c3}(.)), \tag{2.4}$$

where the function atan2(Y, X) returns the four-quadrant inverse tangent of Y and X, which must be real and varies between $-\pi$ and π.

In practice, four-phase-shifting is frequently used because of its simplicity and robustness. By the well-known four-phase-shifting, also called the Carré method[Cre87][Car05], the phase is normally shifted by a quarter period each time, and the phase in wrapped form is then determined independently in each pixel by by the following equation:

$$\phi_c(x_c, y_c) = \text{atan2}(I_{c2}(.) - I_{c4}(.), I_{c1}(.) - I_{c3}(.)), \tag{2.5}$$

which is the conventional method used in PMD.

Note that the phase $\phi_c(x_c, y_c)$ in (2.4) and (2.5) is the wrapped phase, which repeats after every period and still differs from the monotonically increasing or decreasing phase $\Phi_c(x_c, y_c)$ to be determined to recover the absolute (and unique) screen coordinates (x_s, y_s) observed in (x_c, y_c).

2.1.3 Phase Unwrapping

An example of the phase calculated by (2.5) is shown in Fig. 2.4 on the left. The phase of the sinusoidal pattern repeats after each period and is in the range of $[0, 2\pi]$. This leads to an ambiguity problem: a value of $\phi_c(x_c, y_c)$ corresponds to many different locations in the screen; the screen points are not uniquely encoded. It is not possible to obtain the correct correspondence information without further measurements or assumptions. .

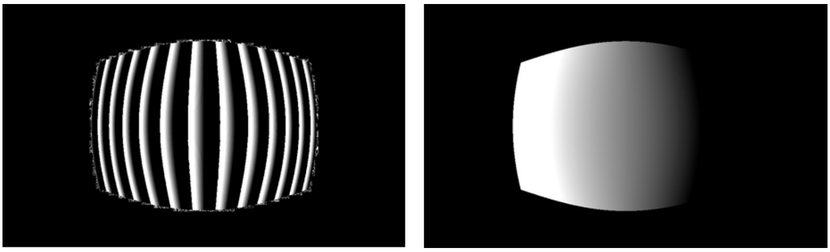


Fig. 2.4: Phase map before (left) and after (right) Unwrapping.

To determine the absolute phase, a so-called "unwrapping" procedure must be performed. The basic idea is to extend the uniqueness range to the whole screen (Fig. 2.4 right).

A possible solution to enlarge the uniqueness region is based on the so-called Chinese Remainder Theorem[Kos83]. In practice, sinusoidal patterns of two different frequencies are employed. Patterns with m and n fringes across the entire screen encode the screen points uniquely if m and n are positive integers and have no common divisors. Let $\phi_{cm}(x_c, y_c)$ denote the repeating phase map obtained with a fringe pattern with m fringes and $\phi_{cn}(x_c, y_c)$ the repeating phase map obtained with a fringe pattern with n fringes, the unique phase map is achieved by:

$$\Phi_c(x_c, y_c) = (p\phi_{cn}(x_c, y_c) + q\phi_{cm}(x_c, y_c))mod(mn), \tag{2.6}$$

with

$$\begin{aligned} p &= 1(mod(m)) \\ p &= 0(mod(n)) \\ q &= 1(mod(n)) \\ q &= 0(mod(m)). \end{aligned} \tag{2.7}$$

For example, if $m = 8$ and $n = 7$, it follows that $p = 49$ and $q = 8$. The application of the Chinese residual theorem is described in detail in [Hor98][Gru91][Lam03].

Another solution is combining a sequential gray code to uniquely encode the global correspondence and the phase-shifting to obtain the refined correspondence in high quality [SCR99][CXJ08]. In terms of the number of required exposures in one measurement, this procedure is not much different from the Chinese Remainder Theorem.

A further phase unwrapping solution is to encode the coarse global correspondence in a random pattern[LCZ14][LF18]. This is implemented by a pattern which is an additive superimposition of a sinusoidal fringe pattern for the periodical phase evaluation with a random, not modulated pattern for global mapping (phase unwrapping). These two patterns are (should be able to be) then separated in the Fourier domain. The uniqueness problem is solved here by adding a single random pattern, while with Chinese Remainder Theorem at least three additional exposures and with Gray coding even more are needed. However, the disadvantage of this method is the loss of gray level dynamics[2].

Another method is based on a lateral context approach, exploiting a smoothness assumption for the object surface (continuity of the surface slopes). In this case, no further measurement is required. The phase jumps of the demodulated phase are detected and removed: The phase values of two successive neighboring pixels are compared; If the phase difference is greater than a threshold value (approximate to 2π), 2π is added to all the subsequent phase values. This method is frequently applied to the single-shot PMD since it does not require any further

---

[2] This is especially disadvantageous in the case of single-shot deflectometry (which will be discussed in Chapter 2.2.1), since here half of the gray scale dynamics is already lost due to the additive cross grid pattern.

camera exposure and does not reduce the gray scale dynamics.

A comprehensive overview of these phase unwrapping methods is provided by [Zha18][JB94][SKS17].

2.1.4 Height Problem

The height problem of PMD [KHL04a], also called the regularization problem [Bal08], is caused by the unknown direction of the incident ray as shown in Fig. 2.5. Given a camera pixel point $\mathbf{r}_c$, the direction of the corresponding reflection ray $\hat{\mathbf{x}}_r$ is obtained according to the view direction of the camera (known by calibration). The screen point $\mathbf{r}_s$ is determined by the phase encoding. However, the direction of the incident ray $\hat{\mathbf{x}}_i$ is ambiguous since the screen is diffusely illuminating. For each point along the reflection ray $\hat{\mathbf{x}}_r$, a matching normal that satisfies the law of reflection for the given screen (source) point (x_s, y_s) can always be found. An unambiguous evaluation of the object height and the surface normal just using the observation made in a single point resp. pixel in the image plane is impossible [3].

One possible solution is to figure out the direction of the incident ray by displacing the screen. This solution is called specular surface triangulation and has been described by [BSG06][PR01] in 2006. The corresponding screen point is traced in both of the screen positions - before and after the displacement. According to these two screen points, the incident ray is fully determined and the specular surface can now be obtained by a triangulation evaluation. However, in this approach a high precision mechanical control of the screen displacement is required because the screen (embodying the reference standard of the measurement apparatus) must move from exactly the same position to the second position for each measurement, which is also very time-consuming.

Another solution has been proposed by [SH00][SH04]. Here, a collimating lens is placed in front of the screen and the screen is located in the focal

---

[3] It is possible to evaluate the surface height and the surface normal simultaneously and unambiguously by using the information of the neighboring camera pixel points and their equivalent on the screen. This will be shown in Chapter 5

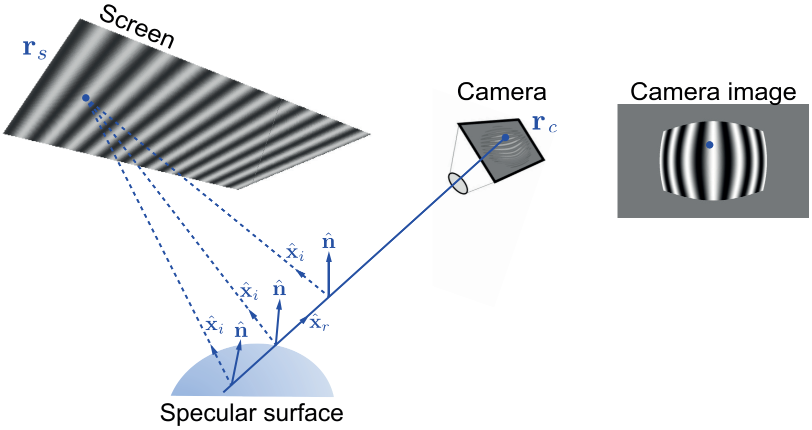


Fig. 2.5: Schematic representation of the height problem of PMD.

plane of this lens imaging the screen to infinity and thus assigning all screen points to a corresponding unique incident direction. The reflection is detected by telecentric observation optics, whereby the lateral position of the object points is directly encoded into the position of the image points. If a camera point and the corresponding screen point are given, the incident ray direction and the reflected beam direction (always parallel to the optical axis) are determined. Thus, the surface normal can be evaluated unambiguously. However, telecentric optics are usually bulky and expensive and also severely limits the measurement field and angular dynamics.

The most widely used approach is stereo PMD [KHL04a][Kic07][SB09]. As shown in Fig. 2.5, the height and the normal of the object surface can not be determined simultaneously and unambiguously since neither the incident ray direction $\hat{\mathbf{x}}_i$ nor the length of the reflection ray is known. However, under an assumption of the surface height, a proper surface normal satisfying the law of reflection can be determined for each such assumption - these normals are denoted as "*potential normals*" resulting in a vector field within the measurement volume in [KHL04a]. Placing a second camera with a different perspective, a second vector field can be defined as shown in Fig. 2.6. The actual object surface is determined by comparing both of the vector fields.

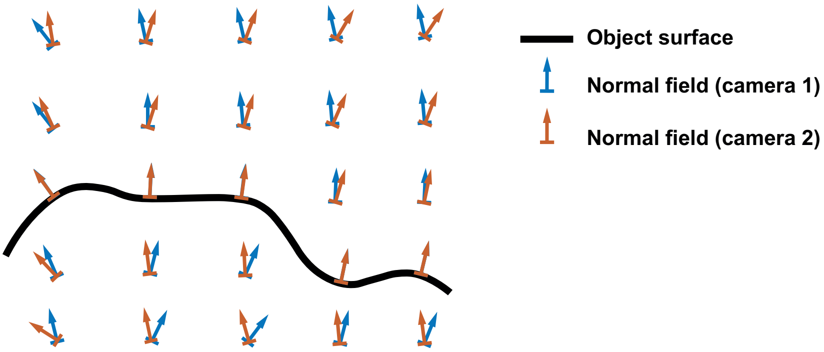


Fig. 2.6: Potential normals of two cameras. Under certain conditions, the normals only agree at the actual surface location of the object. Author's own representation according to [KHL04a], Fig. 3.6

In this approach a second camera is needed. Nowadays, adding a second camera is usually not considered as a financial nor technical drawback anymore. However, it is not feasible in all cases, such as smartphone or tablet PMD with only one front camera [Wil+19][Wil+20]. Additionally, a large overlap of both fields of view is required, and at the same time a sufficiently large difference of the viewing directions is necessary. This limits the practical applications considerably.

Alternatively, it is also possible to determine the height of one object point by using an additional sensor and calculate the rest of the surface by self-consistent iterative integration [SB09]. The drawback of this approach is of course that a second sensor is required and that the global accuracy depends on the accuracy of this single point.

It was also attempted to find the absolute object surface from one single camera perspective by analysing the local integrability exploiting the so-called "Frobenius condition" for the vector field of potential normals [Kam04a]. Theoretically, an absolute object surface can be found where the local integrability condition is best fulfilled. However, this approach is not suitable for practical use since the vector field only varies extremely slightly for typical geometries so that the Frobenius condition is approximately fulfilled almost everywhere, which results in a very poor SNR.

2.2 Single-shot PMD

[TIK82] applied the *Single-Sideband Demodulation* (SSB) to analyse the phase of interferometric fringe patterns from one single image. [HNA11][Liu+14] utilized this phase measuring method in PMD and introduced a new method which calculates the fringe phase in two directions based on only one single camera image, called *Single-Shot PMD* (SSPMD). This allows for measurement in motion since only one camera exposure is required. However, the frequency dynamics of the distorted fringe pattern and therefore the allowed height and slope dynamics of the object surface are severely restricted in order to be able to use this approach.

2.2.1 Principle

SSPMD utilizes the additive superposition of two perpendicular fringe patterns (Fig. 2.7) in order to encode the two screen components in only one pattern, which is called a “cross fringe pattern” and mathematically represented as:

$$I_s(x_s, y_s) = I_{0s} + \frac{\cos(\Phi_{sx}(x_s)) + \cos(\Phi_{sy}(y_s))}{2} M_s. \tag{2.8}$$

(x_s, y_s) denotes the screen coordinate. The phases $\Phi_{sx}(x_s)$ and $\Phi_{sy}(y_s)$ are linearly depending on x_s and y_s respectively. Therefore, they can again be written as $\Phi_{sx}(x_s) = 2\pi f_{sx} x_s + \phi_{x0}$ and $\Phi_{sy}(y_s) = 2\pi f_{sy} y_s + \phi_{y0}$, where f_{sx} and f_{sy} are two constant spatial frequencies in the respective direction.

The cross fringe pattern is reflected on the object surface and recorded by the camera as shown in Fig. 2.8. As the pattern is not strongly deformed by the object surface, and the fringe frequency and the fringe direction of the camera

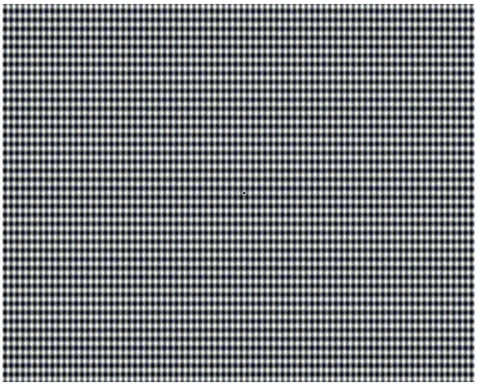


Fig. 2.7: The cross fringe pattern.

Fig. 2.8: The camera image of the cross fringe pattern reflected on a mirror.

image are varying only slightly, the camera image of the reflected pattern can be written as:

$$\begin{aligned} I_c(x_c, y_c) =& I_{c0}(x_c, y_c) \\ &+ \frac{M_c(x_c, y_c)}{2} \left(\exp\left(i\Phi_{cx}(x_c, y_c)\right) + \exp\left(i\Phi_{cy}(x_c, y_c)\right)\right) \\ &+ \frac{M_c(x_c, y_c)}{2} \left(\exp\left(-i\Phi_{cx}(x_c, y_c)\right) + \exp\left(-i\Phi_{cy}(x_c, y_c)\right)\right), \end{aligned} \tag{2.9}$$

with

$$\begin{aligned} \Phi_{cx}(x_c, y_c) &= 2\pi f_{cx} x_c + \delta\phi_{cx}(x_c, y_c) \\ \Phi_{cy}(x_c, y_c) &= 2\pi f_{cy} y_c + \delta\phi_{cy}(x_c, y_c) \end{aligned} \tag{2.10}$$

denoting the encoded phases to be determined, where f_{cx} and f_{cy} designate the spatial-carrier frequencies in x- and y-direction, which are assumed to be constants, $\delta\phi_{cx}(x_c, y_c)$ and $\delta\phi_{cy}(x_c, y_c)$ denote the local phase deformation introduced by the unknown object surface form, $I_{c0}(x_c, y_c)$ represents unwanted irradiance variations as a nuisance variable arising from the nonuniform illuminating direction and reflectivity variations of the surface, and $M_c(x_c, y_c)$ represents both reflectivity variations and variations of the surface roughness of the test object. In most cases $I_{c0}(x_c, y_c)$, $M_c(x_c, y_c)$, $\delta\phi_{cx}(x_c, y_c)$ and $\delta\phi_{cy}(x_c, y_c)$ vary slowly compared with the spatial carrier frequency $2\pi f_{cx} x_c$ and $2\pi f_{cy} y_c$. The Fourier transform (FT) of (2.9) can be presented as approximated to):

$$\begin{aligned}\mathcal{I}(f_u, f_v) = \mathcal{I}_0(f_u, f_v) + \mathcal{M}(f_u - f_{cx}, f_v) + \mathcal{M}(f_u, f_v - f_{cy}) +\\ + \mathcal{M}^*(f_u - f_{cx}, f_v) + \mathcal{M}^*(f_u, f_v - f_{cy}),\end{aligned} \tag{2.11}$$

with f_{cx} and f_{cy} being clearly separated from the DC term in Fourier space (Fig. 2.9). Applying two suitable windows to the carrier frequencies in the sequence - such as the window highlighted in red and green in Fig. 2.9 - and calculating the inverse Fourier transform in each case, the analytical signals $\frac{M_c(x_c, y_c)}{2} \exp\left(-i\Phi_{cx}(x_c, y_c)\right)$ and $\frac{M_c(x_c, y_c)}{2} \exp\left(-i\Phi_{cy}(x_c, y_c)\right)$ are obtained separately. Note that in this stage the unwanted (low-frequency) background variation $I_{c0}(x_c, y_c)$ has been filtered out. The phases $\Phi_{cx}(x_c, y_c)$ and $\Phi_{cy}(x_c, y_c)$ determined by the inverse Fourier transform are shown in Fig. 2.10. Unwrapping the phase and applying the point specially marked in the middle of screen to match the phase obtained from the camera image and the phase of the pattern displayed on the screen, the unambiguous correspondence between the camera points and the screen points is determined.

The basic idea of single-shot PMD was to apply a cross fringe pattern to overcome the sequential acquisition of the two phase components, and to overcome the phase shifting sequence by evaluating the fringe image via SSB in the Fourier domain.

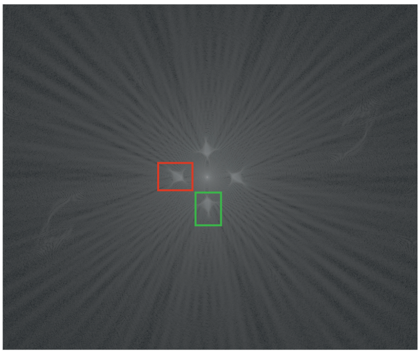


Fig. 2.9: Fourier spectrum of the camera image displayed in Fig. 2.8.

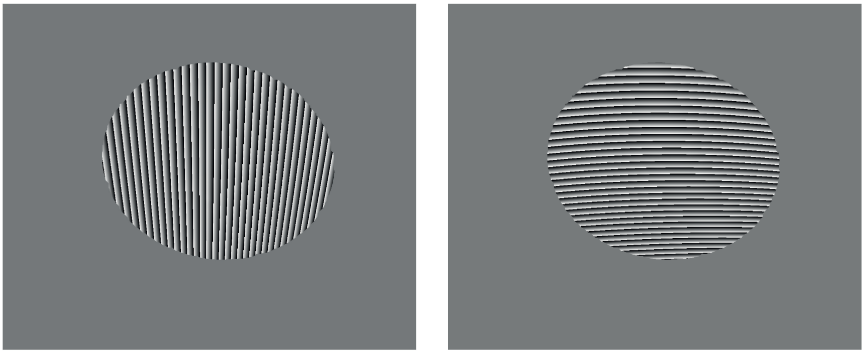


Fig. 2.10: The phase in x- and y- directions calculated using the method introduced in [TIK82] [HNA11][Liu+14].

2.2.2 Single-shot-PMD with Predistorted Pattern

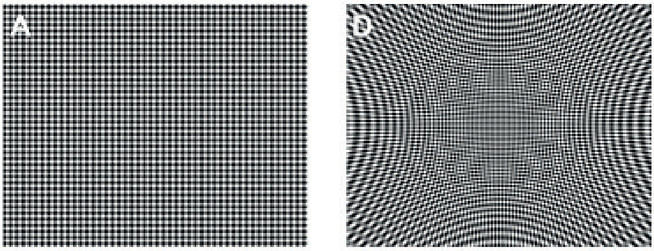
The single-shot method presented in [HNA11][Liu+14] is only suitable when the specular surface is smooth and does not have a strong surface structure. If a specular sphere with 7.5 mm radius ([Lia+16]) is under test, the cross fringe pattern reflected on the object surface is significantly distorted (Fig. 2.11, B). The fringe direction varies strongly and the fringe frequency can not be assumed to be a constant any more. Its Fourier spectrum as shown in Fig. 2.11, C indicates as well that the carrier frequency is not narrow band. A further disadvantage of the strong distortion is that the peripheral area of the observed fringes displays a low contrast introduced by the high frequency.

Both of these problems can be overcome by an appropriate predistortion of the input fringe pattern. If the expected shape of the object surface is known in advance, an input fringe pattern can be designed by ray tracing, so that its reflection on the object surface generally results in a narrowband fringe pattern. Such a pre-distorted fringe pattern, calculated for a spherical surface with 7.5 mm radius is shown in Fig. 2.11, D, with its *regular high-contrast* reflected image displayed in Fig. 2.11, E. Its narrow band Fourier spectrum is displayed in Fig. 2.11, F and now a suitable window can be applied to the carrier frequency.

2.2.3 Problems and Limitations of the Single-Shot PMD

Single-shot PMD is well-established to measure specular surfaces with not too strongly varying slopes. However, this method is not suitable to evaluate the phase of cross fringe patterns reflected on a broadband object surface. An example of such a “broadband object” - a red petrol cap with a ridge - is shown in Fig. 2.12 left, and the cross fringe pattern reflected on this object surface is displayed in Fig. 2.12. The frequency of the fringe pattern in y-direction varies strongly in the ridge area.

As the Fourier spectra of the observed fringe pattern displayed in Fig. 2.13, the observed frequencies vary considerably, so that there can not be assumed as a constant “carrier frequency”to be used for the demodulation any more. therefore, in the y-direction, the carrier frequency f_{cy} can not be separated from the base band $\mathcal{A}(f_u, f_v)$ appropriately. This illustrates the fact that the conventional single-shot method is not suitable for the PMD measurement of a broadband object.



regular fringe pattern

pre-distorted fringe pattern

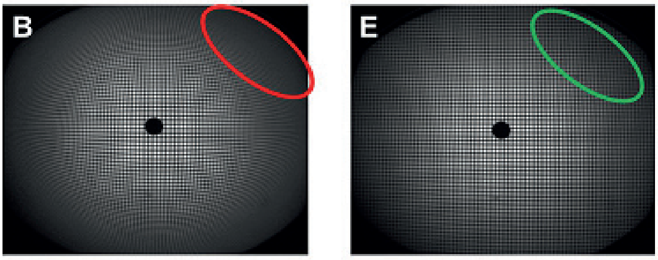
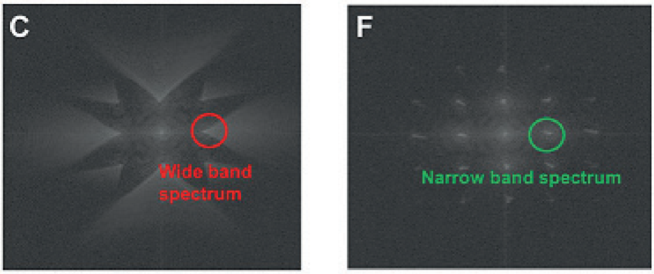


image of reflected regular pattern

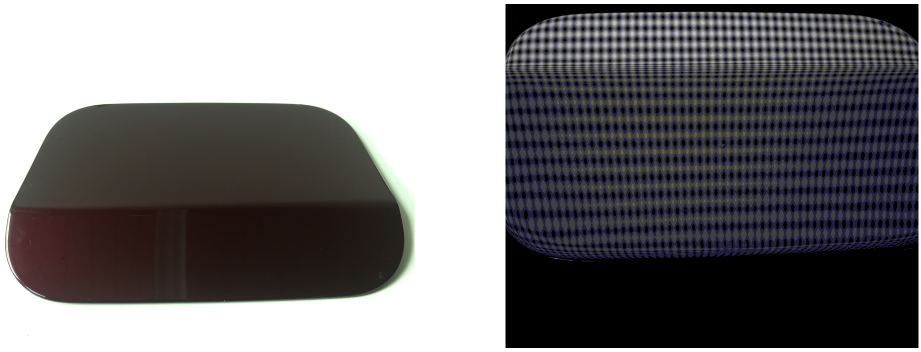
image of reflected pre-distorted pattern



Fourier spectrum of reflected regular pattern

Fourier spectrum of reflected pre-distorted pattern

Fig. 2.11: Projected patterns, camera images, and Fourier spectra of the camera images. In the left column a regular pattern is applied (A, B, C), and in the right column a pre-distorted pattern is applied (D, E, F). The original Fig. 3 in [Lia+16].



(a) A petrol cap with strong surface structures.

(b) The cross fringe pattern reflected by the object surface.

Fig. 2.12: Broadband object.

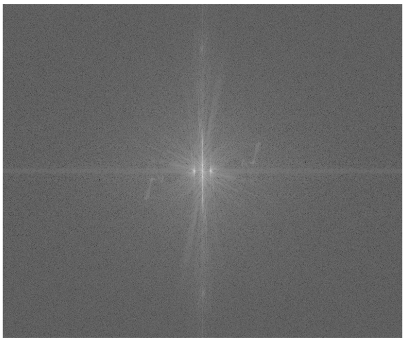


Fig. 2.13: Fourier spectrum of the camera image shown in Fig. 2.12 (b).

Chapter 3

Signal Generation and Limits of SSPMD

It happens that the same signal can be described with different mathematical models. For example, two additive signals $\sin 2\pi f_1 x + \sin 2\pi f_2 x$ can be interpreted as one signal of average frequency $\sin(\pi f_1 x + \pi f_2 x)$ with its amplitude modulated by $2\cos(\pi f_1 x - \pi f_2 x)$. Both models are mathematically equivalent. But depending on the application, one may make more physical sense than the other in a given context

Similar to the example discussed above, a sinusoidal signal can be expressed as

$$I_s(x_s) = I_{s0}(x_s) + M_s(x_s)\cos(\Phi_s(x_s)) \tag{3.1}$$

with $I_{s0}(x_s)$ denoting the DC term and $M_s(x_S)$ representing the amplitude or "modulation". This signal can also be represented as

$$I_s(x_s) = I_{s0}(x_s)(1 + K_s(x_s)\cos(\Phi_s(x_s))), \tag{3.2}$$

where $K_s(x_s)$ represents the contrast or "visibility". For phase shift evaluation, both descriptions are completely equivalent, since the essential information is provided by the sequential phase $\Phi_s(x_s)$. In contrast to the phase shift, the single-shot phase evaluation is based on contextual information. Physical effects such as surface reflectivity jumping or background material changes do not have a strong influence on the phase shift evaluation (stronger noise), while in such situations single-shot phase evaluation is not possible at all.

In this chapter, it will be discussed how such physical effects are manifested in the mathematical model.

3.1 Signal Modelling and Physical Influencing Variables

3.1.1 Contrast

Considering a screen displaying a perfect sinusoidal pattern with the maximum intensity $I_{max} = 0.8$ and the minimum intensity $I_{min} = 0.2$ as shown in Fig.

3.1, the DC term $I_{s0}(x_s)$, the contrast $K_s(x_s)$ and the modulation $M_s(x_s)$ are determined by $I_{s0(}x_s) = \frac{I_{max}+I_{min}}{2}$, $K_s(x_s) = \frac{I_{max}-I_{min}}{I_{max}+I_{min}}$ and $M_s(x_s) = \frac{I_{max}-I_{min}}{2}$.

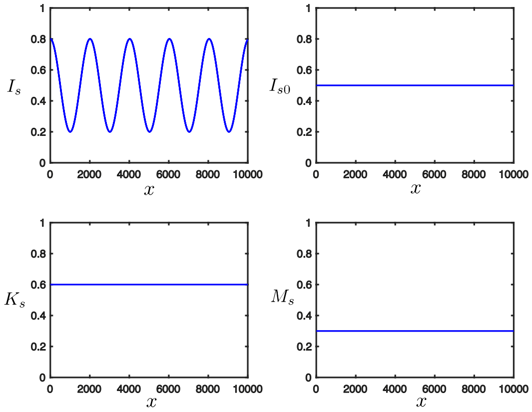


Fig. 3.1: A perfect sinusoidal signal with the maximum intensity $I_{max} = 0.8$ and the minimum intensity $I_{min} = 0.2$; The DC term of the signal $I_{s0}(x_s) = 0.5$; The contrast of the signal $K_s(x_s) = 0.6$; The modulation of the signal $M_s(x_s) = 0.3$.

However, even if this pattern is reflected by a perfect planar mirror and is then captured by a perfect camera, the recorded signal is never the same as $I_s(x_s)$ shown in Fig. 3.1. The radiance of the screen as a function of the angle between the normal of the screen surface and the direction of observation resp. object illumination changes the signal characteristic.

Denoting the total emitted radiation power of a screen pixel as Φ_e $[W]$ as shown in Fig. 3.2, the part of the radiant flux which is emitted into a small solid angle $\mathrm{d}\Omega$ is called the radiant intensity and denoted as $I_e = \frac{d\Phi_e}{\mathrm{cd}\Omega}$ $[W/sr]$. The radiant intensity of most of the radiators depends on the angle between the normal of the screen plane and the direction of the emitted light - β, and can be written as:

$$I_e(\beta) = I_0 \cos^m \beta. \tag{3.3}$$

For $m = 1$, it is a lambertian radiator. A lambertian radiator always has the same brightness, no matter under which direction it is observed, since its radiance, which is generally defined as

$$L_e(\beta) = \frac{dI_e}{\cos\beta dA} \tag{3.4}$$

with A being the pixel area, is a constant $\frac{I_0}{A}$. Another kind of radiator is radiator with a preferential direction (in German: Keulenstrahl) with $m = 3$. The radiance $L_e(\beta) = \frac{I_0 \cos^2\beta}{A}$ is directional. Most of the light is emitted to small angles β and the radiant intensity is the strongest while β equals 0.

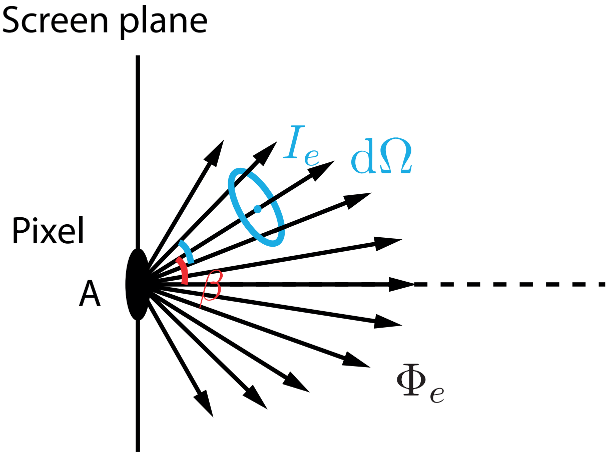


Fig. 3.2: The radiometric parameters of a screen pixel as a light source.

Most of the conventional (LCD) screens are neither perfectly lambertian nor radiators with a preferential direction with exactly $m = 3$. Usually, it is something in between. The radiance characteristic, usually called viewing angle characteristic, is different from screen to screen, and also different from the meridional to sagittal direction. In screen production, the viewing angle characteristics of a screen are measured with the conoscope lens [YG09] or the method proposed in [RKW06], and recorded in the attached data sheet. If the radiance characteristic of the screen is given as a (normalized) function $r(\beta(x_s))$, the signal can be represented as

$$I_s(x_s) = r(\beta(x_s))I_{s0}(x_s) + r(\beta(x_s))M_s(x_s)\cos 2\pi f x_s \tag{3.5}$$

or

$$I_s(x_s) = r(\beta(x_s))I_{s0}(x_s)(1 + K_s(x_s)\cos 2\pi f x_s). \tag{3.6}$$

As shown in Fig. 3.3, the mean value (DC term) and the modulation of the signal are changed according to the radiance condition, while the contrast of the signal remains.

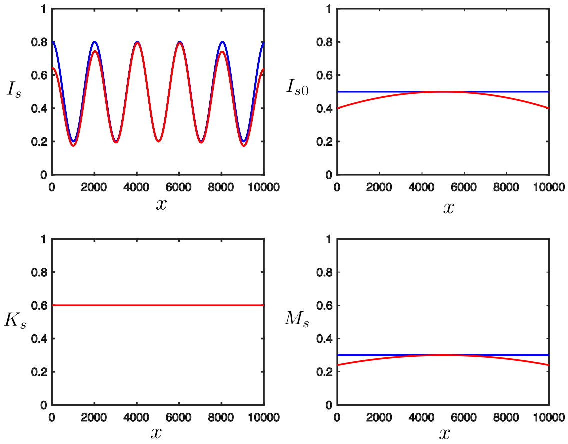
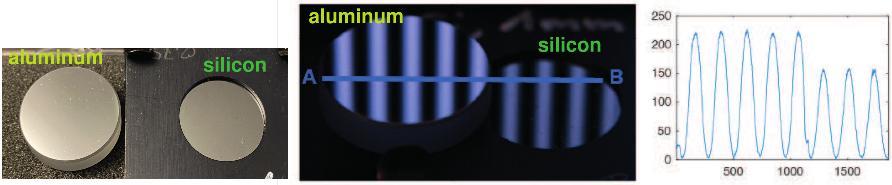


Fig. 3.3: The influence of the viewing angle on the signal.

In addition to the viewing angle characteristic of the screen, the reflectivity of the object surface has the same influence on the signal. Based on the Fresnel equations resp. Schlick's approximation [Sch94], the reflectivity of a reflecting surface can be written as a function of the angle of incidence. The intensity of the signal reflected from the surface is, same as (3.5) and (3.6), obtained by multiplying this function by the signal. Besides angle of incidence, reflectivity is also a material property of the surface. Examples of an aluminum-coated mirror with high reflectivity and a 1 mm-thick silicon with low reflectivity are shown in Fig. 3.4. Both of the mirrors are specular, s. Fig. 3.4 (a). At an angle of incidence of 45°, the reflectivity of the aluminum-coated mirror is about 93% and the reflectivity of the 1 mm-thick silicon is between 25% and 50%, depending on polarization. The angle of incidence of the fringe pattern captured in Fig. 3.4 (b) is about 30°. The modulation M_s, as shown in Fig. 3.4 (c), is reduced by the 1 mm thick silicon, while the contrast K_s remains in both cases above 93%[1].



(a) (b) (c)

Fig. 3.4: (a): The camera image of a specular aluminum-coated mirror and a specular 1 mm-thick silicon. (b): A sinusoidal fringe is reflected from the aluminum-coated mirror and from the 1 mm-thick silicon. (c): The intensity of the section A-B.

For those applications where the surface reflectivity strongly varies or the screen viewing angle characteristics is far from a lambertian radiator, the model $I_s(x_s) = I_{s0}(x_s) + M_s(x_s)\cos(\Phi_s(x_s))$ would make more sense as the modulation changes quickly and the contrast preserves.

3.1.2 Modulation

Suppose that the pattern is displayed on a perfect screen, such as a lambertian radiator, but there is an "offset background illumination" in the room that shines past the object into the camera. This kind of "offset", as a value added to the

---

[1] I_{max} and I_{min} from the aluminum-coated mirror surface are about 220 and about 6, respectively. The contrast $K_s = \frac{I_{max} - I_{min}}{I_{max} + I_{min}}$ is 94.69%; I_{max} and I_{min} from the 1 mm-thick silicon surface are about 157 and about 5, respectively. The contrast K_s is 93.83%

signal, changes the average intensity of the signal and the contrast of the signal, but not the modulation of the signal, as shown in Fig. 3.5.

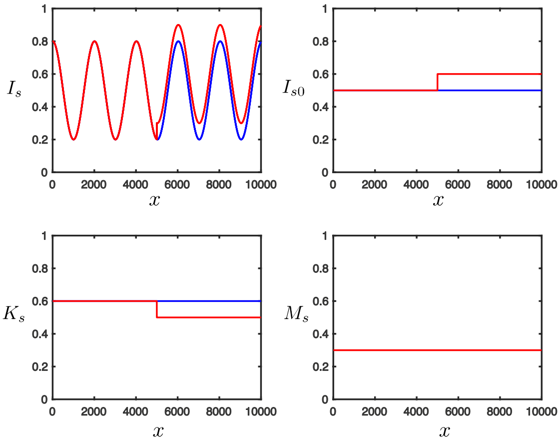


Fig. 3.5: The influence of the background light on the signal.

A “background material change” has the same influence on the signal as the background illumination. Consider a smartphone display as shown in Fig. 3.6, the sinusoidal pattern is reflected on a glass pane (the first layer). The color of the material under the glass pane changes from black (screen) to white (frame). The higher reflectivity of the white frame introduces a constant offset to the reflected signal. As the intensity of the section A-B shown on the right, the average intensity changes, while the modulation of the signal remains the same. This “jumping” effect shown in Fig. 3.6 on the right hand side will cause serious problems in the single-shot evaluation since the lateral context information is not continuous at this jumping position. However, this is not a problem in the phase-shifting evaluation since only the sequential information is applied. For these situations the model $I_s(x_s) = I_{s0}(x_s)(1 + K_s(x_s)\cos(\Phi_s(x_s)))$ is more sensible.

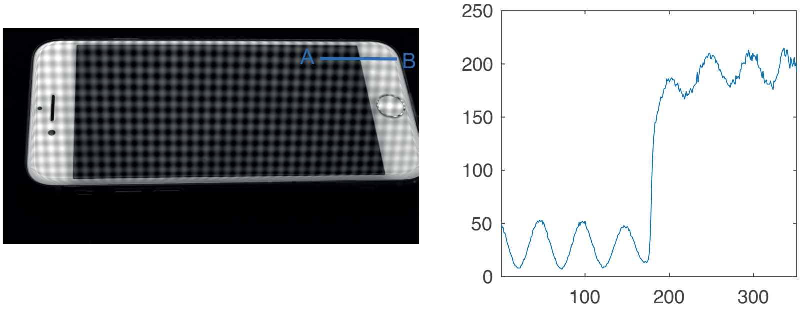


Fig. 3.6: A crossed sinusoidal signal reflected on a smartphone display with black and white background color. The intensity of the section A-B is plotted on the right.

3.2 Physical Limits of Single-Shot PMD

3.2.1 Nyquist-Shannon Sampling Limit

When using conventional phase shift evaluation, at least six, normally eight camera images have to be recorded in each measurement, while in the single-shot evaluation only one is sufficient. As to be expected, the considerable reduction of the required amount of data leads to certain restrictions and limitations.

In the conventional phase shift method, the phase information is obtained by the sequential intensity variation independently of the neighboring pixels. In comparison, in single-shot methods, the phase information is evaluated by exploiting lateral context information. Therefore, the lateral sampling must satisfy the condition of the Nyquist-Shannon sampling theorem [Nyq28]. An example is shown in Fig. 3.7. The object under test has a very small radius of curvature, resulting in a very high fringe frequency (as the screen is extremely demagnified), so that the sampling distances in the screen are greater than half of the fringe period. In this example, the sampling distance is exactly 2π. The recorded intensity in the single-shot camera image is displayed in the upper right corner in Fig. 3.7. The phase of the observed fringe pattern can not be evaluated as the sampling theorem is violated.

However, this case is not problematic for phase shift evaluation, since the phases, as shown in (2.4) and (2.5), are evaluated not by using the lateral context, but by using the sequential contrast (Fig. 3.7 bottom right).

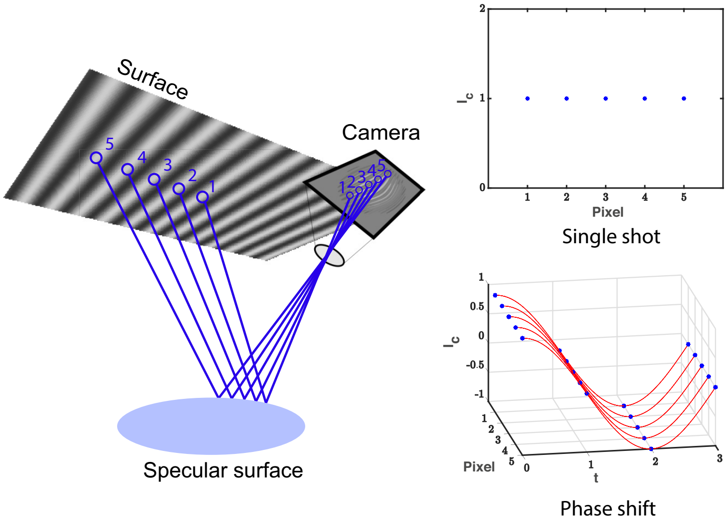


Fig. 3.7: Information acquisition of single-shot PMD and phase shifting PMD: Different effects of violating the sampling theorem.

3.2.2 Local Convergence

The object to be inspected is an optically effective component, acting as a (generally stigmatic) lens. If this lens converges all the camera chief rays backwards to a common screen point, all camera points observe the same screen point and the context information is completely lost (More generally, if all camera rays converge on the same screen line, the lateral context is degenerated in one dimension.). A single-shot evaluation based on the context information is not possible either. Therefore for single-shot evaluation, it is essential to understand when this situation will appear. This happens when the screen distance equals the image distance when considering the camera entrance pupil as the object to be "imaged" and the surface to be measured as the imaging lens. The camera chief rays are under this consideration regarded as the aperture rays. The image distance of the camera entrance pupil can be obtained by applying the lens equation $\frac{1}{o} + \frac{1}{i} = \frac{1}{f}$ in the imaging optic[2].

This critical geometry can now be interpreted mathematically. Consider the case of a coaxial PMD, where the screen, the camera, and the local surface normal are all coaxial (Fig. 3.8). For this case, the lens equation can be applied without further adaption. Applying a local coordinate system $\hat{\mathbf{u}}\hat{\mathbf{v}}\hat{\mathbf{w}}$ to the object surface in which the $\hat{\mathbf{w}}$-axis is coaxial with the surface normal, the local surface can be represented in the second fundamental form $w = \frac{a}{2}u^2 + buv + \frac{c}{2}v^2$. Representing the local object surface by using the Weingarten matrix[Küh08]

$$W = \begin{bmatrix} a & c \\ c & b \end{bmatrix}, \tag{3.7}$$

the principal curvatures of the surface is given by its eigenvalues:

$$k_{1,2} = \frac{a+b}{2} \pm \sqrt{\frac{(a-b)^2}{4} + c^2}. \tag{3.8}$$

For a rotationally symmetric surface $(a = b, c = 0)$, the curvature remains the same in all directions regardless of the cross section with the $\hat{\mathbf{w}}$-axis. Denoting the distance from screen to object as r_{s0} and the distance from the camera to the object as s, the converging condition, or the critical geometry condition is given by

$$\frac{1}{s} + \frac{1}{r_{s0}} = \frac{1}{f} = 2a = 2b \tag{3.9}$$

with f denoting focal length and $f = \frac{1}{2a} = \frac{1}{2b}$[3]. In this case all the chief rays of the camera converge to a point on the screen as shown in Fig. 3.8 on the left. For a general parabolic surface that is rotationally asymmetric, the surface

---

[2] In the imaging optics, o generally denotes the object distance, i denotes the image distance, and f denotes the focal length of the effective lens provided by the curved object surface.

[3] The focal length of a spherical mirror is half of the sphere radius.

curvature differs in each cross section with the $\hat{\mathbf{w}}$-axis. The cross section with the maximum curvature has the strongest optical power and the minimum curvature has the lowest optical power. As a result, the camera rays never converge to a point, but to two focal lines (corresponding to the two principal curvatures) in the respective distance. (3.9) can be rewritten as:

$$\frac{1}{s} + \frac{1}{r_{s0}} = a + b \pm \sqrt{(a-b)^2 + 4c^2}. \tag{3.10}$$

An example is shown in Fig. 3.8 on the right, which is called astigmatism in optics.

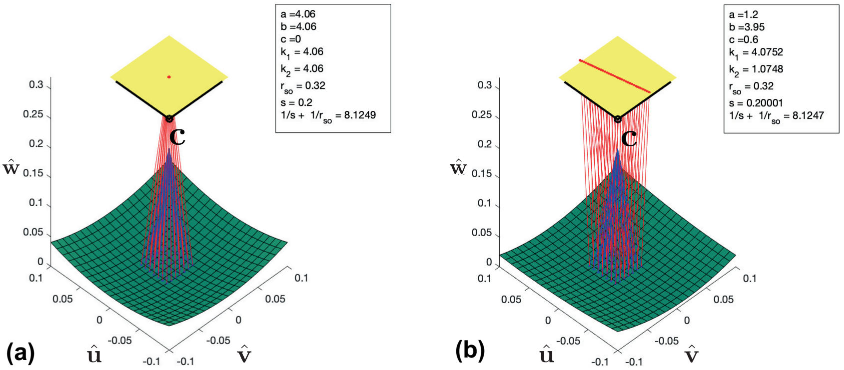


Fig. 3.8: A simulated coaxial PMD. The camera rays all converging to the camera center c given by the center of the entrance pupil of the camera optic are displayed in blue and the incident rays from the screen is displayed in red. The object surface is shown in green and the screen plane is shown in yellow. The geometrical parameters are given in [m]. (a): all camera rays converge to a screen point; (b): all camera rays converge to a screen line. Both situations are not problematic for the the phase shift evaluation, since the sequential information is not affected by the lack of lateral context.

However, most of the conventional PMDs are not coaxial. In a non-coaxial system, even a perfectly rotationally symmetric object surface will result in astigmatism, since the effective focal length differs in the meridional and the sagittal planes. Denoting the angle of incidence as θ and letting the non-coaxial incident and reflection ray lie on the $\hat{\mathbf{u}}\hat{\mathbf{w}}$-plane as shown in Fig. 3.9, the $\hat{\mathbf{u}}\hat{\mathbf{w}}$-plane becomes the meridional plane for the "off axis" imaging process and the $\hat{\mathbf{v}}\hat{\mathbf{w}}$-plane becomes the sagittal plane for this "on axis" imaging process. Projecting all the camera rays onto the sagittal plane, the lens equation can be written as

$$\frac{1}{s\cos\theta} + \frac{1}{r_{so}\cos\theta} = \frac{1}{f} = 2b. \tag{3.11}$$

Rewriting (3.11) as $\frac{1}{s} + \frac{1}{r_{s0}} = \frac{1}{\frac{f}{\cos\theta}} = 2b\cos\theta$, the effective focal length $f_s = \frac{f}{\cos\theta}$ and the effective curvature $b_s = b\cos\theta$ on the sagittal plane are obtained. The effective focal length on the sagittal plane can also be found in [Gro18]. According to [Gro18], the effective focal length on the meridional plane is given by $f_m = f\cos\theta$ and the effective curvature $a_m = \frac{a}{\cos\theta}$.

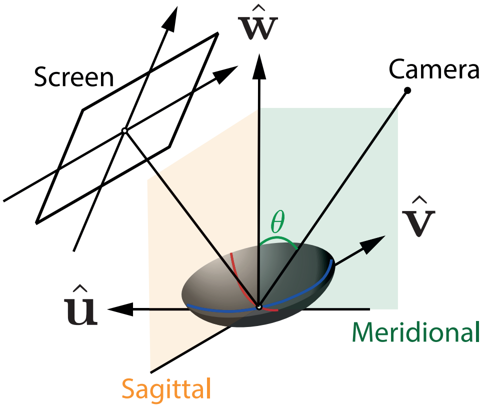


Fig. 3.9: Non-coaxial PMD resulting in an off-axis imaging of the camera center by the object surface, and the meridional and sagittal plane depending on the angle of incidence θ.

To describe this a bit more involved situation, an "effective Weingarten

Matrix"can be introduced, defined by

$$W_e = \begin{bmatrix} \frac{a}{\cos\theta} & c \\ c & b\cos\theta \end{bmatrix}, \tag{3.12}$$

and the effective principal curvature is therefore given by

$$k_{e1,e2} = \frac{\frac{a}{\cos\theta} + b\cos\theta}{2} \pm \sqrt{\frac{(\frac{a}{\cos\theta} - b\cos\theta)^2}{4} + c^2}. \tag{3.13}$$

For a general, coaxial or non-coaxial PMD, if the geometric relation of the image distance, object distance and effective local curvature satisfies the lens equation:

$$\frac{1}{s} + \frac{1}{r_{s0}} = \frac{1}{f_e} = 2k_{e1,e2} \tag{3.14}$$

the camera rays converge to a focus line on the screen. Such a geometry is shown in a simulation in Fig. 3.10. This condition will be found in Chapter 5.3.1 by reintroducing the [SCP05] completely mathematically without the corresponding physical interpretation.

Note that for a diverging convex mirror ($a < 0, b < 0, c^2 < ab$), the camera rays never converge to a focal point or line and (3.14) is never satisfied[4].

3.2.3 Phase Monotony

In most PMD camera images, the phase value is monotonically increasing or decreasing. This happens when the object surface is smooth and has no strong local curvature. This monotony, however, will be broken if there is a focus line and the screen is above the focus line. The monotony is completely irrelevant for the conventional phase shifting evaluation, since the correspondence between screen and camera is evaluated independently of the lateral context by the sequential phase shift. For single-shot evaluation, however, monotony is an

---

[4] A convex mirror follows $a < 0, b < 0, c^2 < ab$. Since $\frac{(\frac{a}{\cos\theta} - b\cos\theta)^2}{4} + c^2 = \frac{\frac{a^2}{\cos^2\theta} + b^2\cos^2\theta - 2ab + 4c^2}{4} < \frac{\frac{a^2}{\cos^2\theta} + b^2\cos^2\theta + 2ab}{4} = \frac{(\frac{a}{\cos\theta} + b\cos\theta)^2}{4}$, $k_{e1,e2}$ are therefore negative and (3.14) is never satisfied.

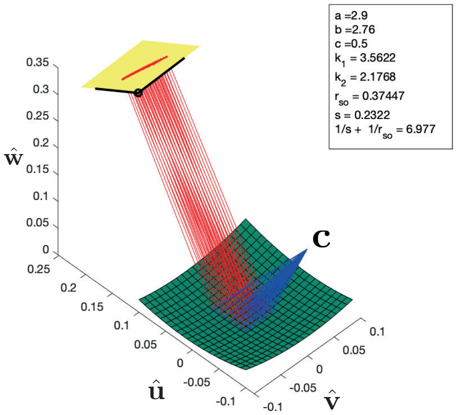


Fig. 3.10: A simulated Non-coaxial PMD while the lens equation 3.14 is satisfied. The camera rays are displayed in blue and the incident rays from the screen are displayed in red. The geometrical parameters are given in [m]

essential requirement for evaluability, since no continuous context information is available at the place where the monotony is broken and additional information about the increasing screen pixel position in camera image is required.

Consider a PMD setup with a fixed camera and object, and let the screen distance r_{s0} be flexible. Denote the screen distance as r_{s0f}[5] if r_{s0} satisfies (3.14) and let r_{s0f1} and r_{s0f2} be the two solutions[6] of (3.14). If for the whole object surface r_{s0f} does not exist (e.g. for a convex mirror), or if r_{s0f} exists but the screen distance r_{s0} is always smaller than $\min(r_{s0f1}, r_{s0f2})$, the camera image of the screen is never mirrored, and monotony holds for the whole area. An example is shown in the simulation in Fig. 3.11 on the left hand side. Single-shot

---

[5] Note that r_{s0f} denotes the distance between *the local object surface patch* to *the corresponding screen point*. For a (locally) convex mirror, as discussed in Chapter 3.2.2, r_{s0f} does not exist. If r_{s0f} exists, r_{s0f} is not a constant for the entire object surface, but varies depending on the *local curvature* and *the angle of incidence*.

[6] Analogous to astigmatism, the camera rays will converge to two focal lines at two different screen distances and the two focal lines will lie in different directions.

PMD is in this case performable.

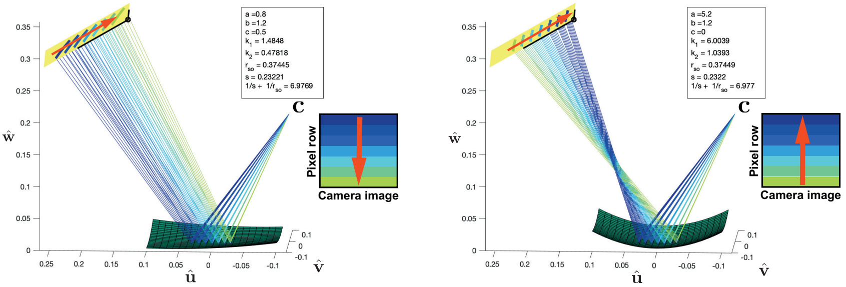


Fig. 3.11: A simulated Non-coaxial PMD. Left: the light rays in a common PMD; Right: the light rays while the screen above the focal line. The geometrical parameters are given in [m]

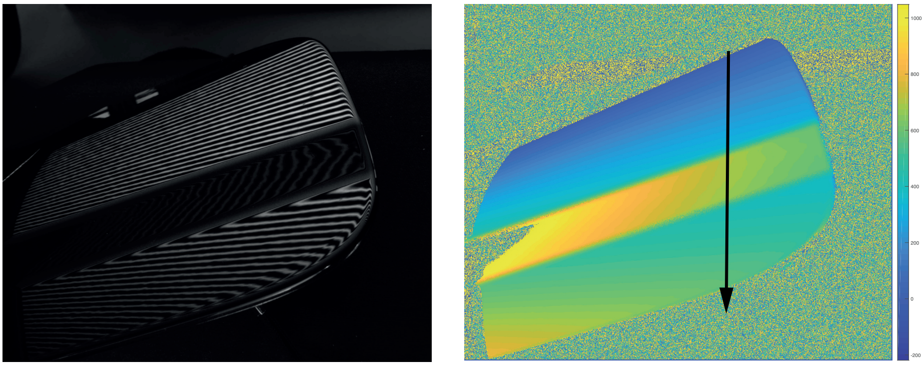
If on the other hand r_{s0f1} and r_{s0f2} both exist for the entire object surface and the screen distance r_{s0} is always between $\min(r_{s0f1}, r_{s0f2})$ and $\max(r_{s0f1}, r_{s0f2})$, the camera image of the screen is *always* mirrored, as shown in Fig. 3.11 on the right hand side. The monotony also holds for the whole area and only the image is mirrored with respect to the focal line[7]. In this case single-shot PMD is also performable, but prior knowledge about the object surface, and the direction in which the fringe phases and therefore the encoded screen coordinates increase is required since the image is mirrored. If r_{s0f1} and r_{s0f2} both exist for the entire object surface and the screen distance r_{s0} is always greater than $\max(r_{s0f1}, r_{s0f2})$, the image is mirrored with respect to *both* of focal lines. The monotony also holds and prior knowledge about the system geometry is needed for the single-shot evaluation, but not for the phase shift evaluation.

If r_{s0f1} and r_{s0f2} are present (not necessarily for the entire surface) and the screen distance r_{s0} is partly greater than $\min(r_{s0f1}, r_{s0f2})$, the monotony is broken since the camera image of the screen is partly mirrored. An example is displayed in Fig. 3.12. The fringe pattern reflected on a petrol cap is recorded by a camera and the camera image is displayed on the left hand side in Fig. 3.12.

---

[7] The focal line given by the screen distance $\min(r_{s0f1}, r_{s0f2})$

The phase of the fringe is evaluated by the phase shift method and shown on the right hand side. Along the black arrow shown in Fig. 3.12, the phase value at first increases, and then decreases, and at the end increases again. In single-shot evaluation, the increasing phase direction may be determined by prior knowledge about the system geometry or by optical flow[HS81][LK81][BWS05]. However the transition area from the non-mirrored screen image to the mirrored screen image, or vice versa, is not suitable for single-shot evaluation because the lateral context is lost at the border.



Camera image

Screen pixel position

Fig. 3.12: Left: a PMD camera image; Right: the PMD phase value.

c

Chapter 4

Time-Frequency Analysis for Phase Evaluation

When using PMD to measure the topography of a specular surface, the geometrical correspondence of a camera pixel point and the observed screen point must be found. This correspondence is encoded in the phase of the sinusoidal fringe pattern, which is conventionally obtained by applying the phase shifting method discussed in Chapter 2.1.2. However, because of its sequential measuring process, the phase shift method is not suitable for moving objects in industrial production, nor for non-fixable objects such as the human cornea. Therefore, single-shot PMD [HNA11][Liu+14], as discussed in Chapter 2.2.1, utilizing the additive superposition of two perpendicular fringe patterns and single-side band demodulation instead of a temporal phase shift in order to determine the phase [TIK82] has been invented. However, single-sideband demodulation only works for camera images presenting narrow bandwidths. The camera image of objects with complex surface geometries (Fig. 2.12) that result in broadband fringe patterns (Fig. 2.13) cannot be evaluated this way, since the single-side band demodulation approach [TIK82] only considers both frequency and phase globally in Fourier space and the global frequencies determined this way cannot be filtered appropriately in the Fourier domain.

To determine the local phases of broadband fringe patterns like this, time-frequency analysing methods muss be applied, in which the signal is evaluated locally both in time and in frequency domain. In this chapter, different time-frequency methods will be discussed and a specially adapted time-frequency method, called "*localized ridgelet transform*", will be introduced for the phase evaluation of the additive broadband fringe patterns occurring when using single-shot deflectometry on complex surfaces.

Although the task at hand deals with spatially rather than temporally varying patterns and functions, common terms like "time-frequency analysis" or "time-frequency domain" etc. will be used throughout this chapter, although the term "frequency" always refers to a spatial frequency and the "time domain" is in fact always a physically spatial domain.

4.1 State of the Art Technology of Time-Frequency Analysis

Applying the same notation as introduced in Chapter 2.2.1, the regular cross fringe pattern (Fig. 2.7) encoding the 2D screen position applied in the SSPMD can be represented in the following normalized form:

$$I_s(x_s, y_s) = \frac{1}{2} + \frac{\cos\phi_x(x_s) + \cos\phi_y(y_s)}{4} \tag{4.1}$$

with (x_s, y_s) denoting the two dimensional coordinates of the plane defined by the screen, and $\phi_x(x_s) = 2\pi f_{sx} x_s$ and $\phi_y(y_s) = 2\pi f_{sy} y_s$ being the phases (the phases depend linearly on x_s and y_s respectively and f_{sx}, f_{sy} are the frequencies) encoding the screen position (x_s, y_s). The cross fringe pattern is distorted according to the geometry of the object surface, and the reflection on the test object surface is captured by a camera. The intensity of the camera image can be represented by[1]:

$$I_c(x_c, y_c) = I_{c0}(x_c, y_c) + M_c(x_c, y_c)(\cos\phi_x(x_c, y_c) + \cos\phi_y(x_c, y_c)), \tag{4.2}$$

where (x_c, y_c) denotes the two dimensional coordinates of the camera image plane, and $\phi_x(x_c, y_c)$ and $\phi_y(x_c, y_c)$ indicate the desired phase information observed by the camera pixel (x_c, y_c), by which the corresponding (observed) point on the screen is determined. Note that the x- or y-instantaneous frequency times 2π equals the gradient of the instantaneous phase $\phi_x(x_c, y_c)$ or $\phi_y(x_c, y_c)$ respectively. The instantaneous phase is, strictly taken, defined by the complex argument of an analytic signal. If the signal I_c is real, the instantaneous phase is determined by the complex argument of the signal's analytic representation $I_c + iH(I_c)$, where $H(I_c)$ represents the Hilbert transform of the signal I_c. If strong structures are present on the specular object surface, the instantaneous frequency varies considerably This chapter discusses how to determine the highly varying instantaneous frequencies, resp. their phase $\phi_x(x_c, y_c)$ and $\phi_y(x_c, y_c)$,

---

[1] This model implicitly assumes that the modulations of the additive fringe structures are the same in both directions

with the time-frequency analysing methods.

The global notation for the camera coordinates and screen coordinates is (x_c, y_c) and (x_s, y_s), respectively. Since we are only interested in the reflected pattern on the image plane captured by the camera in this chapter, (x, y), instead of (x_c, y_c), will be used for camera coordinates in this chapter.

4.1.1 Short-time Fourier Transform

To analyse a signal locally, the signal must be transformed to a suitable time-frequency domain, which represents both the temporal (resp. spatial) and the spectral properties of the signal simultaneously. The *Short-time Fourier Transform* (STFT), also known as *Windowed Fourier Transform* (WFT), is historically the first and simplest approach extending the Fourier transform to a time-frequency analysis. Instead of analysing the signal frequency globally, STFT divides the entire signal into shorter segments of equal length and then computes the Fourier transform on each shorter segment independently. A 2D STFT can be written as[SEV16][Mal08][Add02]:

$$\begin{aligned} X\{I\}(b_x, b_y, f_x, f_y) = \iint & I(x, y) e^{-i2\pi f_x x} e^{-i2\pi f_y y} \\ & w(x - b_x, y - b_y) dx dy, \end{aligned} \tag{4.3}$$

where $w(x, y)$ is a normalized window function ($\parallel w \parallel_{\mathcal{L}^2} = 1$) centered at $(0, 0)$ and $w(x - b_x, y - b_y)$ represents the same window centered at (b_x, b_y). I is the signal function (the intensity of the camera image) and b_x and b_y are the translation factors in x- and y-direction. The "time-frequency atom" of STFT is therefore:

$$\psi_{b_x, b_y, f_x, f_y} = e^{-i2\pi f_x x} e^{-i2\pi f_y y} w(x - b_x, y - b_y). \tag{4.4}$$

The window function of STFT can in principle be chosen arbitrarily. [Gab46] introduced the Gaussian window function into the STFT, called Gabor transform (GBT), which advances the resolution of localization in both the time and frequency domains. GBT is therefore a special and more advanced type of STFT, avoiding the common spectral leakage in the form of multiple side lobes associated with simple rectangular localization windows. However, there is still a fundamental trade-off between the temporal and the spectral resolution.

The time location μ_x, μ_y resp. the frequency location μ_{f_x}, μ_{f_y}, the time spread σ_x^2, σ_y^2 resp. the frequency spread $\sigma_{f_x}^2$, $\sigma_{f_y}^2$ of a time-frequency atom ψ_{b_x,b_y,f_x,f_y} are defined as [Mal08][Add02]:

$$\begin{aligned}
\mu_x &= \int x \parallel \psi_{b_x,b_y,f_x,f_y} \parallel^2 dx,\\
\mu_y &= \int y \parallel \psi_{b_x,b_y,f_x,f_y} \parallel^2 dy,\\
\mu_{f_x} &= \int f_x \parallel \hat{\psi}_{b_x,b_y,f_x,f_y} \parallel^2 df_x,\\
\mu_{f_y} &= \int f_y \parallel \hat{\psi}_{b_x,b_y,f_x,f_y} \parallel^2 df_y,\\
\sigma_x^2 &= \int (x-\mu_x)^2 \parallel \psi_{b_x,b_y,f_x,f_y} \parallel^2 dx,\\
\sigma_y^2 &= \int (y-\mu_y)^2 \parallel \psi_{b_x,b_y,f_x,f_y} \parallel^2 dy,\\
\sigma_{f_x}^2 &= \int (f_x-\mu_{f_x})^2 \parallel \hat{\psi}_{b_x,b_y,f_x,f_y} \parallel^2 df_x,\\
\sigma_{f_y}^2 &= \int (f_y-\mu_{f_y})^2 \parallel \hat{\psi}_{b_x,b_y,f_x,f_y} \parallel^2 df_y,
\end{aligned} \tag{4.5}$$

with $\hat{\psi}$ being the Fourier transform of ψ.

The fundamental trade-off between the temporal and the spectral resolution of all linear transforms used for spectral analysis is called the time-frequency uncertainty, also called the Heisenberg uncertainty principle [2], which is mathematically presented as $\sigma_x \sigma_{f_x} \geq \frac{1}{4\pi}$ in 1D cases, and $\sigma_x \sigma_y \sigma_{f_x} \sigma_{f_y} \geq \frac{1}{16\pi^2}$ in 2D cases. The Heisenberg uncertainty can be visualized by a so-called Heisenberg

---

[2] Heisenberg combined this fundamental trade-off with the (physical) relations between energy and (temporal) frequency (Planck) resp. momentum and spatial frequency (de Broglie) in order to define his famous physical "uncertainty relation".

box in the time-frequency space [3] centered at $(\mu_x, \mu_y, \mu_{f_x}, \mu_{f_y})$ and having a time width equal to σ_x and σ_y, and a frequency width equal to σ_{f_x} and σ_{f_y} as shown in Fig. 4.1.

Since the window of STFT is equidistantly chosen and not correlated with the frequency, the shape of the $4d$ Heisenberg box $\sigma_x\sigma_y\sigma_{f_y}\sigma_{f_y}$ is independent of its position $(\mu_x, \mu_y, \mu_{f_y}, \mu_{f_y})$. The Heisenberg box of STFT atoms is schematically shown in Fig. 4.1. For simplification, the Heisenberg box of a 1D signal is displayed in a time-frequency plane.

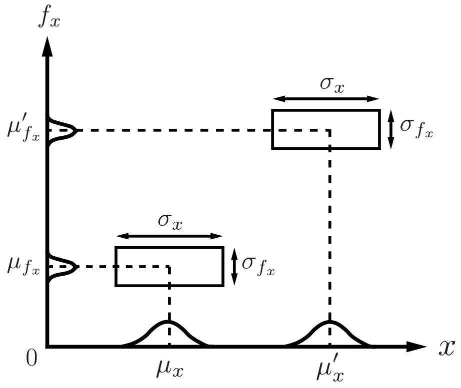


Fig. 4.1: The Heisenberg box representing the time-frequency spread of STFT atoms.

The Heisenberg uncertainty principle [Mal08] states that, the product of the uncertainty by determining the time and the uncertainty by determining the frequency is larger than a constant. This limits the time-frequency resolution of the atoms. In all the linear time-frequency analysing methods, there is always a trade-off between time and frequency resolution. The higher the signal frequency, the better the time resolution and therefore the worse the frequency resolution has to be, and vice versa (Fig. 4.2). The Heisenberg box of STFT, however,

---

[3] In general cases, this term is known as the time-frequency plane. Here we discuss about a 2D signal, whose transform into the time-frequency domain is 4 dimensional. Therefore, it is named as time-frequency space.

always has the same shape. STFT therefore has a good absolute resolution but a poor relative resolution in the time-frequency domain. Using STFT to analysing a signal with significantly varying frequency can therefore lead to serious errors.

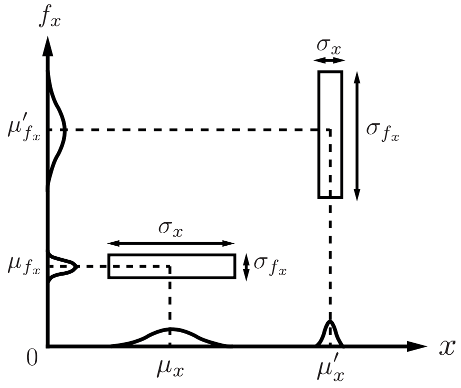


Fig. 4.2: The schematic representation of the Heisenberg uncertainty.

An example is shown in Fig. 4.3 and the scalogram - the visualization of the spectrum of a signal over time - of the STFT of the signal shown in Fig. 4.3 is displayed in Fig. 4.4. An 1D sine wave signal consisting of three frequencies is applied subsequently at different times. The amplitude of the signal and the time-frequency distribution are displayed in Fig. 4.3. The signal is transformed into the time-frequency domain by using STFT of two different window sizes and the resulting time-frequency spectra are shown in Fig. 4.4. Different from a conventional FT, the existing frequencies and their amplitudes and phases are not presented globally but extended in the time axis. The frequency can be fairly well defined (resp. resolved) by using a large window in time (on the left hand side of Fig. 4.3), but - as limited by the Heisenberg uncertainty - then the location is “smeared”. By using a small window as shown on the right hand side of Fig. 4.3, the location is well defined, but the frequency resolution is poor. Given a window width, the shape of the Heisenberg box of STFT remains the same independently from the frequency of the signal. When dealing with a broadband signal, the chosen window length can only be suitable for a certain frequency range. A reliable full-field evaluation is therefore difficult to perform.

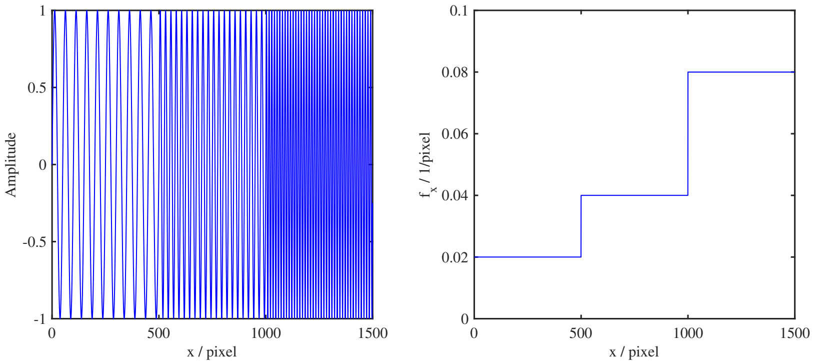


Fig. 4.3: Left: a 1D sine wave signal in time domain; Right: the 1D signal presented in time-frequency domain.

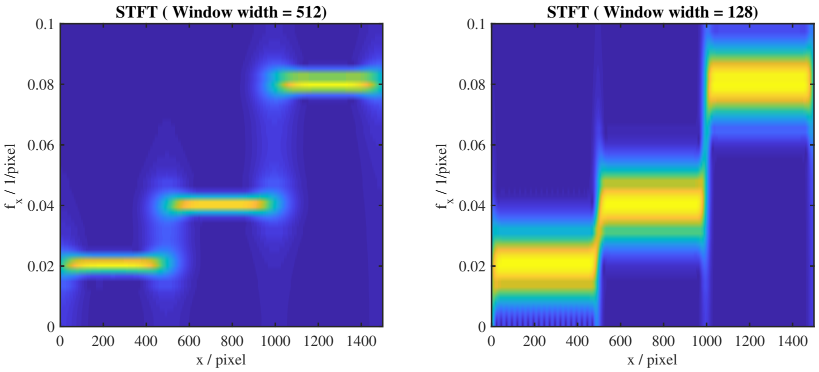


Fig. 4.4: The scalogram of the STFT of the signal shown in Fig 4.3. Left: the window size of the time-frequency atom is 512 pixels in the time domain; Right: the window size of the time-frequency atom is 128 pixels in the time domain.

4.1.2 Wigner-Ville Transform

Besides the linear time-frequency approach like STFT, [Vil] introduced a quadratic method, called *Wigner-Ville Transform* (WVT) or *Wigner-Ville Distribution* (WVD), to analyse the time and frequency distribution of a signal that had been studied by Wigner [Wig32] in quantum thermodynamics. WVT is defined as the Fourier transform of the instantaneous autocorrelation function (ACF) of a signal (representation in 1D for simplification):

$$WV\{I\}(b_x, f_x) = \int I(b_x + \frac{x}{2})I^*(b_x - \frac{x}{2})e^{-i2\pi f_x x}dx, \tag{4.6}$$

where the asterisk denotes the complex conjugate and x is a dummy integration variable in time. Applying the Parseval's formula, the Wigner-Ville transform above can also be equivalently represented as a inverse Fourier transform of the autocorrelation function in frequency:

$$WV\{I\}(b_x, f_x) = \int \hat{I}(f_x + \frac{f}{2})\hat{I}^*(f_x - \frac{f}{2})e^{i2\pi f b_x}df, \tag{4.7}$$

where f is a dummy integration variable in frequency. The WVT can be regarded as using the signal itself as a filter and provides a high-resolution representation in both time and frequency domain, since time and frequency have a symmetric role in WVT.

Both the instantaneous power (in time) and the energy spectrum (in frequency) can be recovered from the WVT simply as marginal distributions: since the integral of the Wigner-Ville distribution over the frequency (the time) at any b_x (f_x) is equal to the energy density of a signal in time (frequency)[Lok15]:

$$\int_{-\infty}^{\infty} WV\{I\}(b_x, f_x)df_x =\mid I(b_x) \mid^2, \tag{4.8}$$

or

$$\int_{-\infty}^{\infty} WV\{I\}(b_x, f_x)db_x =\mid \hat{I}(f_x) \mid^2 . \tag{4.9}$$

(4.8) can be proven by applying the inverse Fourier transform of (4.6):

$$\begin{aligned}\int WV\{I\}(b_x,f_x)e^{i2\pi f_x\tau_x}df_x &= \iint I(b_x+\frac{x}{2})I^*(b_x-\frac{x}{2})e^{-i2\pi f_x x}e^{i2\pi f_x\tau_x}df_x dx\\ &= \int I(b_x+\frac{x}{2})I^*(b_x-\frac{x}{2})\int e^{-i2\pi f_x(x-\tau_x)}df_x dx\\ &= \int I(b_x+\frac{x}{2})I^*(b_x-\frac{x}{2})\delta(x-\tau_x)dx\\ &= I(b_x+\frac{\tau_x}{2})I^*(b_x-\frac{\tau_x}{2}).\end{aligned} \tag{4.10}$$

Letting $\tau_x = 0$, $\int WV\{I\}(b_x, f_x)df_x = I(b_x)I^*(b_x) =\mid I(b_x) \mid^2$ is obtained [Lok15]. Analogously, (4.9) is obtained by performing a Fourier transform of (4.7).

WVD provides the highest possible time-frequency resolution that is mathematically possible within the limits of the uncertainty principle. As discussed in [FS97][BLC12], the time-frequency resolution of the WVD is always $\geq \frac{\|I\|_2^2}{2\pi}$, which depends on the input signal itself. The WVD of the signal shown in Fig. 4.3 is displayed in Fig. 4.5. The signal is solved sharply.

One drawback of WVD is the interference created by the quadratic property of the transform when applied to a composite signal. Let $I = I_1 + I_2$ be a composite signal. The WVT of I can be written as:

$$WV\{I\} = WV\{I_1\} + WV\{I_2\} + WV\{I_1, I_2\} + WV\{I_2, I_1\}, \tag{4.11}$$

with the cross terms:

$$\begin{aligned}WV\{I_1,I_2\} &:= \int I_1(b_x+\frac{x}{2})I_2^*(b_x-\frac{x}{2})e^{-i2\pi f_x x}dx,\\ WV\{I_2,I_1\} &:= \int I_2(b_x+\frac{x}{2})I_1^*(b_x-\frac{x}{2})e^{-i2\pi f_x x}dx.\end{aligned} \tag{4.12}$$

These cross terms appear when the cross-correlation of the two signals I_1 and I_2

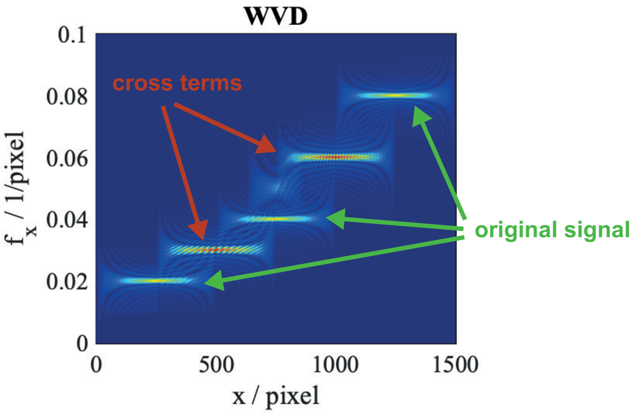


Fig. 4.5: The WVD of the signal shown in Fig. 4.3.

is nonzero, with the energy density of the cross terms lying between the energy densities of I_1 and I_2 in both time and frequency [Mal08]. Due to the additive superposition of two fringe patterns used in SSPMD and the possible shearing of the pattern in the camera image, the signal of an SSPMD measurement is always a multicomponent signal. Therefore, WVD is not suitable for SSPMD evaluation. In addition, camera images have only positive intensity. All camera image signals are the superposition of a modulated signal and a DC component. Hence, camera images themselves are always multicomponent signals. To apply WVD to a camera image, the DC term must be removed (eg. by using empirical mode decomposition [Zei+10]) in advance. The cross term artifacts are also shown in Fig. 4.5 between two signal sequences.

Furthermore, the Wigner-Ville distribution does not contain any phase information. The WVD represented in (4.6) remains real, because it is the Fourier transform of $I(b_x + \frac{x}{2})I^*(b_x - \frac{x}{2})$, which has a Hermitian symmetry in x. The phase of the signal encoding the screen position is not given by the transform. This phase could be recovered by numerical integration after the instantaneous frequency is known. For this, a further evaluation is necessary.

4.1.3 Continuous Wavelet Transform

An advanced linear time-frequency transform not suffering from the unwanted "cross terms" introduced above is the *Continuous Wavelet Transform* (CWT). The fundamental idea of wavelet transforms is to "spread" (stretch resp. squeeze) a basic function called "wavelet" with a varying dilation ("scale") but always with the same shape. The longer the dilation, the lower the frequency while - at the same time - the larger the (temporal resp. spatial) width of the localization window. This way, the basic function adjust itself to the time-frequency resolution. The basic function is called wavelets or time-frequency atoms and defined by

$$\psi_{a_x,a_y,b_x,b_y}(x,y) = \frac{1}{\sqrt{a_x a_y}}\psi_x(\frac{x-b_x}{a_x})\psi_y(\frac{y-b_y}{a_y}), \tag{4.13}$$

where a_x, a_y are the dilation parameters (the scales) and b_x and b_y are the translation parameters. $\psi_{a_x,a_y,b_x,b_y}(x,y)$ is called the mother wavelet when $a_x = 1$, $a_y = 1$, $b_x = 0$ and $b_y = 0$. A signal is transformed into the time-frequency domain by using a range of a_x's, a_y's, b_x's and b_y's. The continuous wavelet transform of the signal $I_c(x_c, y_c)$ is defined as

$$W_\psi\{I\}(a_x,a_y,b_x,b_y) = \frac{1}{\sqrt{a_x a_y}}\iint_{\mathbb{R}^2} I(x,y)\psi_x^*(\frac{x-b_x}{a_x})\psi_y^*(\frac{y-b_y}{a_y})dxdy. \tag{4.14}$$

The wavelet transform can be thought of as the cross-correlation of a signal with a set of wavelets of various 'widths'. In contrast to the STFT, the window width and the frequency of the wavelet atoms change jointly according to the dilation parameters a_x and a_y. In the time-frequency plane, the width and the height of the Heisenberg box of a wavelet atom compromises between the time and frequency resolution. The varying Heisenberg boxes for the 1D case are schematically shown in Fig. 4.6. The area $\sigma_{ox}\sigma_{of_x}$ remains constant due to the uncertainty relation. The width and the height of the rectangle are linearly related to a_x and $\frac{1}{a_x}$, which is adjusted to the signal automatically.

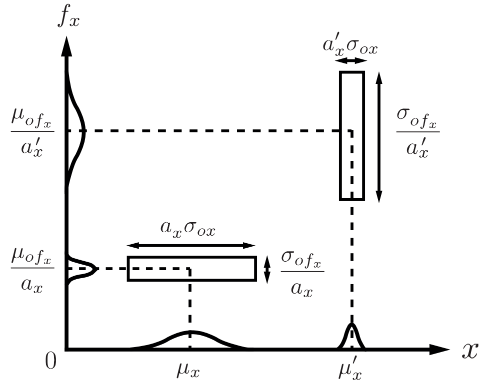


Fig. 4.6: The Heisenberg box representing the time-frequency spread of CWT atoms. μ_{of_x} is the center frequency of the mother wavelet at its origin ($a_x = 1$). σ_{ox} and σ_{of_x} are the time- and frequency-uncertainty of the mother wavelet at its origin. The mathematical expressions of μ_{of_x}, σ_{ox} and σ_{of_x} are analogous to (4.5).

The effectiveness of the CWT in locating the time and frequency of an 1D signal is shown in Fig. 4.7. The signal displayed in Fig. 4.3 is transformed into the time-frequency domain now by employing the CWT with the 1D time-frequency atom of a Morlet wavelet: $\psi_x(x) = \frac{1}{\sqrt{\pi F_b a_x}} e^{i2\pi F_c(\frac{x-b_x}{a_x})} e^{-\frac{1}{F_b}(\frac{x-b_x}{a_x})^2}$. Note that the admissibility condition of this wavelet is not satisfied, which will be discussed in Chapter 4.3.1. Fig. 4.7 demonstrates that the wavelet adjusts its time and frequency resolution to the signal itself and does not need a local parameter definition.

Coming back to the 2D pattern image, the input signal is the superposition of two distorted intrinsically 1D-fringe patterns. The phase of the two fringe patterns must be evaluated separately. One possible evaluation procedure could

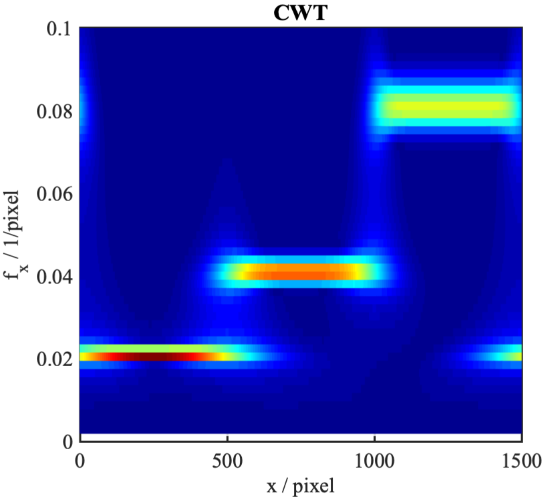


Fig. 4.7: The scalogram of CWT of the signal displayed in Fig 4.3.

be to separate the 2D wavelet transform ((4.14)) into

$$\begin{aligned} W_{\psi}^{(x)}\{I\}(a_x, b_x, b_y) &= \tfrac{1}{\sqrt{a_x}} \iint_{\mathbb{R}} I(x,y)\psi_x^*(\tfrac{x-b_x}{a_x})w_y(y-b_y)dxdy, \\ W_{\psi}^{(y)}\{I\}(a_y, b_x, b_y) &= \tfrac{1}{\sqrt{a_y}} \iint_{\mathbb{R}} I(x,y)\psi_y^*(\tfrac{y-b_y}{a_y})w_x(x-b_x)dxdy, \end{aligned} \tag{4.15}$$

where $w_y(y-b_y)$ and $w_x(x-b_x)$ are two compactly supported bump functions in $y-$ and $x-$ directions, respectively.

However, this evaluation is not sensitive to any fringe direction variation. This procedure allows only for the evaluation of reflected 2D cross patterns with separate phase variations only along the x- and y-directions and no direction variations at all. Therefore, as the directions of the fringes are also expected to vary simply by perspective projection effects, this method is not applicable for the given task.

4.2 Localized 2D Ridgelet Transform applied to Phase Evaluation in PMD

The advantage of utilizing CWT to analyse a broadband signal is discussed in Chapter 4.1.3. However, the conventional 2D CWT is not sensitive to any direction change. To analyse the phase of the PMD pattern as shown in Fig. 2.12, the CWT muss be further adjusted.

4.2.1 2D Ridgelet Transform

Different from the 2D Wavelet transform, the so-called *Ridgelet Transform* (RT) is sensitive to all fringe directions with its mother wavelet given by [Can98]:

$$\psi_{a,b,\theta}(x,y) = \frac{1}{\sqrt{a}}\psi(\frac{x\cos\theta + y\sin\theta - b}{a}), \tag{4.16}$$

where θ is the directional parameter with $\theta \in [0, 2\pi]$. A ridgelet is constant along the line $x\cos\theta + y\sin\theta = const.$ as shown in Fig. 4.8. This direction is denoted as the projection direction or "p-direction". Perpendicular to the "ridge", the ridgelet is an 1D wavelet. As demonstrated in Fig. 4.8, this direction is denoted as the transform direction or "t-direction". The two-dimensional continuous ridgelet transform of the signal $I(x,y)$ is defined as follows:

$$R_\psi\{I\}(a,b,\theta) = \iint_{\mathbb{R}^2} I(x,y)\psi^*_{a,b,\theta}(x,y)dxdy. \tag{4.17}$$

Applying the directional parameter θ, fringe patterns which are not strictly aligned along the x or y direction can now be detected. Since a ridgelet is constant along lines $x\cos\theta + y\sin\theta = const.$, the ridgelet transform in (4.17) can also be implemented as an one dimensional Wavelet transform applied to the signal projected to a line perpendicular to $x\cos\theta + y\sin\theta = const.$:

$$R_\psi\{I\}(a,b,\theta) = \int_{\mathbb{R}} \psi^*_{a,b}(t)Rd\{I\}(\theta,t)dt, \tag{4.18}$$

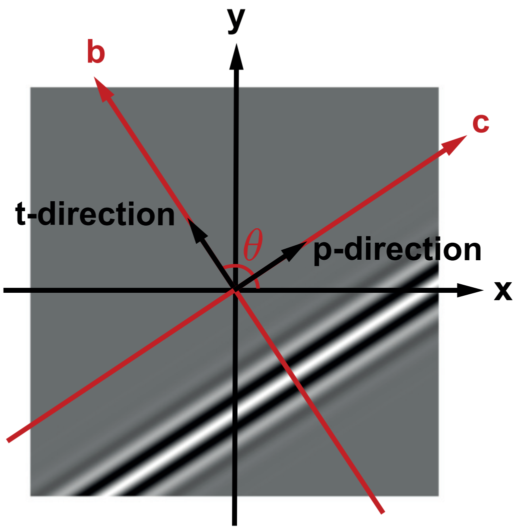


Fig. 4.8: A ridgelet is constant along the direction defined by θ (projection direction or p-direction) and the transverse of the ridge is an one-dimensional wavelet shifted by b (transform direction or t-direction).

where t is a dummy variable, $\psi_{a,b}(t)$ is an one-dimensional wavelet with:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a}), \tag{4.19}$$

and $Rd\{I\}(\theta, t)$ is the projection of the 2D signal to a line, called a Radon transform [4][Dea07]:

$$Rd\{I\}(\theta, t) = \iint_{\mathbb{R}^2} I(x, y)\delta(x\cos\theta + y\sin\theta - t)dxdy. \tag{4.20}$$

In contrast to the 2D Wavelet transform described in the Chapter 4.1.3, the

---

[4] Note that, if instead of an 1D Wavelet transform, an 1D Fourier transform is applied to $Rd\{I\}(\theta, t)$ along t, (4.18) is the 2D Fourier transform of $I(x, y)$ - the projection-slice theorem. The difference between a 2D Fourier transform and a ridgelet transform is whether the time-frequency atom is finite or infinite.

ridgelet transform is sensitive to all fringe directions by rotating the mother wavelet. But at the same time, the local information along the projection direction (p-direction) is lost, since the 2D signal is integrated in this direction and the signal is only time-frequency transformed in the perpendicular direction (t-direction).

4.2.2 Localized 2D Ridgelet Transform

The lost local information along the p-direction in the standard 2D ridgelet transform can be recovered by applying an additional “bump function” w in the p-direction (Fig. 4.8):

$$w_{c,\theta}(x,y) = w(x\sin\theta - y\cos\theta - c). \tag{4.21}$$

Note that $w_{c,\theta}(x,y)$ is constant along the line $x\sin\theta - y\cos\theta = const.$, which is perpendicular to the ridges (p-direction), and serves as a localizing test function in the projection direction (Fig. 4.9). The bump function should be smooth - continuous derivatives of all orders - and compactly supported. The product of the bump function in p-direction and the 2D ridgelet result in a localized 2D ridgelet [LSF20]:

$$\psi_{a,b,c,\theta}(x,y) = \frac{1}{\sqrt{a}}\psi(\frac{x\cos\theta + y\sin\theta - b}{a})w(\frac{x\sin\theta - y\cos\theta - c}{a}). \tag{4.22}$$

The localized two-dimensional continuous ridgelet transform of the signal $I(x,y)$ is defined as:

$$R_{\psi l}\{I\}(a,b,c,\theta) = \iint_{\mathbb{R}^2} I(x,y)\psi^*_{a,b,c,\theta}(x,y)dxdy. \tag{4.23}$$

Compared to the 2D Wavelet transform in (4.14), the translation parameters b_x and b_y are now replaced by b and c - the translation parameters along the t- and p-directions. By introducing the directional parameter θ, which rotates the t- and p-directions, the phases of distorted fringe patterns that do not change

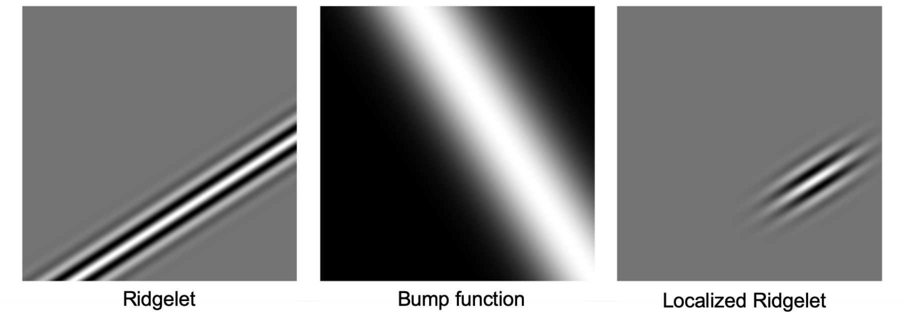


Fig. 4.9: A ridgelet is shown in the left figure. A bump function perpendicular to the ridge is shown in the figure in the middle. Applying the bump function to the ridgelet gives the localized ridgelet as shown in the right.

exactly along the x- or y-direction can now be measured. Different from the 2D Wavelet transform, where the mother wavelets stretch and squeeze in - and only in - x- and y- directions (through the dilation parameters a_x and a_y), the localized 2D ridgelet is only stretching and squeezing in the t-direction (through the dilation parameter a). Different from a standard ridgelet transform, where the 1D Wavelet transform is applied to the projected image, the local information along the projection direction is achieved by the smooth bump function. The relationship between a 2D Fourier transform, a ridgelet transform and a localized ridgelet transform is shown in Fig. 4.10 .

The mother ridgelet defined in (4.22) can also be understood as an 1D mother wavelet ψ rotated by θ and expanded in the perpendicular direction by a bump function w. The properties of various 1D wavelets are studied since decades and well-developed. The most commonly used wavelets are Haar wavelet, Morlet wavelet, Mexican hat wavelet etc.. The Morlet wavelet is suitable for the application of detecting the phase of the sinusoidal fringe patterns of PMD, since its oscillation is also sinusoidal. The localized ridgelet based on a Morlet wavelet and a bump function interpreted as a Gaussian function (which has to

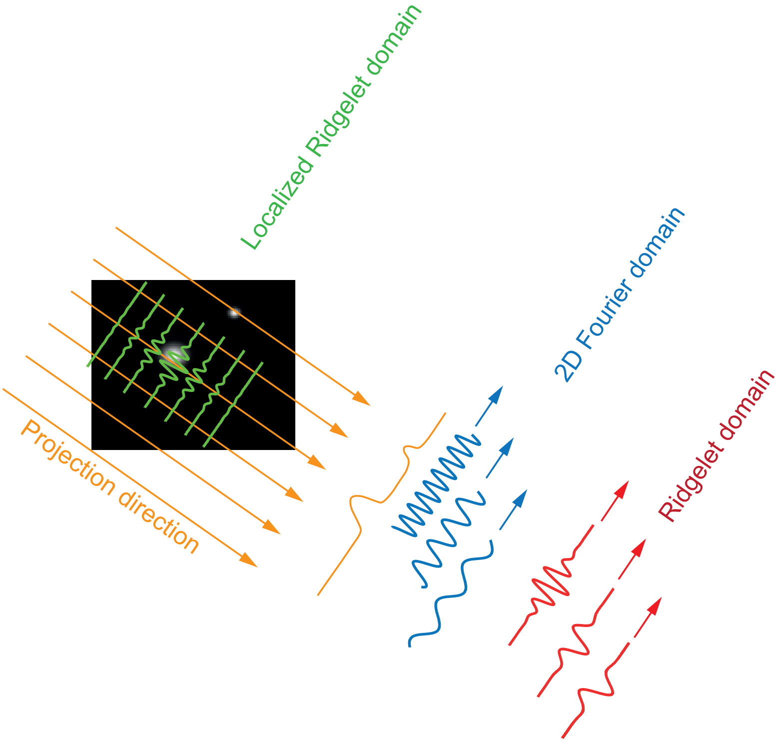


Fig. 4.10: The relationship between the 2D Fourier transform (using the Projection-slice theorem), the ridgelet transform and the localized ridgelet transform. The 1D Wavelet transform of a 2D signal integrated along the projection direction results a ridgelet transform (red). Applying an 1D Fourier transform instead of the 1D Wavelet transform, it results in a 2D Fourier transform (the Projection-slice theorem, blue). Applying a bump function instead of the integration along the direction θ leads to a localized ridgelet transform (green).

be properly clipped at the edges in order to ensure compact support[5]) can be

---

[5] As an alternative, avoiding “step artefacts” and the introduction of high local frequencies at the edges where the Gaussian has to be slightly ”clipped”(forced to zero) in order to ensure a compact support. Wendland functions providing both smoothness and compact support can

explicitly written as:

$$\begin{aligned}\psi_{a,b,c,\theta}(x,y) = \frac{1}{a^2\pi F_b} e^{i2\pi F_c(\frac{x\cos\theta + y\sin\theta - b}{a})} \\ e^{-\frac{1}{F_b}(\frac{x\cos\theta + y\sin\theta - b}{a})^2} e^{-\frac{1}{F_b}(\frac{x\sin\theta - y\cos\theta - c}{\xi a})^2},\end{aligned} \tag{4.24}$$

where F_c is the center frequency of the ridgelet, F_b is the wavelet bandwidth and a is the dilation (scale) parameter in the t-direction. b and c are the translation parameters in the t- and p-directions respectively. ξ is the proportional dilation in the p-direction with regard to the dilation a in the t-direction. The localized ridgelet is schematically shown in Fig. 4.9.

Note that, generally, an 1D wavelet is normalized with $\frac{1}{\sqrt{a}}$ (as shown in (4.16)) and a 2D wavelet with $\frac{1}{a}$, to ensure that the wavelet energy is independent of the dilation parameter a - energy conservation across scales. However, in the PMD fringe phase detection application, a different normalization factor $\frac{1}{a^2}$ (s. (4.24)) should be applied to ensure that the 2D localized "ridgelet coefficients" are independent of the scales (or $\frac{1}{a}$ in 1D cases). The difference of these two kinds of normalization factors is demonstrated in Fig. 4.11 .

Actually, the normalization factor $\frac{1}{a^2}$ for 2D cases and $\frac{1}{a}$ for 1D cases are applied in the Stockwell transform (ST) [SML96] which is defined (in 1D) as[6]:

$$ST\{I\}(f_x, b_x) = \int I(x)\frac{\mid f_x \mid}{\sqrt{2\pi}} e^{\frac{-f_x^2(b_x - x)}{2}} e^{-i2\pi f_x x} dx. \tag{4.25}$$

Except the normalization factor, ST is very similar to CWT with a Morlet wavelet kernel. The only other difference is that ST provides the absolute referenced phase information, since the phase of the ST kernel function, which is multiplied with the signal $I(x)$, is zero at $t = 0$. Therefore, the particular time-frequency atom proposed for this application and described in (4.24) is actually a mixture of CWT, RT and ST.

---

be used [Wen95].

[6] f_x is the dilation parameter expressed in frequency and a_x is the dilation parameter expressed in scale, with $f_x = \frac{const.}{a_x}$

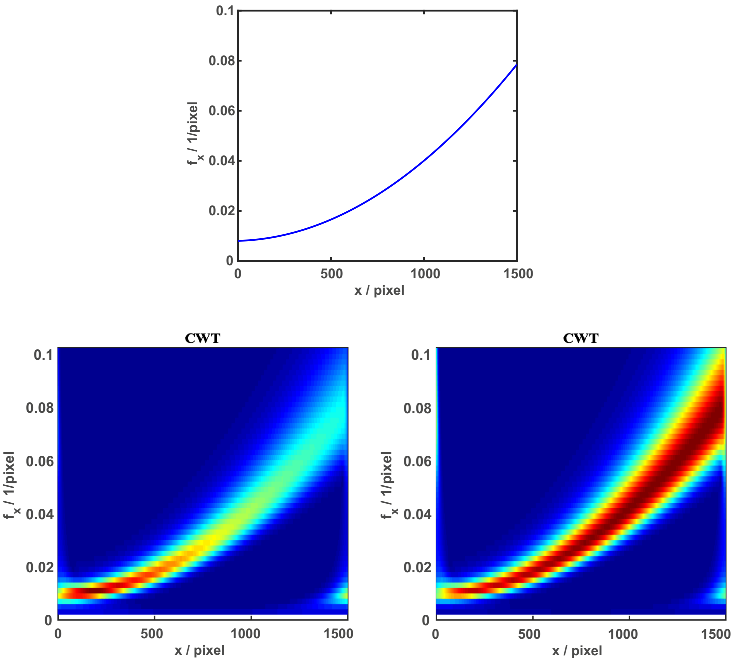


Fig. 4.11: The CWT scalogram of the signal defined by the quadratic instantaneous frequency displayed above. Bottom left: The scalogram normalized with $\frac{1}{\sqrt{a_x}}$; Bottom right: The scalogram normalized with $\frac{1}{a_x}$;

To ensure the invertibility and the energy conservation of the transform, the ridgelets must satisfy the admissibility condition [Mal08]:

$$\iint \frac{\hat{\Psi}_{a,b,c,\theta}}{\mid f_x f_y \mid} df_x df_y < \infty, \tag{4.26}$$

with $\hat{\Psi}_{a,b,c,\theta}$ denoting the 2D Fourier transform of $\psi_{a,b,c,\theta}$. To guarantee that

(4.26) is finite, $\hat{\Psi}_{a,b,c,\theta}(0,0)$ must be 0[7], or presented in time domain:

$$\widehat{\Psi_{a,b,c,\theta}}(0) = \iint \psi_{a,b,c,\theta}(x,y)dxdy = 0, \tag{4.27}$$

which is also denoted as the zero integral condition. The admissibility condition ensures the invertibility of the localized 2D ridgelet transform and ensures that the DC gain is zero. A zero DC gain is important to the application in PMD, since the camera image always has a slightly varying additive term depending on the illuminating condition, that can not be subtracted easily (s. Chapter 3.1).

Note that the ridgelet defined in (4.24) does *not* satisfy the admissibility condition since

$$\iint \psi_{a,b,c,\theta}(x,y)dxdy = \xi e^{-\pi^2 F_b F_c^2}. \tag{4.28}$$

To overcome this, a correction term muss be added to this definition:

$$\begin{aligned}\psi_{a,b,c,\theta}(x,y) = \frac{1}{a^2\pi F_b} & \underbrace{e^{-\frac{1}{F_b}(\frac{x\cos\theta+y\sin\theta-b}{a})^2}}_{\substack{window\ function\\ in\ \text{t}-direction}} \underbrace{e^{-\frac{1}{F_b}(\frac{x\sin\theta-y\cos\theta-c}{\xi a})^2}}_{\substack{bump\ function\\ in\ \text{p}-direction}} \\ & (\underbrace{e^{i2\pi F_c(\frac{x\cos\theta+y\sin\theta-b}{a})}}_{oscillation} - \underbrace{e^{-\pi^2 F_b F_c^2}}_{correction\ term}),\end{aligned} \tag{4.29}$$

or represented in the Fourier domain:

$$\hat{\Psi}_{a,b,c,\theta}(f_t,f_p) = e^{-\pi^2 F_b \xi^2 a^2 f_p^2}(\underbrace{e^{-\pi^2 F_b a^2 (f_t-\frac{F_c}{a})^2}}_{Morlet\ Wavelet} - \underbrace{e^{-\pi^2 F_b a^2 (f_t^2+(\frac{F_c}{a})^2)}}_{correction\ term}), \tag{4.30}$$

where f_t denotes the frequency in the transform direction and f_p denotes the frequency in the projection direction with:

$$\begin{aligned} f_t &= f_x\cos\theta + f_y\sin\theta, \\ f_p &= f_x\sin\theta - f_y\cos\theta. \end{aligned} \tag{4.31}$$

---

[7] This condition is not sufficient for (4.26), but almost. If $\hat{\Psi}_{a,b,c,\theta}(0,0) = 0$ *and* $\hat{\Psi}_{a,b,c,\theta}(f_x,f_y)$ is continuously differentiable, then (4.26) is satisfied [Mal08].

Another important parameter of the Morlet wavelet is the FWHM (full width at half maximum), or more precisely, "the number of cycles [Coh19]". According to (4.29), the standard deviation of the window function in t-direction is $\sigma = a\sqrt{\frac{F_b}{2}}$. From this follows the FWHM: $2a\sqrt{\ln 2F_b}$. Compared with the length of one cycle $\frac{a}{F_c}$, the number of cycles is obtained by $2\sqrt{\ln 2F_b}F_c$. Note that this parameter is independent of the scale a. Applying a large number of cycles, the frequency resolution is high and the temporal resolution is low, and vice versa for a a small number of cycles. An example is shown in Fig. 4.12.

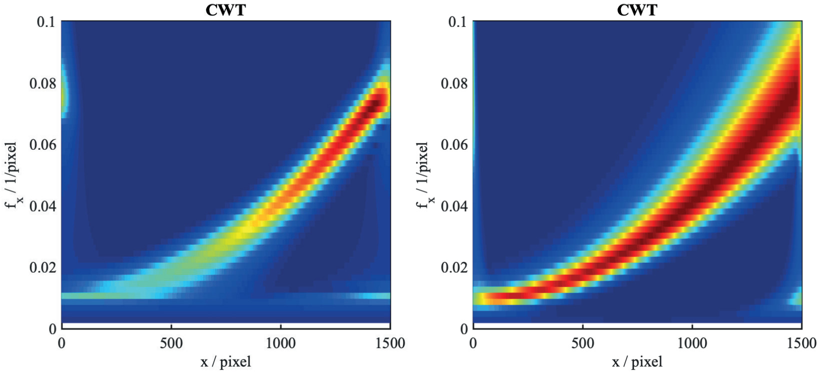


Fig. 4.12: The signal shown in Fig. 4.11 above is transformed into the time-frequency domain with the same Morlet wavelet but different F_b. Left: a large F_b (higher number of cycles) is applied; Right: a small F_b (lower number of cycles) is applied.

There is no definition of optimal number of cycles because it always depends on whether a high resolution in time or in frequency is expected. If the number of cycles is infinite, it turns out to be a Fourier transform with high frequency resolution but without any temporal information. The lowest number of cycles is discussed in [Coh19] and it is suggested not to set it lower than one: $\sqrt{F_b}F_c > \frac{1}{2\sqrt{\ln 2}}$. The lowest number of cycles is also limited by the correction term. The localized 2D ridgelet with a correction term in frequency domain introduced in (4.30) is plotted in Fig. 4.13 on the left hand side. Along the

t-direction (marked red on the left hand side) of the localized 2D ridgelet, it can be considered as the superposition of a positive Gaussian term centred at $\frac{F_c}{a}$ (Morlet wavelet) and a negative Gaussian term centred at 0 (correction term), which are plotted on the right hand side. The FWHM of both of the Gaussian terms are $\frac{2}{a\pi}\sqrt{\frac{\ln 2}{F_b}}$ according to (4.30). To avoid the unwanted influence of the correction term on the Morlet wavelet outside the zero position, the Morlet wavelet center should be greater than the FWHM, which gives $\sqrt{F_b}F_c > \frac{2}{\pi}\sqrt{\ln 2}$.

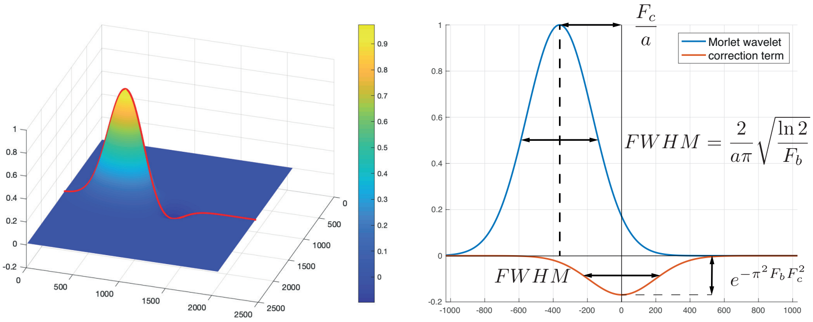


Fig. 4.13: Left: the 3D plot of a localized 2D ridgelet introduced in (4.30); Right: the Morlet wavelet term and the correction term along the red curved on left hand side plotted separately.

Utilizing a Morlet wavelet as the oscillation term has various advantages in the sinusoidal fringe pattern analysis. Firstly, the known pattern displayed on the screen is sinusoidal. The Morlet wavelet also having a sinusoidal oscillation can be matched to the pattern optimally. Secondly, the Morlet wavelet is analytic. It is suitable for applications where the phase information of a signal is to be separated from the amplitude. It can be understood as a single side-band demodulation (Chapter 2.2) with local information and self-adapted testing frequency and window size.

The advantage of the localized ridgelet transform is, firstly, it has all the benefits of Wavelet transforms - varying testing frequencies, self-adapted window size and local transform etc.. Secondly, the evaluation is sensitive to all fringe directions. Distorted fringe patterns can also be detected by varying θ. Thirdly, compared to the normal ridgelet transform, the localized ridgelet transforms the input signal into the time-frequency domain along the t-direction and at the same time the local information along the p-direction is retained, so that the local fringe orientation is allowed to vary throughout the image - which it does considerably for complex surface topographies of the object under test.

4.2.3 Fast Implementation

To implement the localized ridgelet transform in a fast way, the (fast) Fourier transform instead of the correlation[8] in (4.23) is applied. According to the convolution theorem, the Fourier transform of the correlation of $I(x,y)$ and $\psi_{a,b,c,\theta}(x,y)$ equals the pointwise product of their (fast) Fourier transforms $\hat{I}(f_x, f_y)$ and $\hat{\Psi}^*_{a,b,c,\theta}(f_x, f_y)$. Therefore, the localized ridgelet transform can be written as:

$$R_{\psi l}I(a,b,c,\theta) = \mathcal{F}^{-1}\{\hat{I}\hat{\Psi}^*_{a,b,c,\theta}\}, \tag{4.32}$$

where $\mathcal{F}^{-1}\{\cdot\}$ denotes the inverse Fourier transform. The fast implementation of the ridgelet transform is schematically shown in Fig. 4.14. The advantage of using the (fast) Fourier transform is that the input signal only needs to be transformed once. The Fourier transform of the localized ridgelet is analytically defined, and can be computed and stored in advance. The localized ridgelet can be performed only by using $n_\theta n_a$ inverse Fourier transforms with n_θ and n_a denoting the number of testing angles and dilations respectively.

The Fourier transform of the localized ridgelet with the correction term was defined in (4.30). The Fourier transform of the localized ridgelets are dilated by a and rotated by θ with regard to the Fourier transform of the original localized ridgelet. The computation of the localized ridgelets can therefore be simplified by just stretching and rotating the Fourier transform of the original

---

[8] As the (localized) ridgelet applied here is symmetrical with respect to the t-direction, correlation and convolution are the same.

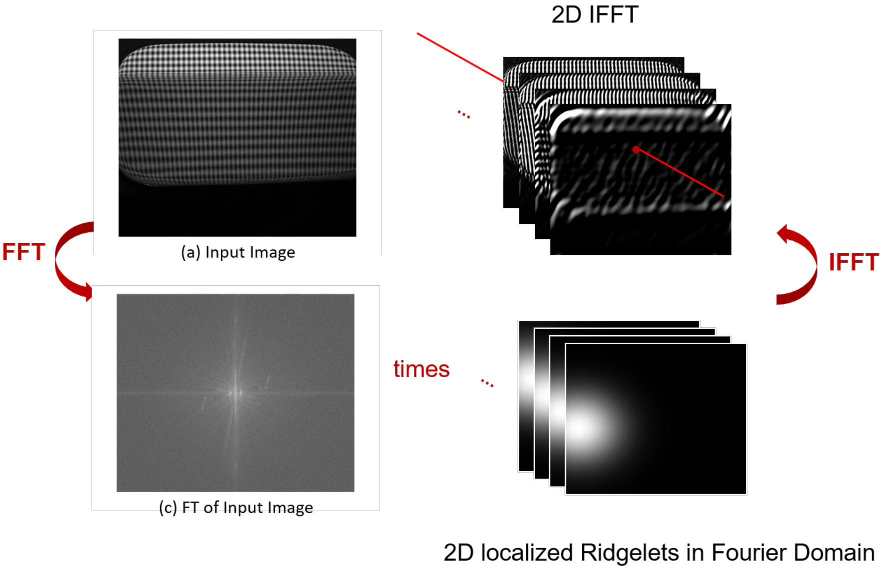


Fig. 4.14: The input image is Fourier transformed (left panel). The localized ridgelets are computed in the Fourier domain (bottom right). The product of the transformed image and the ridgelets in Fourier domain are inverse transformed respectively. The maximal modulus under all the tested angles and scales indicates the local frequency and local phase.

localized ridgelet as shown in Fig. 4.15. The angle parameter θ should be chosen according to the directional dynamics of the fringe pattern in the camera image. The dilation parameter a should be chosen so that the Heisenberg boxes of the wavelets cover the time-frequency domain. As shown in Fig. 4.6, a wavelet has its energy in time located in μ_x with a spread of $a_x\sigma_{os}$ while its energy in frequency centered at $\frac{\mu_{0f_x}}{a_x}$ with a spread of $\frac{\sigma_{of_x}}{a_x}$. To fully cover the time-frequency domain, the dilation parameter should be chosen as an exponential sequence [Mal08]. The Fourier transform of the original localized ridgelet is shown in Fig. 4.15 on the left hand side. This wavelet is rotated in the middle and stretched on the right hand side. The red grid demonstrates the sampling points. Note that the Fourier transform of the original localized ridgelet is analytically known. No interpolation is necessary after the rotation.

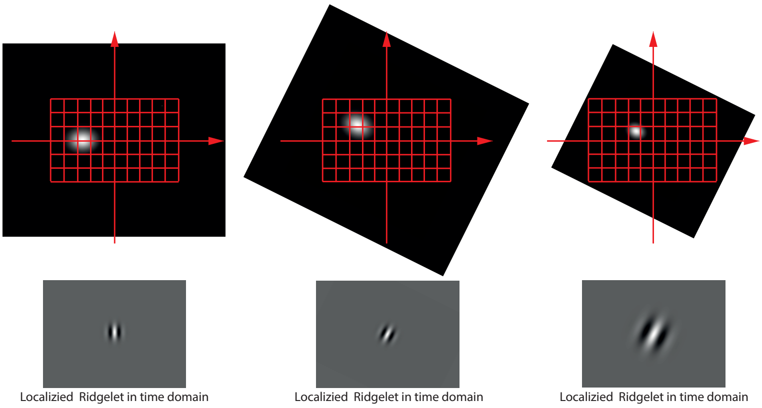


Fig. 4.15: From left to right: The original localized Ridgelet with $a = 1$ and $\theta = 0$; Varying θ by rotating the applied coordinate system; Varying a by dilating the applied coordinate system. Top: representation in Fourier domain; Bottom: representation in time comain.

4.2.4 Phase Evaluation

The local phase of the camera pixel point (b, c) is determined by locating the local maximum of the modulus of the localized ridgelet transform:

$$\theta_m, a_m = \arg \max_{\theta, a \in \mathbb{R}_{>0}} |\mathcal{F}^{-1}\{\hat{I}\hat{\Psi}^*_{a,b,c,\theta}\}|. \tag{4.33}$$

The phase $\phi(b, c)$ equals the angle of the complex value $\mathcal{F}^{-1}\{\hat{I}\hat{\Psi}^*_{a_m,b,c,\theta_m}\}$ to the real axis. Note that, applying the localized ridgelet transform to a cross pattern image results in *two* local maxima[9], at (θ_{mx}, a_{mx}) and (θ_{my}, a_{my}), since the signal has two dominant fringe directions. As shown in Fig. 4.16, the pattern on the left is generated by the superposition of a sinusoidal fringe pattern in the x -direction with a fringe period of 16 pixels and a sinusoidal fringe pattern in the y -direction with a fringe period of 32 pixels. This pattern is transformed using the localized ridgelets. The image on the right displays the modulus of the localized ridgelet transform at one pixel (marked yellow on the left) with the x-axis representing the varying angle θ and y-axis representing the varying dilation a. Two peaks indicating $|\mathcal{F}^{-1}\{\hat{I}\hat{\Psi}^*_{a_{mx},b,c,\theta_{mx}}\}|$ and $|\mathcal{F}^{-1}\{\hat{I}\hat{\Psi}^*_{a_{my},b,c,\theta_{my}}\}|$ are observed in the image on the right. As long as the two local maxima are separable in the angle-frequency domain, the x-phase $\phi_x(b,c)$ and the y-phase $\phi_x(b,c)$ can be obtained respectively.

The localized ridgelet transform is applied to evaluate the cross pattern image displayed in Fig. 2.12 (b), in which the screen position is encoded with a sinusoidal pattern of 16 pixels period in the x-direction and 24 pixels period in the y-direction, since the angular dynamic is much stronger in the y-direction. To verify the accuracy of the phase evaluation by using localized ridgelet transform, the object remaining in the same position is measured using the conventional phase shifting method and this phase is considered as the ground truth. As the screen positions in x-direction and in y-direction are encoded using different fringe periods, the phase deviations from the 'ground truth' are converted to screen position deviations for comparison and displayed in Fig. 4.17. The screen position deviation in x-direction varies from about -0.5 to 0.5 screen pixels and the screen position deviation in y-direction varies from about -1 to 1 screen pixels. Note that the screen pixel size is 294 μm. The distance between the object surface and the screen is about 100 mm, or more. One screen pixel deviation is, therefore, corresponding to a surface slope deviation of ± 5 $arcmin$.

There are still some outliers as displayed in Fig. 4.17. The outlier in the middle of the petrol cap (black circle) is introduced by a paint blister. Above the paint blister, there are horizontal outliers in the y-phase deviation (rectangle),

---

[9] degenerate case where one of the fringe directionss is lost due to a perfect astigmatic imaging by the object surface of the camera center onto a one-dimensional line screen as discussed in Chapter 3.2.2.

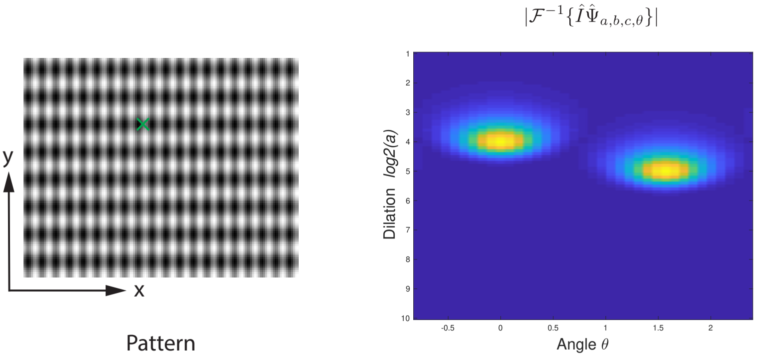


Fig. 4.16: Left: A pattern is generated by the superposition of a sinusoidal fringe pattern in the x-direction with a fringe period of 16 pixels and a sinusoidal fringe pattern in the y-direction with a fringe period of 32 pixels; Right: The modulus of the localized ridgelet transform of the pattern centered at the pixel marked green in the pattern is displayed with the x-axis representing the variation of the angle (θ) and the y-axis representing the (logarithmic) variation of the dilation parameter a.

which are introduced by the rapidly changing fringe frequencies and will be discussed in the next section.

4.3 Limits of the 2D Ridgelet Transform and Solution

4.3.1 Problem with Local Broadband Signals

As discussed above, the local phases of the cross pattern with a broadband spectrum can be evaluated quite well using the localised Ridgelet transform. However, even this evaluation can become a pitfall if the signal has a locally rapidly varying frequency.

To simplify the explanation, this problem will be discussed with an one

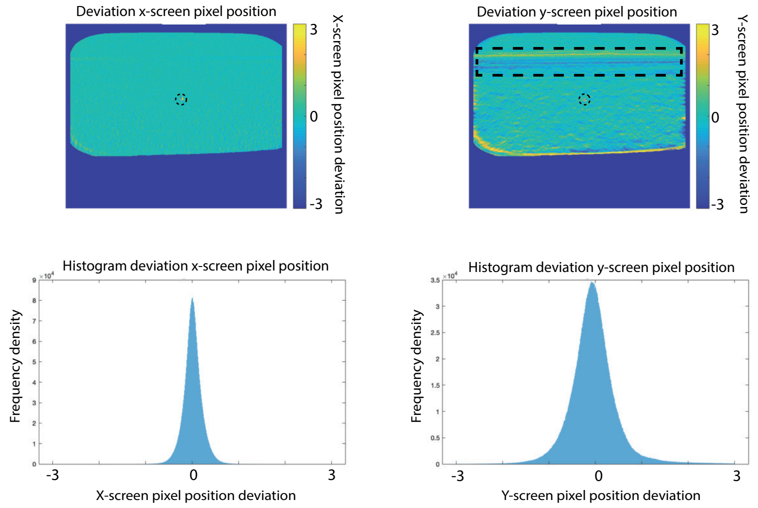


Fig. 4.17: Comparison of the phase evaluation of a petrol cap by using the localized ridgelet transform and by conventional using phase-shifting method. The phase deviations are all converted into units of ßcreen pixels"(SP) with 1 SP corresponding to a surface slope variation of 5 arcmin. Top left and top right: The phase deviations in x- and y-direction respectively. Bottom left and bottom right: the histogram of the deviations top left and top right respectively.

dimensional signal written as:

$$I(x) = I_0(x) + \frac{1}{2}M(x)e^{i\phi(x)} + \frac{1}{2}M(x)e^{-i\phi(x)}. \tag{4.34}$$

Note that both the background irradiance $I_0(x)$ and the modulation $M(x)$ vary very slowly compared to phase $\phi(x)$, and the phase $\phi(x)$ can be expanded to a Taylor series around b as follows:

$$\phi(x) = \phi(b) + \phi^{'}(b)(x-b) + \frac{\phi^{''}(b)}{2!}(x-b)^2 + ..., \tag{4.35}$$

Employing the Morlet wavelet [10] defined as:

$$\psi_{a,b}(x) = \frac{1}{a\sqrt{\pi F_b}} e^{i2\pi F_c(\frac{x-b}{a})} e^{-\frac{1}{F_b}(\frac{x-b}{a})^2}, \tag{4.36}$$

the 1D signal is transformed into the time-frequency domain. F_c is the center frequency, F_b is the wavelet bandwidth, a is the dilation (scale) parameter, and b is the translation parameter. Note that the Morlet wavelet defined above doesn't satisfy the admissibility condition. To overcome this, a correction term has to be added. For simplification, this term is not considered in the discussion below, as it can be safely neglected for sufficiently high frequencies.

The transform is computed by:

$$\begin{aligned} R_{\psi l}\{I\}(a,b) = &\int I_0(x)\psi^*_{a,b}(x)dx + \\ &\int \frac{1}{2}M(x)e^{i\phi(x)}\psi^*_{a,b}(x)dx + \\ &\int \frac{1}{2}M(x)e^{-i\phi(x)}\psi^*_{a,b}(x)dx \\ &= R_1 + R_2 + R_3. \end{aligned} \tag{4.37}$$

Since $I_0(x,y)$ varies only slightly, the first term in (4.37) is approximately 0. Employing the Taylor expansion up to the second order in (4.35) and assuming $M(x)$ to be a constant M_0, the second and third term of (4.37) becomes (s.

---

[10] In one dimensional problem the localized ridgelet equals a wavelet.

Appendix A.1):

$$\begin{aligned} R_2 &= \frac{1}{\sqrt{2}(4+M^2)^{\frac{1}{4}}} M_0 e^{-\frac{N_2}{4+M^2}} e^{i(\phi(b)+\frac{\theta}{2}-\frac{MN_2}{2(4+M^2)})}, \\ R_3 &= \frac{1}{\sqrt{2}(4+M^2)^{\frac{1}{4}}} M_0 e^{-(\frac{N_3}{4+M^2})} e^{-i(\phi(b)+\frac{\theta}{2}-\frac{MN_3}{2(4+M^2)})}, \end{aligned} \tag{4.38}$$

where

$$\begin{aligned} M &= a^2 F_b \phi''(b), \\ N_2 &= F_b(2\pi F_c - a\phi'(b))^2, \\ N_3 &= F_b(2\pi F_c + a\phi'(b))^2, \\ \theta &= \arctan(\tfrac{M}{2}). \end{aligned} , \tag{4.39}$$

Note that the first order derivative of the signal phase $\phi'(x)$ divided by 2π is the instantaneous frequency and the second order derivative of signal phase $\phi''(x)$ divided by 2π is the rate of change of the instantaneous frequency. If the second derivative of the phase $\phi''(x)$ is close to 0, (4.38) can be written as:

$$\begin{aligned} R_2 &= \tfrac{1}{2} M_0 e^{-\frac{F_b(2\pi F_c - a\phi'(b))^2}{4}} e^{i\phi(b)}, \\ R_3 &= \tfrac{1}{2} M_0 e^{-\frac{F_b(2\pi F_c + a\phi'(b))^2}{4}} e^{i\phi(b)}. \end{aligned} , \tag{4.40}$$

In the case of positive carrier frequency ($\phi'(b) > 0$), R_3 is approximately 0 and $\mid R_2 \mid$ represents the wavelet coefficient, or vice versa, in the case of negative carrier frequency ($\phi'(b) < 0$), R_2 is approximately 0 and $\mid R_3 \mid$ represents the wavelet coefficient. The local signal phase $\phi(b)$ is determined by computing the angle of the complex value of R_2 or R_3 and the maximal norm is obtained at $2\pi F_c - a\phi'(b) = 0$ or $2\pi F_c + a\phi'(b) = 0$, which represents the local frequency as $\frac{F_c}{a}$ or $-\frac{F_c}{a}$.

As already discussed in Chapter 4.2, the motherlet in (4.36) is normalized with $\frac{1}{a}$ (in 2D cases $\frac{1}{a^2}$) here, which differs from the common normalization $\frac{1}{\sqrt{a}}$ (in 2D cases $\frac{1}{a}$). The normalization with $\frac{1}{a}$ (in 2D cases $\frac{1}{a^2}$) ensures that the maximum norm of R_2 or R_3 in (4.40) always equals to $\frac{1}{2}M_0$ by $2\pi\frac{F_c}{a} = \phi'(b)$

or $2\pi\frac{F_c}{a} = -\phi'(b)$ respectively, through which the instantaneous frequency is determined. With the common normalization factor of $\frac{1}{\sqrt{a}}$, the maximal norm does not represent the local frequency anymore.

If the frequency of the input signal varies rapidly (locally broadband) - the second derivative of the phase $\phi^{''}(b)$ is not negligible, or if $\phi^{''}(b)$ is not continuous, the angle of the complex term R_2 or R_3 no longer represents exactly the phase of the input signal $\phi(b)$, but one has $\phi(b) + \frac{\theta}{2} - \frac{MN_2}{2(4+M^2)}$ as shown in (4.38). However, the second and third phase terms, $\frac{\theta}{2}$ and $-\frac{MN_2}{2(4+M^2)}$ both decrease when minimizing F_b. The error introduced by $\phi^{''}$ can be reduced by applying a smaller F_b which provides a better time resolution[11]. An example is shown in Fig. 4.18. The fringe phase in y-direction of the camera image of a petrol cap as shown in Fig. 2.13 is evaluated with the localized ridgelet transform with different F_b values. The determined phases converted into the screen pixel positions compared with the phase-shift evaluation are shown. The strong deviation shown in the figure on the left hand side is introduced by the rapidly varying fringe frequencies (s. the y-fringes shown in Fig. 2.13). This error is decreased by minimizing F_b from 1 to 0.6.

Another example is demonstrated in the following Fig. 4.19 and Fig. 4.20. A signal

$$I(x) = \begin{cases} \sin 2\pi\frac{x}{200} & \text{if } 0 \leq x < 2000 \\ \sin 2\pi(\frac{x^3}{4\cdot 10^7} - \frac{x^2}{10^4} + \frac{21x}{200}) & \text{if } 2000 \leq x \leq 3000 \end{cases}$$

is simulated and displayed in Fig. 4.19 (a). The instantaneous frequency of the signal

$$f(x) = \begin{cases} \frac{1}{200} & \text{if } 0 \leq x < 2000 \\ \frac{3x^2}{4\cdot 10^7} - \frac{2x}{10^4} + \frac{21}{200} & \text{if } 2000 \leq x \leq 3000 \end{cases}$$

and its phase are displayed in Fig. 4.19 (b) and (c).

The frequency evaluated with Morlet wavelet is shown Fig. 4.20 (a) in magenta. The signal phase (ground truth) and the determined phase are shown in Fig. 4.20 (b) in blue and magenta respectively. At $x = 2000$, where the second derivative

---

[11] Note, however, that other problems can appear by minimizing F_b. The minimal F_b was discussed in Chapter 4.2.

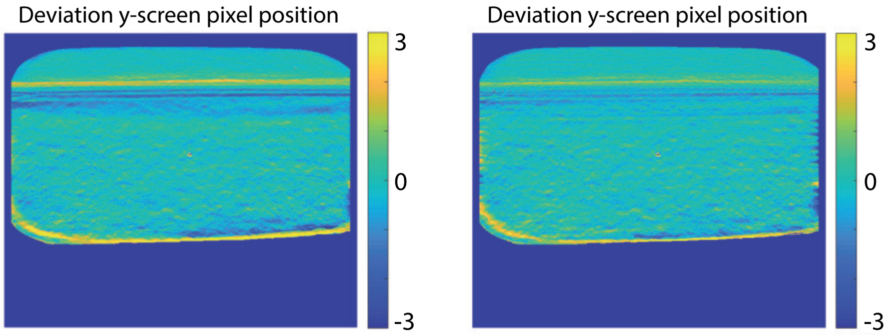


Fig. 4.18: The error of the y-phase evaluation (converted to the screen pixel position error) of the fringe pattern reflected on a petrol cap (Fig. 2.13) by using the localized ridgelet transform. Left: $F_b = 1$; Right: $F_b = 0.6$.

of the phase is not continuous, deviations arise in both the phase determination and the frequency determination. Also at $x > 2000$, where the second derivative of the phase is growing, the error of the frequency determination increases. However, at the same time the error of the phase determination is decreasing. Let $a_m = a_0 + \Delta a$, with a_m resulting in the maximum wavelet coefficient (norm), a_0 determining the actual instantaneous frequency $\frac{2\pi F_c}{a_0} = \phi^{'}(b)$ and Δa being the error. Assuming that $\Delta a \ll a_0$ and the error by determining the instantaneous frequency $\Delta\phi^{'}(b) = \frac{2\pi F_c}{a_m} - \phi^{'}(b)$ is in a reasonable range, the second and the third phase term,

$$\frac{\theta}{2} = \frac{\arctan \frac{a^2 F_b \phi^{''}(b)}{2}}{2}, \text{ and } -\frac{MN_2}{2(4+M^2)} = -\frac{a^4 F_b^2 \phi^{''}(b)(\frac{2\pi F_c}{a} - \phi^{'}(b))^2}{2(4+a^4 F_b^2 \phi^{''2}(b))}$$

of R_2 are decreasing while the scale a is dropping. Therefore, the disruptive effect of the rapidly varying frequencies on the phase determination is small by small a values, namely by high signal frequencies.

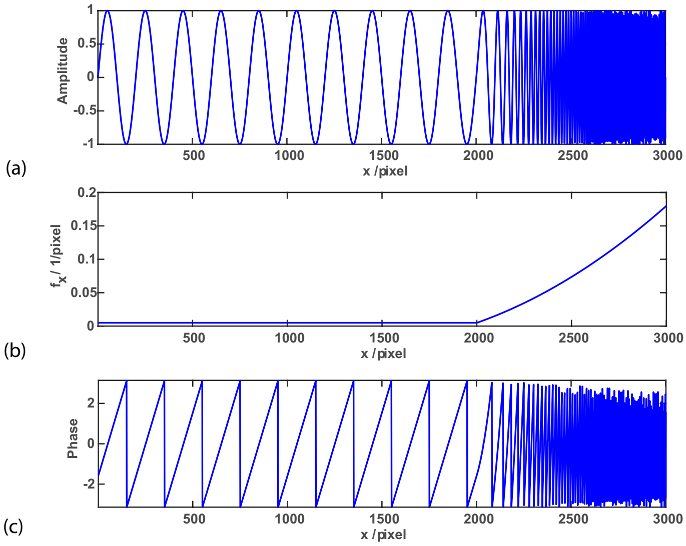


Fig. 4.19: (a) The amplitude of a chirp signal. (b) The frequency of a chirp signal (is not continuous at $x = 2000$). (c) The amplitude of a chirp signal.

4.3.2 Chirplet Transform

A signal where the second derivative of the instantaneous phase is called a chirp signal. *Chirplet Transform* (CT, [MH95]) is another time-frequency method which is specifically designed for the analysis of the chirp signals, and commonly applied to sonar, radar, and laser systems. A chirplet transform is similar to a wavelet transform but with its mother-let multiplied by a sweep term:

$$\psi_{a,b,\alpha}(x) = \frac{1}{a\sqrt{\pi F_b}} e^{i2\pi F_c(\frac{x-b}{a})} e^{-\frac{1}{F_b}(\frac{x-b}{a})^2} e^{i\pi\alpha(x-b)^2}, \tag{4.41}$$

where α represents the chirp rate - the rate of change of the frequency.

Similar to a wavelet transform, the chirplet transform is carried out by performing the correlation of the signal and the chirplet. Let the signal be $I(x) = I_0(x) + \frac{1}{2}M(x)e^{i\phi(x)} + \frac{1}{2}M(x)e^{-i\phi(x)}$ with the phase represented as

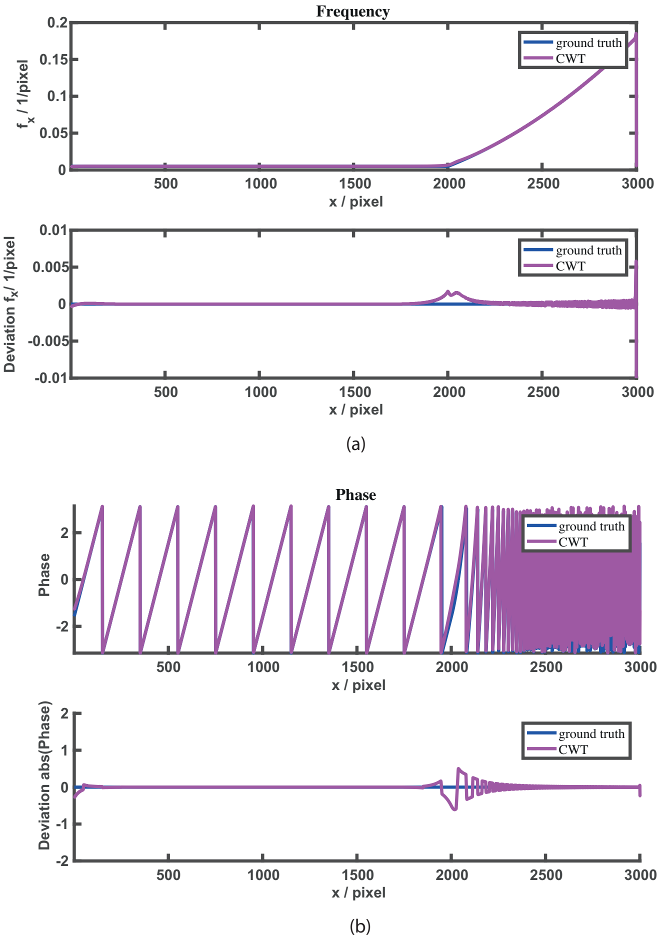


Fig. 4.20: The frequency and the phase of the signal shown in Fig. 4.19 is evaluated with the localized Ridgelet transform (in 1D case also Wavelet transform). (a): The computed frequency is shown in magenta in the upper image. To comparison the ground truth frequency is shown in blue. The deviation of the computed frequency from the ground truth is shown in the bottom image; (b): The computed phase is shown in magenta in the upper image. To comparison the ground truth phase is shown in blue. The deviation of the computed phase from the ground truth is shown in the bottom image.

$\phi(x) = \phi(b) + \phi^{'}(b)(x-b) + \frac{\phi^{''}(b)}{2!}(x-b)^2 + ...$ as defined in (4.34) and (4.35). The chirplet transform can be written as

$$
\begin{aligned}
C_\psi\{I\}(a,b) = \int I_0(x)\psi^*_{a,b,\alpha}(x)dx + \\
\int \frac{1}{2}M(x)e^{i\phi(x)}\psi^*_{a,b,\alpha}(x)dx + \\
\int \frac{1}{2}M(x)e^{-i\phi(x)}\psi^*_{a,b,\alpha}(x)dx \\
= C_1 + C_2 + C_3.
\end{aligned}
\tag{4.42}
$$

Assuming that $I_0(x)$ and $M(x)$ are constants, it follows that C_1 is also a constant and

$$
\begin{array}{l}
C_2 = \frac{1}{\sqrt{2}(4+M_2^2)^{\frac{1}{4}}} M_0 e^{-\frac{N_2}{4+M_2^2}} e^{i(\phi(b)+\frac{\theta_2}{2}-\frac{M_2N_3}{2(4+M_{c2}^2)})}, \\
C_3 = \frac{1}{\sqrt{2}(4+M_3^2)^{\frac{1}{4}}} M_0 e^{-(\frac{N_3}{4+M_3^2})} e^{-i(\phi(b)+\frac{\theta_3}{2}-\frac{M_3N_3}{2(4+M_{c3}^2)})},
\end{array}
\tag{4.43}
$$

where

$$
\begin{array}{l}
M_2 = a^2F_b(\phi''(b) - 2\pi\alpha) \\
M_3 = a^2F_b(\phi''(b) + 2\pi\alpha) \\
N_2 = F_b(2\pi F_c - a\phi'(b))^2, \\
N_3 = F_b(2\pi F_c + a\phi'(b))^2, \\
\theta_2 = \arctan(\frac{M_2}{2}), \\
\theta_3 = \arctan(\frac{M_3}{2}).
\end{array},
\tag{4.44}
$$

Rewrite the norm of C_2 as a function of p and q:

$$
\begin{aligned}
\mid C_2(p,q) \mid &= \frac{1}{\sqrt{2}(4+M_2^2)^{\frac{1}{4}}} M_0 e^{-\frac{N_2}{4+M_2^2}} \\
&= \frac{1}{\sqrt{2}(4+a^4F_b^2p^2)^{\frac{1}{4}}} M_0 e^{-\frac{F_bq^2}{4+a^4F_b^2p^2}},
\end{aligned}
\tag{4.45}
$$

with $q = 2\pi F_c - a\phi'(b)$ and $p = \phi''(b) - 2\pi\alpha$, and plot $\mid C_2(p,q) \mid$ in Fig. 4.21. The maximal norm is present when both p and q equal zero. $a = 2\pi\frac{F_c}{\phi'(b)}$ and

$\alpha = \frac{\phi''(b)}{2\pi}$, indicating the instantaneous frequency and the rate of change of the frequency respectively, can be determined by locating the maximal norm. When $a = 2\pi\frac{F_c}{\phi'(b)}$ and $\alpha = \frac{\phi''(b)}{2\pi}$, the second and third phase term of C_2, $\frac{\theta}{2}$ and $-\frac{M_2 N_3}{2(4+M_{c2}^2)}$ both equal zero and only the first phase term $\phi(b)$ remains. The disruptive effect introduced by the signal phase in second derivative is compensated by the chirp rate parameter α.

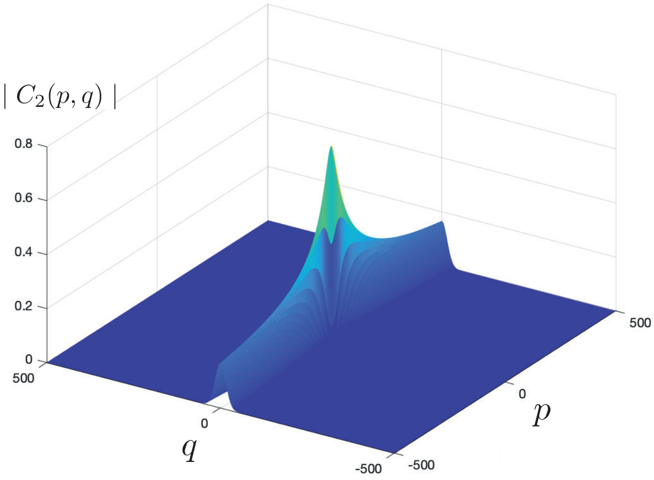


Fig. 4.21: Plotting $| C_2(p,q) |$ as a function of p and q as presented in (4.45). $a = 2$ and $F_b = \frac{0.36}{2\pi^2}$ are applied for the plotting.

Applying the chirplet transform to the signal shown in Fig. 4.19, the phase of the signal is determined and the phase error is displayed in Fig. 4.22 above. The phase error of a CWT evaluation is plotted in Fig. 4.22 in magenta for comparison. At $x = 2000$, where the second derivative of the phase is not continuous, there is no significant difference between these two methods in the respect of phase determination (Fig. 4.22 (a)). For $x > 2000$, where the signal frequency rate is non-zero, CT shows its potential to reduce measurement inaccuracy due to the rapid changing signal frequency (Fig. 4.22 (b)).

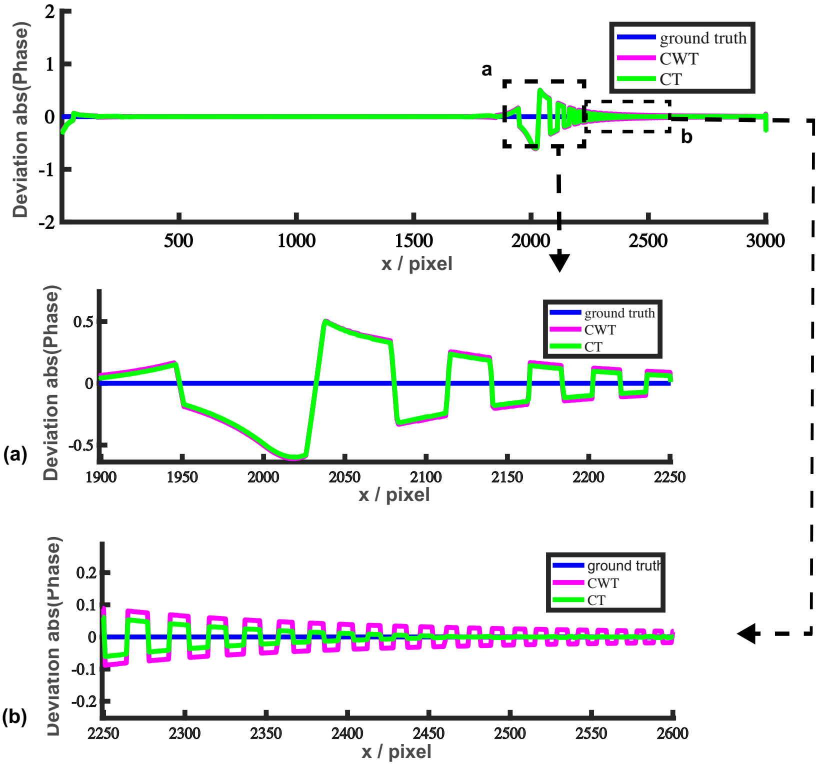


Fig. 4.22: Determining the phase of the signal shown in Fig. 4.19 by applying CWT and CT.

4.3.3 Critical Area Recognition

The second order derivative of the signal phase can be evaluated by the chirplet transform, in which an additional parameter α is introduced compared to the wavelet transform. The chirplet transform of an 1D signal is an $\mathbb{R} \to \mathbb{R}^3$ projection from the time domain to the time-frequency-chirp domain, $f(x) \rightarrow g(a, b, \alpha)$. Analogously, applying a chirp term to the localized 2D ridgelet transform results in an $\mathbb{R}^2 \rightarrow \mathbb{R}^5$ projection, which is $f(x, y) \rightarrow g(a, b, c, \theta, \alpha)$. The expansion into a higher dimensional space significantly increases the computation time. Therefore, although the chirplet transform is a possible solution for phase evaluation of fringe patterns with locally rapidly changing frequencies, it is not suitable for industrial use due to the time-consuming computation.

Since the inaccuracy of the phase evaluation is introduced by its second-order derivative when the localized 2D-Ridgelet transform is applied, the "critical" region can be detected by the frequency change rate, which can be estimated after the localized 2D-Ridgelet transform. Calculating the gradient of the phase in x- and y- directions [12], the norm of the gradient gives the instantaneous frequency. Calculating the gradient of the instantaneous frequency itself, the norm of the gradient gives the rate of change of the instantaneous frequency. The frequency rate of change evaluated in this way is not completely accurate since the phase estimated by the localized 2D-Ridgelet transform is not reliable if the signal frequency varies rapidly. However, it is accurate enough for locating the "critical" region for a localized 2D-Ridgelet transform.

The frequency rate of change α of the fringe pattern in the y-direction of the camera image shown in Fig. 2.13 is evaluated by the localized 2D ridgelet transform and by performing the gradient of the phase. The result is shown in Fig. 4.23. The "critical" region can be identified and localized very well by large values of the frequency change rate. If necessary, the phase evaluation can be improved by applying a chirplet transform only in these critical regions.

---

[12] The gradient muss be performed since the phase does not change along the x- or y-direction, but in any direction.

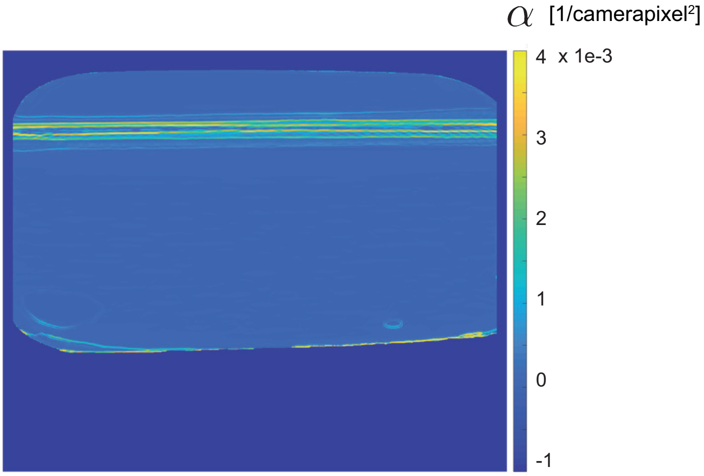


Fig. 4.23: The chirp rate of the fringe pattern (in y direction) reflected on a petrol cap as shown in Fig. 2.13.

Chapter 5

Monocular Solution of the Height Problem

The height-normal ambiguity caused by the unknown direction of the incident ray described in Chapter 2.1.4 complicates the 3D reconstruction of a specular surface considerably. Several solutions addressing this problem are discussed in Chapter 2.1.4. The majority of the solutions for this problem require different kinds of changes and adaptations on the hardware like a second camera [KHL04a; Kic07], a second screen position [BSG06] or a specific camera with a telecentric aperture [SH00]. It is also possible to find the absolute surface without any hardware changes by using the ‚Frobenius condition‘ [Kam04b], but this approach is not suitable for practical use since the local integrability condition exploited in this approach is very weak and the vector field only varies extremely slightly for typical test objects like a spherical mirror or a plane mirror.

A new method solving the height-normal ambiguity without any change of the PMD setup is possible by utilizing, adapting and advancing an approach suggested by Savarese [SCP05] based on the local differential geometry of a reflected checkerboard pattern. This approach is hereafter implemented and considerably extended for the new use case of PMD evaluation. The new method developed here allows for a monocular absolute PMD measurement of the entire surface area, which was not possible neither in Savarese's original proposal nor with the original PMD evaluation.

5.1 The Principle of the “Savarese Method”

The principle of the Savarese method is discussed in full detail in [SCP04][SCP05]. It will not be repeated in full in this thesis. Only the part which is important as a starting point for the solution of the height problem of PMD will be re-introduced and its physical meaning will be discussed.

Savarese applied a checkerboard pattern and an observing camera to measure discrete points on specular surfaces. The checkerboard pattern, which can be printed on a paper or displayed on a screen (called screen plane or scene plane), is reflected on the specular surface, and the reflected pattern is captured by the camera. The checkerboard and the camera are both calibrated and known. The corners of the checkerboard’s squares are the intersection points to be measured. The correspondence between the intersection points on the pattern and those in the camera image are supposed to be known as well. Two scene lines - two crossing lines lying on the screen - are defined by five intersection points on the screen plane (Fig. 5.1 left). Their correspondences on the camera plane are defined (Fig. 5.1 right) by interpolating those five corresponding neighboring intersection points on the camera plane. Applying the local differential geometry of the two curves, the position of the reflection object point is evaluated.

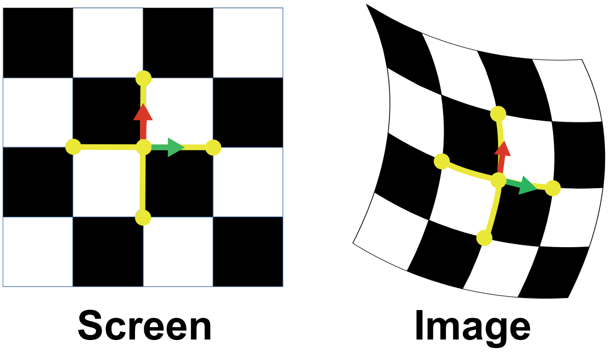


Fig. 5.1: The correspondence of screen and camera image are obtained by the intersecting points of the checkerboard pattern. Five neighboring points (yellow) in the screen define two intersecting lines (yellow). The green and the red arrows are the direction vectors of the intersection lines. The five corresponding points in the camera image can be found, and consequently, the corresponding direction vectors of the intersection curves are determined on the image plane.

Notations and the basic geometry are displayed in Fig. 5.2. The intersecting point on the checkerboard is denoted by $\mathbf{r}_{s0}$. $\mathbf{r}_{s0}$ is reflected by the object point $\mathbf{r}_0$ and observed by (and therefore mapped onto) the point $\mathbf{r}_{c0}$ on the camera image plane. $\hat{\mathbf{d}}$ is the unit vector from the camera projection center $\mathbf{c}$ to $\mathbf{r}_{c0}$, which defines the normalized direction vector of the corresponding ray of sight, known by the internal calibration of the camera. The distance between $\mathbf{c}$ and the image plane is the camera constant l of the utilized pinhole camera model. The distance between $\mathbf{c}$ and $\mathbf{r}_0$ is denoted by s and the angle of incidence on the object surface is denoted by θ.

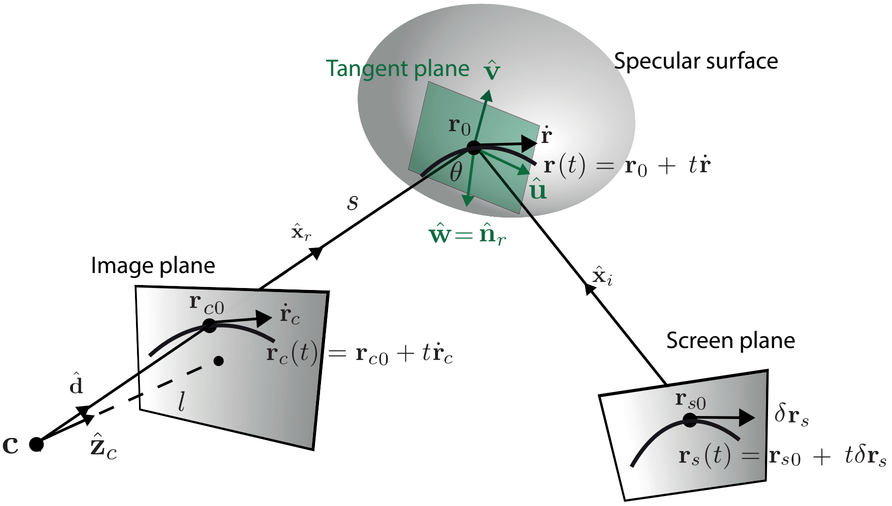


Fig. 5.2: Geometry of specular reflections. A curve lying on the camera plane is denoted by $\mathbf{r}_c(t) = \mathbf{r}_{c0} + t\dot{\mathbf{r}}_c$. The perspective projection of $\mathbf{r}_c(t) = \mathbf{r}_{c0} + t\dot{\mathbf{r}}_c$ onto the tangent plane of the object surface is written as $\mathbf{r}(t) = \mathbf{r}_0 + t\dot{\mathbf{r}}$ (camera projection center: $\mathbf{c}$). The correspondence curve on the screen plane is denoted by $\mathbf{r}_s(t) = \mathbf{r}_{s0} + t\delta\mathbf{r}_s$. ($\hat{\mathbf{u}}\ \hat{\mathbf{v}}\ \hat{\mathbf{w}}$) is the local principal reference coordinate system defined by the surface normal $\hat{\mathbf{n}}_r$ and the principal plane spanned by the incident and reflected ray to simplify the surface representation.

A so-called principal reference system ($\hat{\mathbf{u}}\ \hat{\mathbf{v}}\ \hat{\mathbf{w}}$) is introduced in [SCP05] for each observation point to simplify the surface representation. The object point $\mathbf{r}_0$ defines the coordinate origin of the reference system. $\hat{\mathbf{w}}$ is given by the surface normal vector $\hat{\mathbf{n}}_r$, which is perpendicular to the tangent plane of the surface. $\hat{\mathbf{v}}$ is the unit normal vector of the so-called “principal plane” spanned by the incident ray $\mathbf{r}_0 - \mathbf{r}_{s0}$ and the reflected ray $\mathbf{r}_0 - \mathbf{r}_{c0}$. Note that $\hat{\mathbf{v}}$ lies in the tangent plane. According to the law of reflection, $\hat{\mathbf{n}}_r$ lies in the principal plane and bisects the angle between the incident ray and the reflected ray. $\hat{\mathbf{u}}$ is given by $\hat{\mathbf{v}} \times \hat{\mathbf{w}}$ and lies in both the tangent plane and also the principal plane. The local surface around $\mathbf{r}_0$ can now be written as (*special Monge form in an implicit form*, Eq. (14) in [SCP05]):

$$g(u,v,w) = w - \frac{1}{2!}(au^2 + 2cuv + bv^2) - \frac{1}{3!}(eu^3 + 3fu^2v + 3guv^2 + hv^3) + ... = 0, \tag{5.1}$$

where *a,b,c* and *e, f, g, h* are the second and the third-order surface parameter around $\mathbf{r}_0$ respectively, and (u, v, w) are the Cartesian coordinates of a given vector $\hat{\mathbf{r}} = u\hat{\mathbf{u}} + v\hat{\mathbf{v}} + w\hat{\mathbf{w}}$ in the system defined by $(\hat{\mathbf{u}}\ \hat{\mathbf{v}}\ \hat{\mathbf{w}})$.

A curve passing through a screen point $\mathbf{r}_{s0}$ in the checkerboard can be described in parametric form: $\mathbf{r}_s(t) = \mathbf{r}_{s0} + t\delta\mathbf{r}_s$, where t is a parameter and $\delta\mathbf{r}_s$ is the 3D direction vector of the curve. $\mathbf{r}_{si}(t)$ denotes the ith space curve lying on the screen plane[1]. $\mathbf{r}_{si}(t)$ is reflected on the object surface and captured by (and therefore mapped to) the camera. The image of $\mathbf{r}_{si}(t)$ lying on the camera plane is denoted by $\mathbf{r}_{ci}(t) = \mathbf{r}_{c0} + t\dot{\mathbf{r}}_{ci}$. $\mathbf{r}_i(t) = \mathbf{r}_0 + t\dot{\mathbf{r}}_i$ denotes the perspective projection of $\mathbf{r}_{ci}(t) = \mathbf{r}_{c0} + t\dot{\mathbf{r}}_{ci}$ onto the object surface. Taking the linear approximation, the curve $\mathbf{r}_i(t)$ lies within the tangent plane of the specular surface defined by $\hat{\mathbf{w}}$ or $\hat{\mathbf{n}}_r$.

The direction of the incident ray $\hat{\mathbf{x}}_i$ and reflected ray $\hat{\mathbf{x}}_r$ can be obtained from (Eq. (16) (17) in [SCP05]):

$$\begin{aligned} \hat{\mathbf{x}}_i &= \frac{\mathbf{r}_0 - \mathbf{r}_s(t)}{\|\mathbf{r}_0 - \mathbf{r}_s(t)\|}, \\ \hat{\mathbf{x}}_r &= \frac{\mathbf{r}_0 - \mathbf{c}}{\|\mathbf{r}_0 - \mathbf{c}\|}. \end{aligned} \tag{5.2}$$

Applying the law of reflection - the direction of the normal vector $\hat{\mathbf{n}}_r$ (in the

---

[1] According to Fig. 5.1, at least two intersecting curves have to be considered for each surface point

implicit form $\nabla g(\mathbf{r})$ [2] of the reflection point is parallel to $\hat{\mathbf{x}}_r + \hat{\mathbf{x}}_i$, therefore (Eq. (15) in [SCP05]):

$$\begin{cases} \hat{\mathbf{x}}_i(\mathbf{r}, t) + \hat{\mathbf{x}}_r(\mathbf{r}) + \lambda \nabla g(\mathbf{r}) = 0 \\ g(\mathbf{r}) = 0 \end{cases} \tag{5.3}$$

where λ is a scaling parameter. It can be denoted by $\mathbf{F}(t, \mathbf{r}, \lambda) = \mathbf{0}$ [3]. By differentiating $\mathbf{F}$ with respect to t and taking the chain rule into account, a new equation is obtained (Eq. (19) in [SCP05][4]):

$$\mathbf{B}(t, \mathbf{r}, \lambda) = \mathbf{J}(t, \mathbf{r}, \lambda)\mathbf{S}(t), \tag{5.4}$$

where $\mathbf{B}(t, \mathbf{r}, \lambda) = \frac{\partial \mathbf{F}(t,\mathbf{r},\lambda)}{\partial t}$, $\mathbf{J}(t, \mathbf{r}, \lambda) = \frac{\partial \mathbf{F}(t,\mathbf{r},\lambda)}{\partial(\mathbf{r},\lambda)}$, $\mathbf{S}(t) = [\dot{r}_u(t)\ \dot{r}_v(t)\ \dot{r}_w(t)\ \dot{\lambda}(t)]^T = [\dot{\mathbf{r}}(t)\ \dot{\lambda}(t)]^T$.

Performing the partial derivative of $\partial \mathbf{F}(t, \mathbf{r}, \lambda)$ with respect to t and $\mathbf{r}$ respectively, it turns out (the entire deduction in details is shown in Appendix. A.2 and Appendix. A.3, Eq. (21) and Eq. (21) in [SCP05]):

$$\mathbf{B} = \begin{bmatrix} -\frac{(\mathbf{I}-\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T)\delta\mathbf{r}_s}{|\mathbf{r}-\mathbf{r}_s(t)|} \\ 0 \end{bmatrix}, \text{and} \tag{5.5}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\mathbf{I}-\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T}{|\mathbf{r}-\mathbf{r}_s(t)|} + \frac{\mathbf{I}-\hat{\mathbf{x}}_r\hat{\mathbf{x}}_r^T}{|\mathbf{r}-\mathbf{c}|} + \frac{\partial \lambda \nabla g(\mathbf{r})}{\partial \mathbf{r}} & \nabla g(\mathbf{r}) \\ (\nabla g(\mathbf{r}))^T & 0 \end{bmatrix}. \tag{5.6}$$

Expressing $\hat{\mathbf{x}}_r$ and $\hat{\mathbf{x}}_i$ in the principal reference system in terms of the

---

[2] Note that the implicit function $g(u, v, w)$ is always invertible. $g(u, v, w)$ is obtained by the explicit function $w(u, v) = \frac{1}{2!}(au^2 + 2cuv + bv^2) + \frac{1}{3!}(eu^3 + 3fu^2v + 3guv^2 + hv^3) + ...$ The derivative of $g(u, v, w)$ with respect to w is always 1. Therefore, $g(u, v, w)$ is always invertible.

[3] $\mathbf{F}$ denotes the Lagrange multiplier, with $\mathbf{F} = [F_1, F_2, F_3, F_4]^T$ and:
$F_1 = \hat{\mathbf{x}}_{r,u} + \hat{\mathbf{x}}_{i,u} + \lambda \nabla g(\mathbf{r})_u$,
$F_2 = \hat{\mathbf{x}}_{r,v} + \hat{\mathbf{x}}_{i,v} + \lambda \nabla g(\mathbf{r})_v$,
$F_3 = \hat{\mathbf{x}}_{r,w} + \hat{\mathbf{x}}_{i,w} + \lambda \nabla g(\mathbf{r})_w$,
and $F_4 = g(\mathbf{r})$,
where $\nabla g(\mathbf{r})_u$ and $\nabla g(\mathbf{r})_u$ always equal to zero.

[4] The original Eq. (19) in [SCP05] is $\mathbf{B}(t, \mathbf{r}, \lambda) = -\mathbf{J}(t, \mathbf{r}, \lambda)\mathbf{S}(t)$, in which the sign is not correct according to the chain rule. However, in Savarese's proposal only the norms of two vectors are compared at the end, therefore, the wrong sign has no effect on the result.

reflection angle θ ([SCP05]), which is a function only of the length of the reflection ray s:

$$\hat{\mathbf{x}}_i(s) = (-\sin\theta(s), 0, -\cos\theta(s))^T, \hat{\mathbf{x}}_r(s) = (\sin\theta(s), 0, -\cos\theta(s))^T. \qquad (5.7)$$

(5.4) can be rewritten as (only taking the surface parameter up to the second-order a, b and c into account, Eq. (19) in [SCP05]):

$$\begin{bmatrix} B_u(s) \\ B_v(s) \\ B_w(s) \\ 0 \end{bmatrix} = \begin{bmatrix} J_u(s) - 2a\cos\theta(s) & -2c\cos\theta(s) & J_w(s) & 0 \\ -2c\cos\theta(s) & J_v(s) - 2b\cos\theta(s) & 0 & 0 \\ J_w(s) & 0 & J_v(s)\sin^2\theta(s) & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{r}_u \\ \dot{r}_v \\ \dot{r}_w \\ \dot{\lambda} \end{bmatrix}, \qquad (5.8)$$

where (Eq. (10) and (24) in [SCP05])

$$\begin{aligned} B_u(s) &= \frac{\delta r_{s,w}\cos\theta(s)\sin\theta(s) - \delta r_{s,u}\cos^2\theta(s)}{\mid \mathbf{r}_{s0}(s) \mid}, \\ B_v(s) &= -\frac{\delta r_{s,v}}{\mid \mathbf{r}_{s0}(s) \mid}, \\ B_w(s) &= \frac{\delta r_{s,u}\cos\theta(s)\sin\theta(s) - \delta r_{s,w}\sin^2\theta(s)}{\mid \mathbf{r}_{s0}(s) \mid}, \end{aligned} \qquad (5.9)$$

with $\delta r_{s,u}$, $\delta r_{s,v}$ and $\delta r_{s,w}$ denoting the projection of $\delta\mathbf{r}_s$ (the measurand) on $\hat{\mathbf{u}}$, $\hat{\mathbf{v}}$ and $\hat{\mathbf{w}}$ respectively, and (Eq. (11) and (25) in [SCP05])

$$\begin{aligned} J_u(s) &= J_v(s)\cos^2\theta(s), \\ J_v(s) &= \frac{1}{s} + \frac{1}{|\mathbf{r}_{s0}(s)\mid}, \\ J_w(s) &= \frac{\mid \mathbf{r}_{s0}(s) \mid - s}{\mid \mathbf{r}_{s0}(s) \mid s}\cos\theta(s)\sin\theta(s). \end{aligned} \qquad (5.10)$$

Note that s is the distance from the camera projection center $\mathbf{c}$ to the object point $\mathbf{r}_0$, which is the first surface parameter. The reflection angle θ of a given camera pixel depends only on s. $\mid \mathbf{r}_{s0} \mid$ is the distance between the surface point

$\mathbf{r}_0$ and the screen point $\mathbf{r}_{s0}$, which also only depends on s if a camera pixel is given. Therefore, J_u, J_v and J_w are only functions of s. Since $\delta r_{s,u}$, $\delta r_{s,v}$ and $\delta r_{s,w}$ are measured values, B_u, B_v and B_w are thus only functions of s as well. (5.8) is the core relation of this differential problem (the height problem of PMD). It is obtained according to the chain rule applied on (5.3). This equation can be interpreted and understood more comprehensibly by considering its physical meaning, which is *not* discussed in [SCP05]. (5.5) implies that $\mathbf{B}$ is the projection of $\delta\mathbf{r}_s$ onto a plane which is perpendicular to the incident ray $\hat{\mathbf{x}}_i$ (the red plane in Fig. 5.3) and scaled with the inverse of $\mid \mathbf{r}_{s0} \mid$. The first three equations of (5.8) is the mapping of the vector $\dot{\mathbf{r}}$ (corresponding to $\delta\mathbf{r}_s$ on the screen) on the tangent plane onto $[B_u\ B_v\ B_w]^T$. This mapping matrix is completely determined by the surface parameter s, a, b and c.

Since $\dot{\mathbf{r}}$ lies on the tangent plane, and also because of the last equation of (5.8), $\dot{r}_w$ always equals to 0. Ignoring the third equation in (5.8) and taking the inverse of the remaining matrix,

$$\begin{bmatrix}\dot{r}_u\\ \dot{r}_v\end{bmatrix} = \frac{1}{d(s,a,b,c)}\begin{bmatrix}J_v(s) - 2b\cos\theta(s) & 2c\cos\theta(s)\\ 2c\cos\theta(s) & J_u(s) - 2a\cos\theta(s)\end{bmatrix}\begin{bmatrix}B_u(s)\\ B_v(s)\end{bmatrix}, \quad (5.11)$$

is obtained[5] (Eq. (88) in [SCP05]), where

$$d(s,a,b,c) = (J_u(s) - 2a\cos\theta(s))(J_v - 2b\cos\theta(s)) - 4c^2\cos^2\theta(s) \quad (5.12)$$

is the determinant. Represent the 2×2 matrix in (5.11) with

$$\mathbf{M}(s,a,b,c) = \begin{bmatrix}M_2(s,b)\ M_3(s,c)\\ M_3(s,c)\ M_1(s,a)\end{bmatrix}, \quad (5.13)$$

---

[5] The analytically inverted matrix is only for the theoretical derivation. In the actual computation, of course numerically more stable methods are utilized.

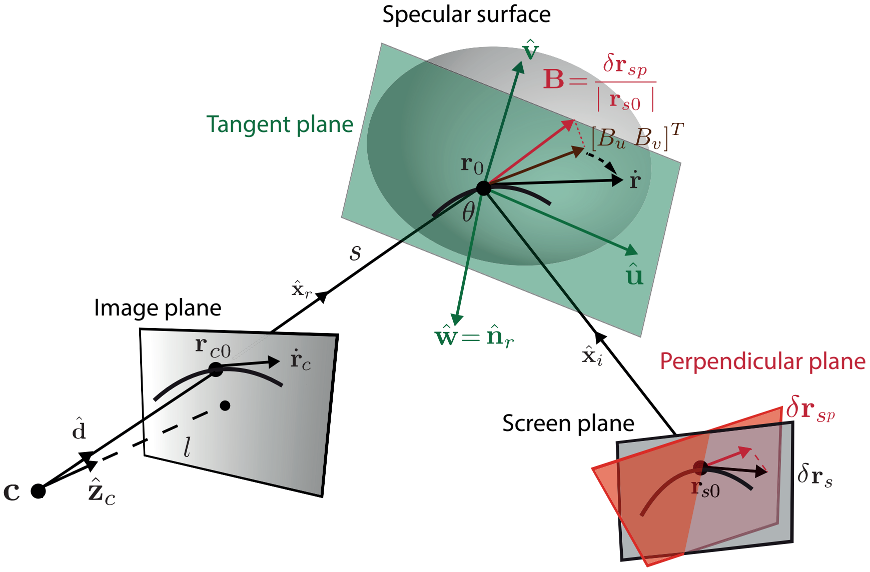


Fig. 5.3: The physical interpretation of (5.4) and (5.11)(not given in [SCP05]). The 3D direction vector of the screen curve $\delta\mathbf{r}_s$ projected onto a plane perpendicular to the incident ray $\hat{\mathbf{x}}_i$ (the red plane) is denoted by $\delta\mathbf{r}_{sp}$. $\mathbf{B}$ equals to $\delta\mathbf{r}_{sp}$ while being scaled with $\frac{1}{|\mathbf{r}_{s0}|}$ (s. (5.9)) - the inverse of the length of the incident ray. The geometrical interpretation of (5.11) is a 2D mapping on the tangent plane (the green plane), which maps the vector $[B_u\ B_v]^T$ (the projection of $\mathbf{B}$ onto the tangent plane) to $[\dot{r}_u\ \dot{r}_v]^T$ (the projection of $\dot{\mathbf{r}}_c$ onto the tangent plane). The mapping is illustrated by the black dashed arrow.

where

$$\begin{aligned} M_1(s,a) &= J_u(s) - 2a\cos\theta(s), \\ M_2(s,b) &= J_v(s) - 2b\cos\theta(s), \\ M_3(s,c) &= 2c\cos\theta(s). \end{aligned} \tag{5.14}$$

$\frac{1}{M_1 M_2 - M_3^2}\mathbf{M}$ can be interpreted as a 2D mapping matrix on the tangent plane, which maps the vector $[B_u(s)\ B_v(s)]^T$ (the projection of $\mathbf{B}$ onto the tangent plane, also denoted by $\mathbf{B}_{uv}$) to $[\dot{r}_u \ \dot{r}_v]^T$.

Rewriting (5.11) as

$$\begin{bmatrix} \dot{r}_u \\ \dot{r}_v \end{bmatrix} = \frac{1}{M_1(s,a)M_2(s,b) - M_3^2(s,c)} \begin{bmatrix} M_2(s,b) & M_3(s,c) \\ M_3(s,c) & M_1(s,a) \end{bmatrix} \begin{bmatrix} B_u(s) \\ B_v(s) \end{bmatrix}, \tag{5.15}$$

it is a non-linear system of 2 equations with four unknowns (a, b, c and s). Therefore, to recover a surface point, theoretically only two noncollinear intersecting lines are needed. But to recover the whole surface, for every single pixel point there is a new non-linear system with four variables to be solved. This problem can be simplified according to [SCP05] by introducing a new auxiliary parameter ϕ, through which the four variables in (5.11) can be reduced to two. Considering two scene lines, a new equation with only one variable can be generated. ϕ is the angle between $\dot{\mathbf{r}}$ and $\hat{\mathbf{u}}$ axis on the tangent plane, and $\tan\phi$ can be expressed as $\frac{\dot{r}_v}{\dot{r}_u}$. According to (5.11), the tangent direction $\tan\phi$ can be computed by (Eq. (68) in [SCP05]):

$$\tan\phi = \frac{\dot{r}_v}{\dot{r}_u} = \frac{M_1(s,a)B_v(s) + M_3(s,c)B_u(s)}{M_2(s,b)B_u(s) + M_3(s,c)B_v(s)}. \tag{5.16}$$

The introduction of the angle ϕ is only an auxiliary construction for the evaluation of the mapping matrix on the tangent plane. It is necessary if, as input data, only the differential correspondence in two directions is available. However, this approach is quite disadvantageous, because $tan\phi$ introduces bad mathematical properties as $tan\frac{\pi}{2}$ is not defined. In Chapter 5.4, it will be shown that it is possible to compute the absolute height without introducing this auxiliary tangent value if the local differential correspondence is provided in full area in 2D as it is the case for PMD.

(5.16) can be represented as (Eq. (75) in [SCP05])

$$\begin{bmatrix} B_v(s) & -B_u(s)\tan\phi & B_u(s) - B_v(s)\tan\phi \end{bmatrix} \begin{bmatrix} M_1(s,a) \\ M_2(s,b) \\ M_3(s,c) \end{bmatrix} = 0. \tag{5.17}$$

In the following, the vector to the left is denoted by $\mathbf{h}$ with

$$\mathbf{h}(s,\phi) := \begin{bmatrix} B_v(s) & -B_u(s)\tan\phi & B_u(s) - B_v(s)\tan\phi \end{bmatrix}. \tag{5.18}$$

The vector on the right hand side is denoted by $\mathbf{m}$ with

$$\mathbf{m}(s,a,b,c) := [M_1(s,a) \;\; M_2(s,b) \;\; M_3(s,c)]^T. \tag{5.19}$$

Considering two coplanar scene lines intersecting at $\mathbf{r}_{s0}$, two vectors $\mathbf{h}_1(s,\phi_1)$ and $\mathbf{h}_2(s,\phi_2)$ are to be obtained. Since $\mathbf{h}_1$ and $\mathbf{h}_2$ are both perpendicular to $\mathbf{m}$, it results in (Chapter 5.2 in [SCP05]):

$$\mathbf{m}(s,a,b,c) = r(\mathbf{h}_1(s,\phi_1) \times \mathbf{h}_2(s,\phi_2)). \tag{5.20}$$

Note that r is a only a scaling parameter. $\mathbf{h}_1(s,\phi_1) \times \mathbf{h}_2(s,\phi_2)$ is totally determined by the first surface parameter s and the ϕ value of the chosen lines (given by the measurement).

Letting (Section 5.2 in [SCP05])

$$\mathbf{h}_T(s,\phi_1,\phi_2) = \begin{bmatrix} h_{T1}(s,\phi_1,\phi_2) \\ h_{T2}(s,\phi_1,\phi_2) \\ h_{T3}(s,\phi_1,\phi_2) \end{bmatrix} := \mathbf{h}_1(s,\phi_1) \times \mathbf{h}_2(s,\phi_2), \tag{5.21}$$

(5.11) can be rewritten as (Eq. (98) in [SCP05]):

$$\dot{\mathbf{r}} = \frac{1}{r(h_{T1}h_{T2} - h_{T3}^2)} \begin{bmatrix} h_{T2} & h_{T3} \\ h_{T3} & h_{T1} \end{bmatrix} \begin{bmatrix} B_u \\ B_v \end{bmatrix} = \frac{1}{r}\mathbf{V}\mathbf{B}_{uv}, \tag{5.22}$$

where

$$\mathbf{V} := \frac{1}{h_{T1}h_{T2} - h_{T3}^2} \begin{bmatrix} h_{T2} & h_{T3} \\ h_{T3} & h_{T1} \end{bmatrix}. \tag{5.23}$$

The $\mathbb{R}^2 \to \mathbb{R}^2$ mapping on the tangent plane based on the (more general) $\mathbf{M}$ matrix introduced here in (5.11) is now performed using the $\mathbf{V}$ matrix of Savarese's proposal. The difference between $\mathbf{M}$ and $\mathbf{V}$ is that $\mathbf{M}$ depends on the first order surface parameter s and the second order surface parameters a, b and c, while $\mathbf{V}$ only depends on the first order surface parameter s and two measured angles ϕ_1 and ϕ_2. Additionally, a new scaling parameter r is introduced by utilizing the $\mathbf{V}$ matrix in (5.22). In total, four unknown variables in (5.11) are now reduced to two in (5.22).

It can be proven that the projection of the first order derivative $\dot{\mathbf{r}}_c$ of the observed curve to the image plane can be expressed as (Eq. (37) in [SCP05]):

$$\dot{\mathbf{r}}_c = \mathbf{T}\,\dot{\mathbf{r}} = -\frac{1}{r}\mathbf{T}\begin{bmatrix}\mathbf{V}\mathbf{B}_{uv}\\ 0\end{bmatrix}, \text{ with } \mathbf{T} = \frac{l}{s\langle\hat{\mathbf{d}}, \hat{\mathbf{z}}_c\rangle}\left[\mathbf{I} - \frac{\hat{\mathbf{d}}\hat{\mathbf{z}}_c^T}{\langle\hat{\mathbf{d}}, \hat{\mathbf{z}}_c\rangle}\right]. \tag{5.24}$$

There are only two unknowns r and s left in the (5.24). Using two crossing curves, a new equation with only one unknown variable -s, can be obtained (Eq. (101) in [SCP05]):

$$f(s) = \frac{|\dot{\mathbf{r}}_{c1}|}{|\dot{\mathbf{r}}_{c2}|} - \frac{\begin{bmatrix}\mathbf{B}_{uv1}^T\mathbf{V}^T & 0\end{bmatrix}\mathbf{T}^T\mathbf{T}\begin{bmatrix}\mathbf{V}\mathbf{B}_{uv1}\\ 0\end{bmatrix}}{\begin{bmatrix}\mathbf{B}_{uv2}^T\mathbf{V}^T & 0\end{bmatrix}\mathbf{T}^T\mathbf{T}\begin{bmatrix}\mathbf{V}\mathbf{B}_{uv2}\\ 0\end{bmatrix}}. \tag{5.25}$$

The first order surface parameter s can be determined by solving the target function $f(s) = 0$:

The 3D reconstruction procedure introduced by Savarese can be summarized in 2 steps: (*i*), estimating the first-order derivative $\dot{\mathbf{r}}_{c1}(t)$ and $\dot{\mathbf{r}}_{c2}(t)$ according to the pixel position map; (*ii*), projecting $\dot{\mathbf{r}}_{c1}(t)$ and $\dot{\mathbf{r}}_{c2}(t)$ onto the tangent plane and computing the angle ϕ_1 and ϕ_2; (*iii*), searching for the first-order surface parameter s satisfying $f(s) = 0$.

5.2 Extension to PMD

5.2.1 Full Area Evaluation with $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ Mapping

Savarese's method evaluates the shape and the height of the local object surface by taking not only the point to point correspondence of screen and image, but also the lateral differential correspondence $\dot{\mathbf{r}}_{c1}(t)$, $\dot{\mathbf{r}}_{c2}(t)$ and $\dot{\mathbf{r}}_{s1}(t)$, $\dot{\mathbf{r}}_{s2}(t)$ into account. This lateral correspondence is obtained by observing and evaluating two intersecting lines in a checkerboard by Savarese: Two curves were determined by interpolating the five neighbouring yellow points in Fig. 5.1. The direction vectors of these two curves were numerically computed.

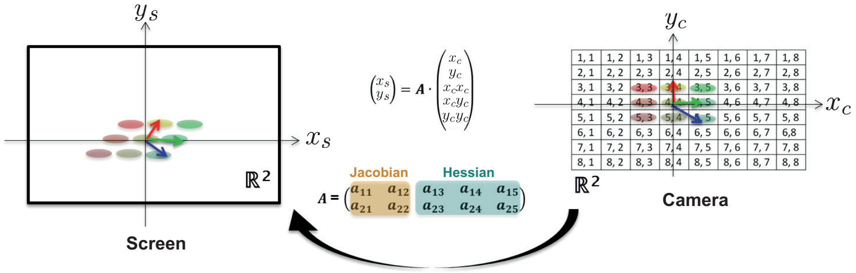
Estimating the local correspondence in this way is rather disadvantageous: 1. the object surface is evaluated only on discrete points in positions that are pre-determined by the pattern on the screen and not on the surface itself; 2. the amount of calculation required is large; 3. the local corresponding differential geometry is determined by only five pre-defined points using arbitrarily chosen curve directions determined by the orientation of the checkerboard pattern on the screen, which is numerically unstable. Indeed, the local lateral differential correspondence is provided in full by every PMD measurement, which is never fully exploited in the conventional PMD evaluation. Utilizing the PMD data to analyze the lateral differential correspondence can avoid the drawbacks mentioned above.

In a PMD measurement, the observed screen points are continuously coded via their coordinates by the phases of the utilized fringe patterns. The lateral correspondence can be estimated in a much more efficient (and stable) way by applying this continuous coding and by applying a $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ mapping evaluation (Fig. 5.4). A local 2 dimensional coordinate system (x_c, y_c) was utilized with its origin locating in the camera pixel to be evaluated. Analogously, a local coordinate system (x_s, y_s) was applied in the screen plane with its origin in the corresponding screen point. The neighbouring pixels in the image $\left[x_{ci}, y_{ci}\right]^T$ with i being a positive integer representing the pixel index and their corresponding screen points $\left[x_{si}, y_{si}\right]^T$ allow for the estimation of the local lateral mapping from the image to the screen plane by using a transformation matrix - "A matrix" with [Lia+19a][Lia+19b][Lia+19c]:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \end{pmatrix}, \tag{5.26}$$

and

$$\begin{bmatrix} x_{si} \\ y_{si} \end{bmatrix} = A \begin{bmatrix} x_{ci} \\ y_{ci} \\ x_{ci}^2 \\ x_{ci}y_{ci} \\ y_{ci}^2 \end{bmatrix}. \tag{5.27}$$



1, 1	1, 2	1, 3	1, 4	1, 5	1, 6	1, 7	1, 8
2, 1	2, 2	2, 3	2, 4	2, 5	2, 6	2, 7	2, 8
3, 1	3, 2	3, 3	3, 4	3, 5	3, 6	3, 7	3, 8
4, 1	4, 2	4, 3	4, 4	4, 5	4, 6	4, 7	4, 8
5, 1	5, 2	5, 3	5, 4	5, 5	5, 6	5, 7	5, 8
6, 1	6, 2	6, 3	6, 4	6, 5	6, 6	6, 7	6,8
7, 1	7, 2	7, 3	7, 4	7, 5	7, 6	7, 7	7, 8
8, 1	8, 2	8, 3	8, 4	8, 5	8, 6	8, 7	8, 8

Fig. 5.4: The lateral correspondence of screen and image is obtained by using a $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ mapping. A local 2 dimensional coordinate system (x_c, y_c) is applied in the image plane with its origin at the pixel point to be evaluated. The corresponding screen point position is given by the PMD phase map. A local 2 dimensional coordinate system (x_s, y_s) is used at this screen point. The local lateral mapping *camera* → *screen* is evaluated by applying the A matrix to given vectors in the image plane.

The transformation matrix A can be determined analytically and efficiently. Choose n^2 neighbouring camera pixels to determine A with $n = 2k + 1$ and k

being an positive integer. (5.27) can be extended to be:

$$P_s = AP_c, \tag{5.28}$$

with $P_c = \begin{bmatrix} x_{c1} & x_{c2} & \cdots & x_{cn^2} \\ y_{c1} & y_{c2} & \cdots & y_{cn^2} \\ x_{c1}^2 & x_{c2}^2 & \cdots & x_{cn^2}^2 \\ x_{c1}y_{c1} & x_{c2}y_{c2} & \cdots & x_{cn^2}y_{cn^2} \\ y_{c1}^2 & y_{c2}^2 & \cdots & y_{cn^2}^2 \end{bmatrix}$ and $P_s = \begin{bmatrix} x_{s1} & x_{s2} & \cdots & x_{sn^2} \\ y_{s1} & y_{s2} & \cdots & y_{sn^2} \end{bmatrix}$.

Since the camera pixels $\begin{bmatrix} x_{ci}, y_{ci} \end{bmatrix}^T$ are located in a 2 dimensional regular square grid, the positions of the neighbouring pixels relative to the central pixel are always the same for every camera pixel to be evaluated. P_c is therefore always the same and known. Transposing (5.28) to $P_s^T = P_c^T A^T$, A^T can then be determined by applying the left pseudo inverse of P_c^T by $A^T = (P_c P_c^T)^{-1} P_c P_s^T$.

Depending on the desired precision, the A matrix is a $m \times 2$ matrix representing the coefficients of the Taylor expansion of the corresponding local mapping on the screen. If $m = 2$, A is a Jacobian matrix. If $m = 5$, A is a Jacobian matrix plus a Hessian matrix. If necessary, further coefficients of the multi-dimensional Taylor expansion can be used. The entire evaluation can also be understood as a local 2 dimensional polynomial fit.

Given the linear approximation of an arbitrary curve passing through the camera pixel $\begin{bmatrix} x_{c0}, y_{c0} \end{bmatrix}$: $\begin{bmatrix} x_c(t) & y_c(t) \end{bmatrix}^T = \begin{bmatrix} x_{c0} & y_{c0} \end{bmatrix}^T + t \begin{bmatrix} \dot{x}_{c0} & \dot{y}_{c0} \end{bmatrix}^T$ parameterized by t, the corresponding curve in the screen plane can be linearly approximated by $\begin{bmatrix} x_{s0} & y_{s0} \end{bmatrix}^T + A \begin{bmatrix} t\dot{x}_{c0} & t\dot{y}_{c0} \end{bmatrix}^T$, or by further coefficients $\begin{bmatrix} x_{s0} & y_{s0} \end{bmatrix}^T + A \begin{bmatrix} t\dot{x}_{c0} & t\dot{y}_{c0} & t^2\dot{x}_{c0}^2 & t^2\dot{x}_{c0}\dot{y}_{c0} & t^2\dot{y}_{c0}^2 \end{bmatrix}^T$. The 3D direction vector of the curve on the camera image plane $\dot{\mathbf{r}}_c$ is obtained by using the camera rotation matrix [6] $\mathbf{R}_c$ with $\dot{\mathbf{r}}_c = \mathbf{R}_c \begin{bmatrix} \dot{x}_{c0} & \dot{y}_{c0} & 0 \end{bmatrix}^T$. Analogously, the direction vector of the curve on the screen plane $\delta\mathbf{r}_s$ is maintained by utilizing the screen

---

[6] $\mathbf{R}_c$ is a 3×3 matrix which is commonly defined as $\mathbf{R}_c = \begin{bmatrix} \hat{\mathbf{x}}_c & \hat{\mathbf{y}}_c & \hat{\mathbf{z}}_c \end{bmatrix}$ where $\hat{\mathbf{x}}_c$, $\hat{\mathbf{y}}_c$, and $\hat{\mathbf{z}}_c$ respectively denote the unit vectors of the x-, y- and z-axis of the camera coordinate system presented in world coordinates. For simplification reasons, the height problem is discussed in this chapter on the tangent plane of the object surface. The principal coordinate system is

rotation matrix [7] $\mathbf{R}_s$ with $\delta\mathbf{r}_s = \mathbf{R}_s \begin{bmatrix} A \begin{bmatrix} \dot{x}_{c0} \\ \dot{y}_{c0} \end{bmatrix} \\ 0 \end{bmatrix}$. According to (5.9), (5.18), (5.21) and (5.23), if $\delta\mathbf{r}_s$ is obtained, $\mathbf{V}$ and $\mathbf{B}_{uv}$ are determined under an assumption of s. The target function (5.25) is therefore a function that depends only on s. Solving the target function $f(s) = 0$, the first order surface parameter s is be determined.

To verify the feasibility and the accuracy of the new approach combining the Savarese evaluation and the local differential mapping of the PMD data, a simulation result is presented and a real measurement of a precise planar mirror is performed.

According to the calibration data of an existing laboratorial PMD setup, the phase of the pattern reflected on an assumed convex surface ($z = -0.25x^2 + 0.001xy - 0.25y^2$) is simulated. Applying the PMD phase and performing the $\mathbb{R}^2 \to \mathbb{R}^2$ mapping, a Savarese evaluation based on the approach introduced above is carried out. The height difference of the evaluated surface with the original surface (the ground truth) is displayed in the left panel in Fig. 5.5. This deviation map displays a full surface evaluation. At each data point, the absolute height is evaluated by using the A matrix evaluated from the simulated PMD phase map using a local 3×3 neighbourhood. The deviation of the evaluated surface from the known surface varies essentially up to 10 nm (Fig. 5.5, right). The deviation is particularly large along a central line in the image (also marked red in the right panel in Fig. 5.5). The reason for this large deviation is correlated with the object - screen geometry (in the case that the principal plane is perpendicular to the screen plane). This will be discussed more thoroughly in Chapter 5.3.4 and Chapter 5.4. In practice, such a geometry can be avoided in advance by rotating

---

therefore applied as the world coordinate in the following discussion. Note that the principal coordinate system is individually defined for every object point. The camera rotation matrix $\mathbf{R}_c$ must be individually calculated at every assumed object surface point by $\begin{bmatrix} \hat{\mathbf{u}} & \hat{\mathbf{v}} & \hat{\mathbf{w}} \end{bmatrix}$.

[7] $\mathbf{R}_s$ is a 3×3 matrix which is commonly defined as $\mathbf{R}_s = \begin{bmatrix} \hat{\mathbf{x}}_s & \hat{\mathbf{y}}_s & \hat{\mathbf{z}}_s \end{bmatrix}$ where $\hat{\mathbf{x}}_s$, $\hat{\mathbf{y}}_s$, and $\hat{\mathbf{z}}_s$ respectively denote the unit vectors of the x-, y- and z-axis of the screen coordinate system presented in the world coordinate. For the same reason as discussed for $\mathbf{R}_c$, the principal coordinate system is applied as the world coordinate in the following discussion. Therefore, $\mathbf{R}_s$ must be individually calculated at every assumed object surface point by $\begin{bmatrix} \hat{\mathbf{u}} & \hat{\mathbf{v}} & \hat{\mathbf{w}} \end{bmatrix} \mathbf{R}_s$.

the object or the entire setup.

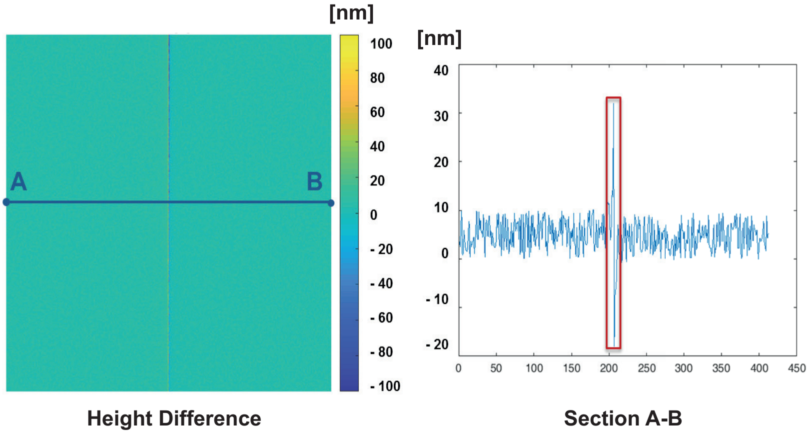


Fig. 5.5: A convex surface ($z = -0.25x^2 + 0.001xy - 0.25y^2$) is evaluated by using a simulated PMD phase map. The height difference of the evaluated surface from the known surface is shown in the left. A section A-B of the difference data is shown in the right.

A planar mirror (Fig. 5.6) with a surface flatness of $\lambda/20$ is under test. The surface points are evaluated by using the lateral differential correspondence determined by 31×31, 51×51, 71×71 and 91×91 local neighbouring camera pixels respectively. The exact physical position and the tilt of the mirror in the coordinate system is unknown in this experiment. The known "ground truth" at this point is only the precise flatness of the mirror. The deviation of the measured surface points against the best-fit plane is therefore considered as the measurement accuracy. The deviations of evaluations using diverse number of neighbouring camera pixels are displayed in (a), (b), (c) and (d) in Fig. 5.7. The height of every surface point is evaluated absolutely only according to the local lateral phase information from one camera using the new method introduced above. Compared to the simulation in Fig. 5.5, in which only 3×3 pixels are used, more local neighbouring pixels are needed in the real measurement to suppress the noise (the differential approach always increases the noise in real measurements). As to be expected, the more local pixels are used, the smoother the evaluated surfaces appears (Fig. 5.7). The height analysis using the local differential correspondence exhibits stable surface points with an accuracy of up to ±500 μm ((d) in Fig. 5.7). It should be noted that the other monocular evaluation [Kam04b], which employs the surface integrability theorem, is difficult to implement in practice, and that the accuracy of an other PMD height evaluation approach applying multiple cameras, as shown in [KHL04a], is about ±70 μm. An accuracy of ±500 μm is sufficient for a preliminary assumption of the object surface.

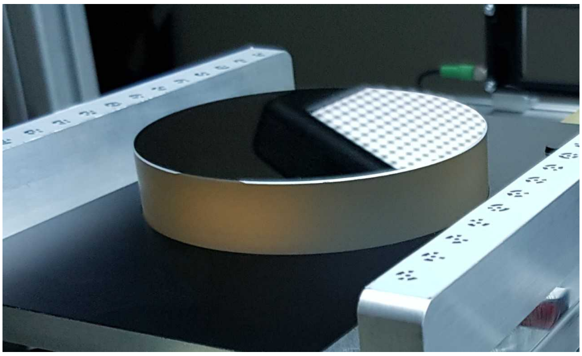


Fig. 5.6: A precise planar mirror with a surface flatness of $\lambda/20$ is under test.

There are several advantages of utilizing the $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ mapping to estimate the lateral correspondence. Firstly, the A matrix can be quickly and robustly evaluated by applying the pseudo inverse of P_c^T based on the camera pixel grid. Secondly, in contrast to Savarese's original approach, A completely represents the lateral correspondence by mapping the lateral vectors in all directions. The entire lateral correspondence offers many advantages, which will be discussed in Chapter 5.3.1. Thirdly, the lateral correspondence is not estimated from discrete points, but from the entire continuous local context information, which suppresses noise and provides a more stable measurement.

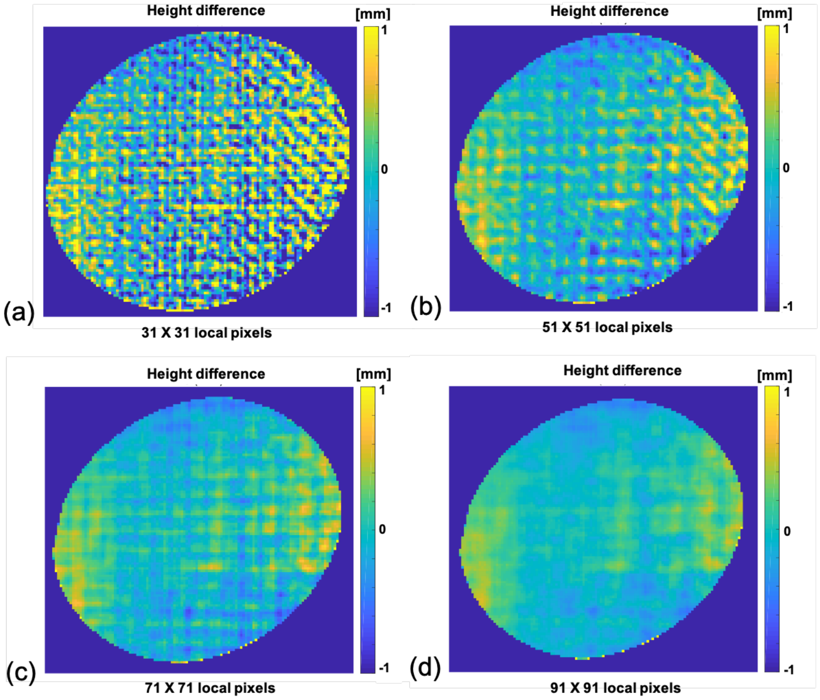


Fig. 5.7: A precise planar mirror is measured using the new monocular approach for an absolute height measurement combining the Savarese method and the lateral differential correspondence of PMD. The mirror surface is evaluated by using 31×31, 51×51, 71×71 and 91×91 local neighbouring camera pixels of the PMD phase map in every measurement point, resulting in a dense height map (in contrast to Savarese's original method). The deviation of the measured surface against the best-fit plane is shown in (a), (b) (c) and (d) respectively.

5.3 Geometrical Properties and Bad Conditions

5.3.1 Surface Properties

Note that (5.11) is well defined only under the condition $d \neq 0$, which guarantees that the mapping matrix $\mathbf{M}$ is non-singular and the smooth and unique mapping around the neighborhood of $\mathbf{r}_0$ exists [SCP05]. The condition $d \neq 0$ can be written as

$$\frac{1}{s} + \frac{1}{\mid r_{s0} \mid} = \frac{a}{\cos\theta} + b\cos\theta \pm \sqrt{(\frac{a}{\cos\theta} - b\cos\theta)^2 + 4c^2} \tag{5.29}$$

by considering (5.14) and (5.10). (5.29) is equal to (3.14), with (5.29) being the interpretation in mathematics and (3.14) being the interpretation in imaging optics. Moreover, the condition number of $\mathbf{M}$ as a measure of how close $\mathbf{M}$ is to being singular, should be small. Note that in some cases a non-singular $\mathbf{M}$ may have a large condition number. Large condition number, also called ill-conditioned, indicates that the output of the function (5.15) is sensitive to small changes or errors in the input signal - the numerical stability is low. An example of a non-continuous mapping ($\mathbf{M}$ is singular) on the object surface is shown in the Fig. 4 in [SCP05].

The continuity property of the mapped pattern can be explained in terms of geometrical optics. If the specular surface (which in deflectometry has to be considered as part of the imaging system) is a convex surface, it acts as a divergent lens on the chief rays, therefore, its determinant never equals zero[8]. This can also be proven mathematically (Proposition 10 in [SCP05]): A convex surface has $a < 0, b < 0$ and $c^2 < ab$. J_u and J_v are both positive. The angle of incidence θ is between 0 and $\frac{\pi}{2}$. Consequently $\frac{J_u}{2\cos\theta} - a$ and $\frac{J_v}{2\cos\theta} - b$ are both positive and greater than $|a|$ and $|b|$ respectively. Therefore $(\frac{J_u}{2\cos\theta} - a)(\frac{J_v}{2\cos\theta} - b)$ is always greater than c^2 and thus the determinant is always positive.

---

[8] Note that $d = (J_u - 2a\cos\theta)(J_v - 2b\cos\theta) - 4c^2\cos^2\theta$ is not only determined by the surface shape a, b and c, but also by the geometry of the setup and the surface position: J_u, J_v and θ. But in the case of a convex mirror, d is always positive.

If the specular surface is a concave surface or a saddle surface, it acts as a (partly) convergent lens in the optical path between the camera and the screen. Depending on the object and image positions and also the surface shape, the rays of light can be converged in varying degrees. The mathematical measure of the optical convergence[9], resp. local magnification scale of the imaging process, is the determinant of the mapping d. In both cases (a concave specular surface and a saddle specular surface), the case $d = 0$ is not excluded. Examples are shown in Fig. 5.8. In Fig. 5.8, the camera view rays of 5×5 camera pixels are traced backwards according to a given specular surface in a given position to the scene. The intersection points of the view rays with the scene (the yellow area) are presented as red points. In Fig. 5.8 (a), (b) and (c), three saddle surfaces with decreasing d are simulated, and in Fig. 5.8 (e), (f) and (g), three concave surfaces also with decreasing d are simulated. In both cases, the imaged intersection points are converging to a line with d approaching to zero.

In a more general way: d or the convergence does not only depend on the (local) geometry of the surface, but also on the geometry of the apparatus; $d = 0$ means, that the neighborhood of $\mathbf{r}_0$ locally images the rays originating from a 2d areal patch (neighbourhood) on the screen onto a 1d line in the camera and vice versa. Since a convex paraboloid mirror surface is diverging, therefore, d is always positive for all convex surfaces.

5.3.2 Properties of the Chosen Scene Lines

Note that the $\mathbf{h}_T$ vector ((5.21)) is parallel to the normal vector of the plane spanned by $\mathbf{h}_1$ and $\mathbf{h}_2$. $\mathbf{h}_1$ and $\mathbf{h}_2$ are both determined by the surface parameter in $0th$ order (s), and also by the values of $\tan\phi_1$ and $\tan\phi_2$ of the chosen scene lines respectively. If the chosen scene line maps onto a curve in the tangent plane introducing $\phi = \frac{\pi}{2} + n\pi$, with $n = 0, \pm1, \pm2, ...$, $\tan\phi$ is undefined. For these scene lines, s can not be computed. If ϕ is in the vicinity of $\frac{\pi}{2} + n\pi$, the error by locating s will be significantly magnified. Since the tangent value is not a surface

---

[9] Not the convergent power of the lens/the surface shape (expressed by the focal length), but how strong the image is contracted resp. expanded by a given lens, in a given object and image position (expressed by the magnification scale).

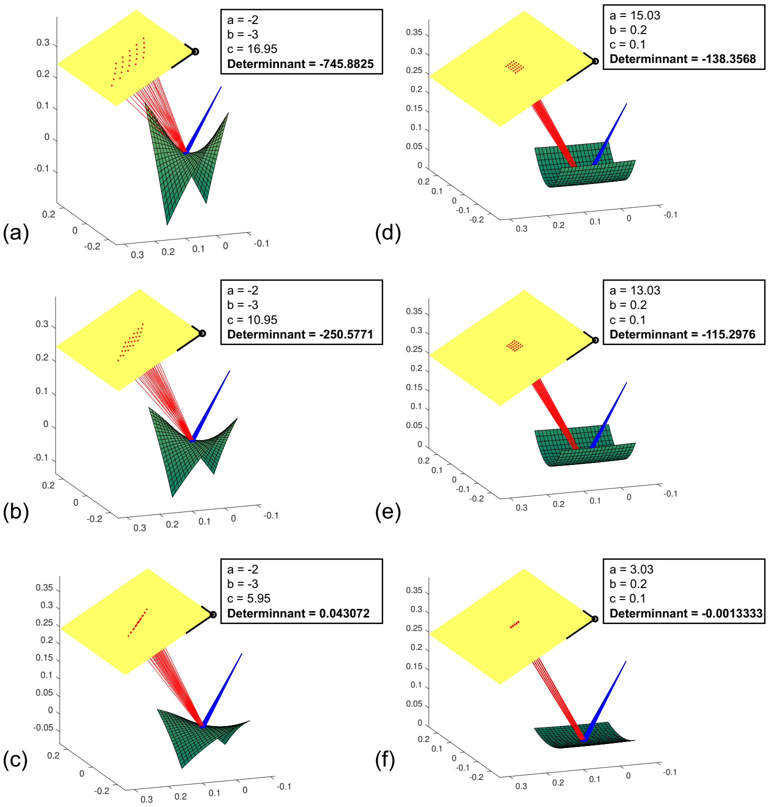


Fig. 5.8: Examples of concave surfaces and saddle surfaces. The view rays of 5×5 camera pixels are traced backwards. The red lines are incident rays from the scene and blue lines are reflection rays to the camera (landing on the 5×5 camera pixels). The scene plane is marked yellow and the specular surface is shown in green: (a), (b) and (c) are saddle surfaces with $\Delta = -745.8825\frac{1}{m^2}$, $\Delta = -250.5771\frac{1}{m^2}$ and $\Delta = 0.043072\frac{1}{m^2}$ respectively; (d), (e) and (f) are concave surfaces with $\Delta = -138.3568\frac{1}{m^2}$, $\Delta = -115.2976\frac{1}{m^2}$ and $\Delta = -0.0013333\frac{1}{m^2}$ respectively;

property, but rather strongly depends on the chosen scene lines, the error induced by $\tan\phi$ can partially be corrected by rotating the chosen scene line. This is not possible in the case of Savarese's original method where the scene lines are predefined at capture time by the edges of the utilized checkerboard pattern. In the case of PMD, however, any direction can be chosen at evaluation time, as the full $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ correspondence is available from the phase map provided by PMD.

Besides the condition of $\phi \neq \frac{\pi}{2} + n\pi$, the two scene lines must be chosen so that $\mathbf{h}_1$ and $\mathbf{h}_2$ are linearly independent, which means $Rank[\mathbf{h}_1; \mathbf{h}_2] \neq 1$. In the linearly dependent case, $\mathbf{h}_1$ and $\mathbf{h}_2$ do not span any plane. Therefore (5.20) is not valid and consequently $\mathbf{V}$ (through (5.23)) is not computable. This leads to a poor accuracy in estimating s, if $\mathbf{h}_1$ and $\mathbf{h}_2$ are close to colinear. Note that $\mathbf{V}$ is theoretically independent of the chosen scene lines and totally determined by the surface parameters and screen position. To simplify the nonlinear problem described in (5.11), the "values of $\tan\phi$" of two scene lines (measured values) are introduced as auxiliary variables to evaluate $\mathbf{V}$. The bad condition introduced by the linear dependence of $\mathbf{h}_1$ and $\mathbf{h}_2$ can be solved by rotating the scene lines.

Both of the problems discussed above are induced by the particular chosen scene lines and can be solved by just rotating the scene lines. An example is shown in Fig. 5.9. The PMD phase data of a spherical surface is simulated (according to the calibration data of the laboratorial PMD set up: screen position, camera position, camera focal length...). The error of estimating the surface points using the method introduced in [SCP05] and in Chapter 5.2 is shown in Fig. 5.9 (a). Fig. 5.9 (b) displays the surface area with large tangent values and Fig. 5.9 (c) displays the mask of the surface area having two linear dependent vectors, $\mathbf{h}_1$ and $\mathbf{h}_2$. The surface points are evaluated again with the same method but with the chosen scene lines rotated in the marked out areas. The error of the second evaluation is displayed in Fig. 5.9 (d). The error introduced by the bad condition of the chosen scene lines (in both of mask in (b) and (c)) are significantly corrected.

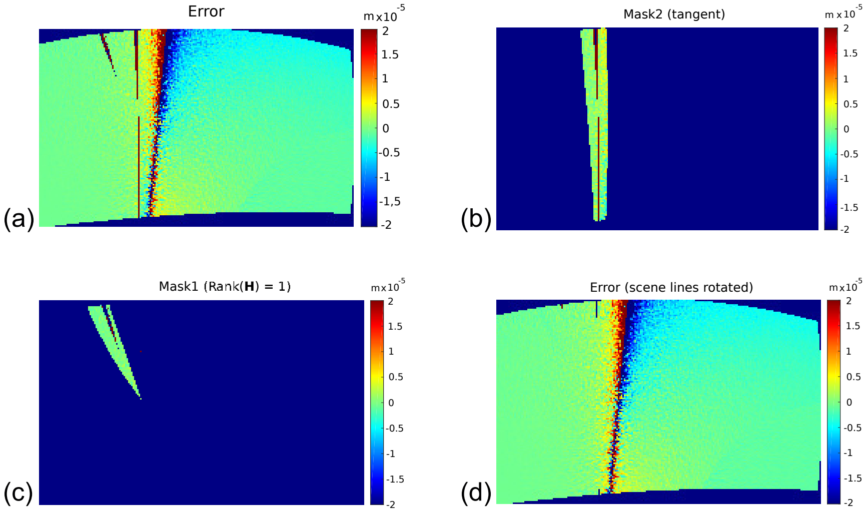


Fig. 5.9: Tangent and linear dependence correction in simulated data: (a) Error by the reconstruction of a spherical surface; (b) The mask of the area with large tangent values; (c) The mask of the area where $\mathbf{h}_1$ and $\mathbf{h}_2$ are highly linear dependent; (d) Error by the reconstruction of a spherical surface after rotating the scene lines accordingly in the marked out areas.

5.3.3 Ghost Solutions

The surface point is evaluated by solving the non-linear equation $f(s) = 0$ ((5.25)). However, the empirical analysis in Matlab shows that, the target function $f(s)$ vanishes sometimes not only at the actual surface, but also at some "ghost" solutions. This problem is also discussed in [SCP05]: the ghost solution is called degenerate geometrical configuration, which means all the scene lines have the same ratio of B_u to B_v ($\forall i \neq j, \frac{B_{ui}}{B_{vi}} = \frac{B_{uj}}{B_{vj}}$). The degenerate geometrical configuration occurs when the "ghost surface point" ($\mathbf{r}_g$) belongs to the plane spanned by the scene lines, i.e. the "ghost solution" *lies within the plane of the screen*. This is an obvious possibility, as only first order parameters have been used to determine s and a "direct" observation of the screen without any intermediate reflection at all is obviously a possible solution as well. In the simulations performed above, the scene is located in a way that the camera rays never intersect the scene plane (even if they intersect the scene plane, the intersection points only arise in the far field out of the numerical search range.). The degenerate geometrical configuration ($\forall i \neq j, \frac{B_{ui}}{B_{vi}} = \frac{B_{uj}}{B_{vj}}$) mentioned in [SCP05] never arises in the simulation analysis considered here. But still, multiple minima show up in the profiles of some target functions that have not been considered or discussed in [SCP05].

Examples of profiles of log $\mid f(s) \mid$ are displayed in Fig. 5.10. Note that $f(s)$ is the target function and its absolute value varies in the search range ($s \in [0.1m, 0.4m]$) from $1e-4$ to $1e-12$. To make the local minima visible, the absolute value of the target function is demonstrated in logarithmic scale. Moreover, finding the solution of $log \mid f(s) \mid$ is numerically advantageous because the derivative of $log \mid f(s) \mid$ is better scaled for small values of $f(s)$ than the derivative of $\mid f(s) \mid$. As in Fig. 5.10 (a), the ghost solution can even approach the actual solution, which contradicts the statement of degenerate geometrical configuration in [SCP05]. Even worse, in some cases the ghost solution has a lower target function than the actual solution (see case (d) in Fig. 5.10)! In general, the ghost solutions are rather uncommon. This is problematic for the reconstruction because ghost solutions can not be ruled out analytically. Further work is needed in order to derive sufficient conditions to reject ghost solutions. A possible solution can be constraining the variation of the curvature of the evaluated surface.

5.3.4 Perpendicular Configuration

Independently of the multiple solutions, outliers of the reconstruction errors are observed frequently (e.g. the middle region of the height map in Fig. 5.5). In order to validate the source of this kind of errors, specular reflections of different polynomial surfaces in various setup geometries (diverse scene positions and camera positions) are simulated. The simulated apparatus and polynomial surface are displayed in the left panel of Fig. 5.11. The polynomial surfaces are reconstructed by locating the local minimum of $f(s)$. The error of the

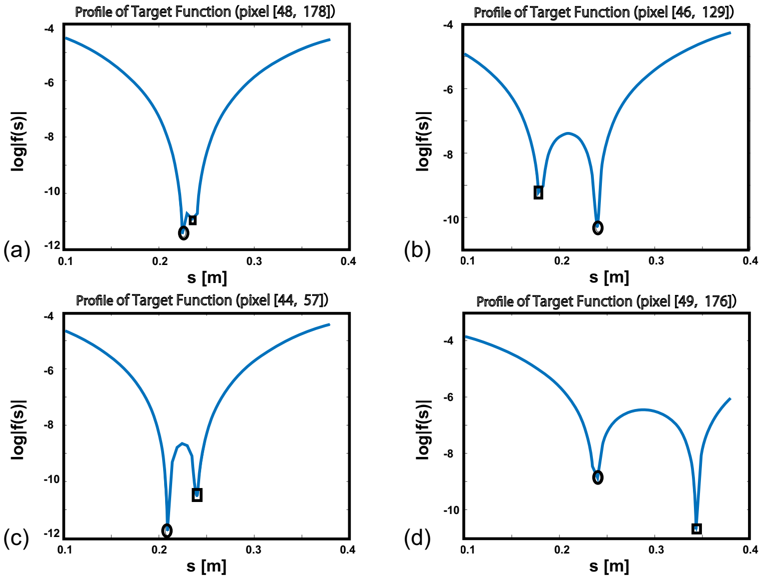


Fig. 5.10: Examples of profiles of log | $f(s)$ |. Actual solutions are marked with ellipses and ghost solutions are marked with rectangles. (a) The profile of log | $f(s)$ | of pixel [48, 178]; (b) The profile of log | $f(s)$ | of pixel [46, 129]; (c) The profile of log | $f(s)$ | of pixel [44, 57]; (d) The profile of log | $f(s)$ | of pixel [49, 176];

configuration	surface property	screen and camera rotation
config.1	saddle surface $-0.5x^2 - 0.001xy + 0.5y^2$	screen rotated by 0° camera rotated by 0°
config.2	saddle surface $-0.5x^2 - 0.001xy + 0.5y^2$	screen rotated by 2° camera rotated by 0°
config.3	saddle surface $-0.5x^2 - 0.001xy + 0.5y^2$	screen rotated by 4° camera rotated by 0°
config.4	concave surface $0.05x^2 + 0.001xy + 0.05y^2$	screen rotated by 0° camera rotated by 0°
config.5	concave surface $0.05x^2 + 0.001xy + 0.05y^2$	screen rotated by 0° camera rotated by 0.75°
config.6	convex surface $-0.25x^2 + 0.001xy - 0.25y^2$	screen rotated by -2° camera rotated by 2°

Tab. 5.1: The surface properties and the screen and camera positions of the simulation displayed in Fig. 5.11. Rotation = 0° means the camera image plane or the screen plane is perpendicular to the yz-plane.

reconstructed surface is shown in the right panel of Fig. 5.11. The surface and apparatus properties are listed in Tab. 5.1 respectively. The evaluation of the same saddle surface (Config.1, Config.2 and Config.3) with the screen rotated by 0°, 2° and 4° indicates that the error region of the reconstructed surface rotates with the increase of the screen angle. The evaluation of a concave surface is shown in Config.4, Config.5 in Fig. 5.11. Rotating the camera slightly by 0.75°, the error region is shifted to the left hand site as well. Rotating the screen and the camera at the same time (Config.6 in Fig. 5.11), the error region stays in the same position on the specular surface. Obviously, this error is the result of a certain combination of the apparatus geometry and the object surface. The empirical analysis shows that the outliers always arise in the *perpendicular configuration*, where the principle plane defined in Chapter 5.1 (spanned by the incident and reflection ray) is perpendicular to the scene plane.

This conjecture can be further confirmed by observing the profile of $\mid f(s) \mid$ while searching the numerical minimum. Fig. 5.12 (a) displays the error of the evaluated surface in full-area. The left area in the picture shows very large errors.

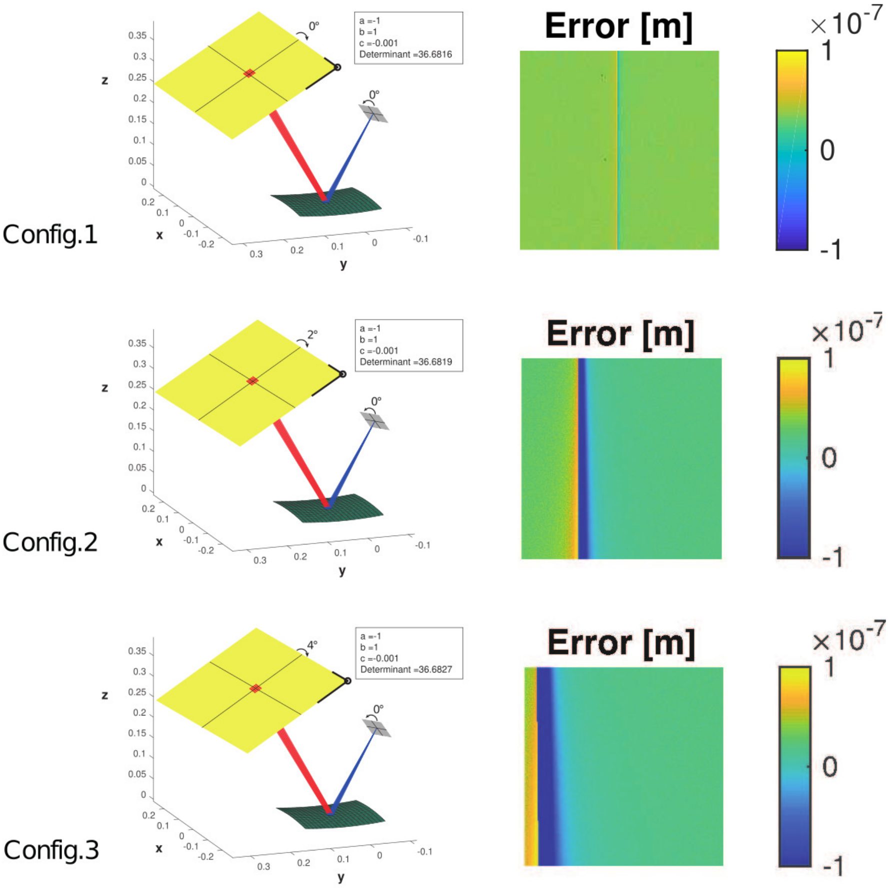


Fig. 5.11: Outliers in various geometrical configurations for different surface shapes. The first column displays the position of the simulated apparatus: screen (yellow), the specular surface (green), the incident ray (red) and the reflection ray (blue). The apparatus and the specular surface property are listed in Tab. 5.1. The second column displays the error of the reconstructed surface respectively.

It is at the same time the surface area having a perpendicular configuration (known from the simulation: the surface points lying on the curve where the error

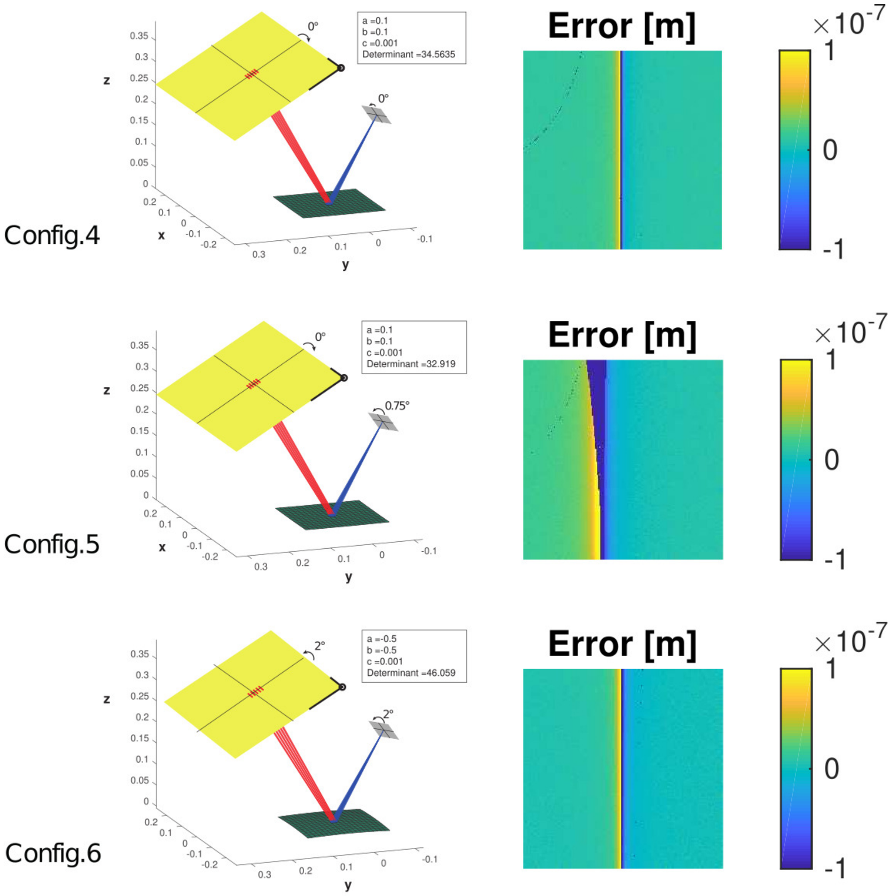


Fig. 5.11: Outliers in various geometrical configurations for different surface shapes. The first column displays the position of the simulated apparatus: screen (yellow), the specular surface (green), the incident ray (red) and the reflection ray (blue). The apparatus and the specular surface property are listed in Tab. 5.1. The second column displays the error of the reconstructed surface respectively.

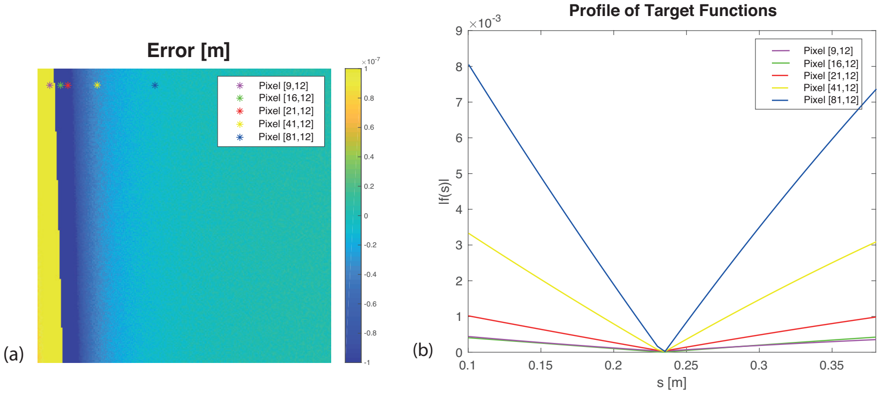


Fig. 5.12: Examples of profiles of $\mid f(s) \mid$. (a) Reconstruction error in full-area; (b) the profiles of $\mid f(s) \mid$ of pixel $[9,12]$, $[16,12]$, $[21,12]$, $[41,12]$ and $[81,12]$.

jumps from positive (yellow) to negative (blue) lead exactly to a perpendicular configuration). Picking out five pixel points as marked in Fig. 5.12 (a), the profiles of $\mid f(s) \mid$ along varying s is shown in (b) in Fig. 5.12 respectively. The further the points are from the area leading to the perpendicular configuration, the steeper $\mid f(s) \mid$ varies, and vice versa. The points close to the perpendicular configuration area (Pixel $[9,12]$, $[16,12]$ and $[21,12]$ in Fig. 5.12) have a rather flat profile. The sensitivity when searching the numerical minimum of $\mid f(s) \mid$ is therefore relatively weak. Large errors appear even in the case of low noise.

Empirical analysis with Matlab shows that: (1) The outliers always arise in the perpendicular configuration, where the principle plane is perpendicular to the scene plane; (2) In the perpendicular configuration, the slopes of f(s) are distinctly low. The lower is the slope of f(s), the worse is the accuracy in estimating s (an example is shown in Fig. 8); (3) The reconstruction errors in the perpendicular configuration are independent of the chosen scene lines. This error in estimating s can not be corrected by rotating the scene lines.

5.4 Monocular Height Estimation without Particular Scene Lines

The method introduced in [SCP05] and [SCP04] applied the differential correspondence between the object plane (the screen) and the image plane (the camera) to estimate the position of the optical imaging surface. The relation between the differential geometries ((5.11)) is generated by utilizing the chain rule on a equation group (Eq. 5.3) created by considering the law of reflection and the surface representation in Monge form (Eq. 5.1). This relation associating with the surface position and surface shape is mathematically interpreted by the the mapping matrix $\mathbf{M}$ on the tangent plane. [SCP05] and [SCP04] applied the tangent value of two arbitrary curves to estimate $\mathbf{M}$, which is necessary since there are only differential correspondences in two directions available, but disadvantageous since the two curves are predefined and have bad mathematical properties (like $tan\frac{\pi}{2}$ is not defined). The PMD phase data provide a full $\mathbb{R}^2 \to \mathbb{R}^2$ mapping between the screen and the image. Differential correspondence in all directions are described in this 2D mapping. Bad mathematical conditions can be avoided by changing the chosen angle ϕ. However, the height problem can also be solved *without* introducing any arbitrary curves by considering the entire system in local 2D patches mapped or transformed from plane to plane.

Denote a patch on the camera image plane by $\mathbf{O}_c$ with

$$\mathbf{O}_c = \begin{bmatrix} x_{c1} - x_c^* & x_{c2} - x_c^* & x_{c3} - x_c^* & ... & x_{ci} - x_c^* \\ y_{c1} - y_c^* & y_{c2} - y_c^* & y_{c3} - y_c^* & ... & y_{ci} - y_c^* \\ & & \mathbf{0} & & \end{bmatrix}, \tag{5.30}$$

where $\begin{bmatrix} x_c^* & y_c^* \end{bmatrix}^T$ denotes the pixel to be measured and $\begin{bmatrix} x_{ci} & y_{ci} \end{bmatrix}^T$ denotes the ith pixel belonging to the patch. The patch can be rotated to be parallel to the camera plane by left multiplying the patch with the camera rotation matrix $\mathbf{R}_c\mathbf{O}_c$ (s. Fig. 5.13). At this point, only the rotation is considered and the translation is not included since only the lateral deformation of the patch is important and the position of patch center $\begin{bmatrix} x_c^* & y_c^* \end{bmatrix}^T$ in the 3D space is not relevant. The rotated patch $\mathbf{R}_c\mathbf{O}_c$ can then be projected onto the oblique tangent plane under a perspective projection with respect to the camera perspective as shown in Fig. 5.13. The projection onto the oblique tangent plane can be approximately represented as (derivation see Appendix. A.4):

$$\mathbf{O} = \frac{s}{l}\langle\hat{\mathbf{d}}, \hat{\mathbf{z}}_c\rangle(I - \frac{\hat{\mathbf{d}}\hat{\mathbf{w}}^T}{\langle\hat{\mathbf{d}}, \hat{\mathbf{w}}\rangle})\mathbf{R}_c\mathbf{O}_c, \tag{5.31}$$

with s denoting the distance between the projection center and the object point and l denoting the distance between the projection center and the camera plane. $\hat{\mathbf{z}}_c$ denotes the camera axis and $\hat{\mathbf{w}}$ denotes the normal vector of the tangent plane.

Note that (5.31) is not a homogeneous transformation. (5.31) is the transformation of lateral vectors on the image plane onto the tangent plane. Translation is not considered since the starting point of the vectors is not important and only the direction and the norm of the vectors before and after the transformation are relevant. Also note that this is not a standard perspective projection since (5.31) projects a patch on the image plane onto the tangent plane, *neither* of which needs to be perpendicular to the reflection ray. The derivation in detail is shown in Appendix. A.4. $\mathbf{O}$ is the extension of $\dot{\mathbf{r}}$ denoted in Chapter 5.1 while $\dot{\mathbf{r}}$ contains only the lateral relation in one direction, but $\mathbf{O}$ contains the lateral relation in all directions without any preference. Note that the third row of $\mathbf{O}$ is $\mathbf{0}$ since the patch $\mathbf{O}$ is lying on the $\hat{\mathbf{u}}\hat{\mathbf{v}}$ plane.

The patch can also be transformed onto the tangent plane with respect to the screen. The patch $\mathbf{O}_c$ related to the screen plane is given by a $\mathbb{R}^3 \rightarrow \mathbb{R}^3$ mapping matrix $\tilde{\mathbf{A}}$ with $\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}$, where $\mathbf{A}$ is defined in (5.26). As shown in Chapter 5.2, The 3D representation of the screen patch (only considering its orientation but not its translation) is obtained by $\mathbf{R}_s\tilde{\mathbf{A}}\mathbf{O}_c$. The screen patch is mapped onto the tangent plane in two steps. Firstly, the screen patch is projected onto a plane perpendicular to the incident ray (the red plane in Fig. 5.14) by using the

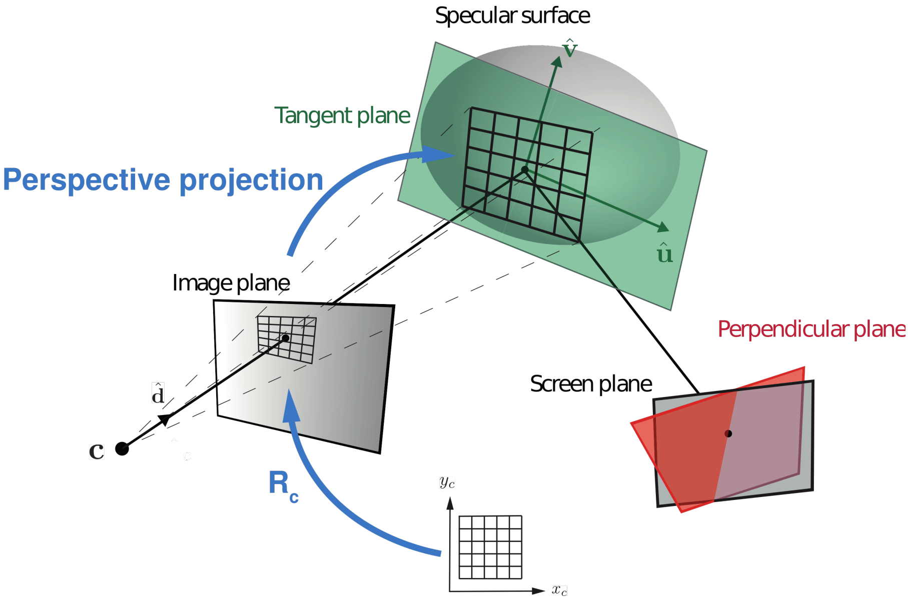


Fig. 5.13: The projection of a given patch onto the tangent plane in terms of the camera perspective projection. A 2D patch is rotated and presented in 3D according to the camera perspective $\mathbf{R}_c$. The 3D patch is projected onto the tangent plane of the object surface by applying (5.31).

projection matrix:

$$\mathbf{P} = -(\mathbf{I} - \mathbf{x}_i \mathbf{x}_i^T). \tag{5.32}$$

Expressing the incident ray in terms of the reflection angle θ with $\mathbf{x}_i = \begin{bmatrix} -\sin\theta & 0 & -\cos\theta \end{bmatrix}^T$, it results:

$$\mathbf{P} = \begin{bmatrix} -\cos^2\theta & 0 & \cos\theta\sin\theta \\ 0 & -1 & 0 \\ \cos\theta\sin\theta & 0 & -\sin^2\theta \end{bmatrix}, \tag{5.33}$$

and the patch on the perpendicular plane is then computed by $\mathbf{P}\mathbf{R}_s\tilde{\mathbf{A}}\mathbf{O}_c$. Scaling the patch on the perpendicular plane with the inverse of the length of the incident ray and denoting it as $\mathbf{O}_s$ with:

$$\mathbf{O}_s = \frac{1}{\mid \mathbf{r}_{s0} \mid}\mathbf{P}\mathbf{R}_s\tilde{\mathbf{A}}\mathbf{O}_c, \tag{5.34}$$

$\mathbf{O}_s$ results in the extension of $\mathbf{B}$ defined in (5.5) and (5.9). $\mathbf{B}$ specifies the (scaled) projection of one curve direction onto the perpendicular plane wheres $\mathbf{O}_s$ specifies the entire (scaled) lateral patch on the red plane in Fig. 5.14. The (second) projection of $\mathbf{O}_s$ onto the tangent plane is computed by simply letting the third row of $\mathbf{O}_s$ to be $\mathbf{0}$.

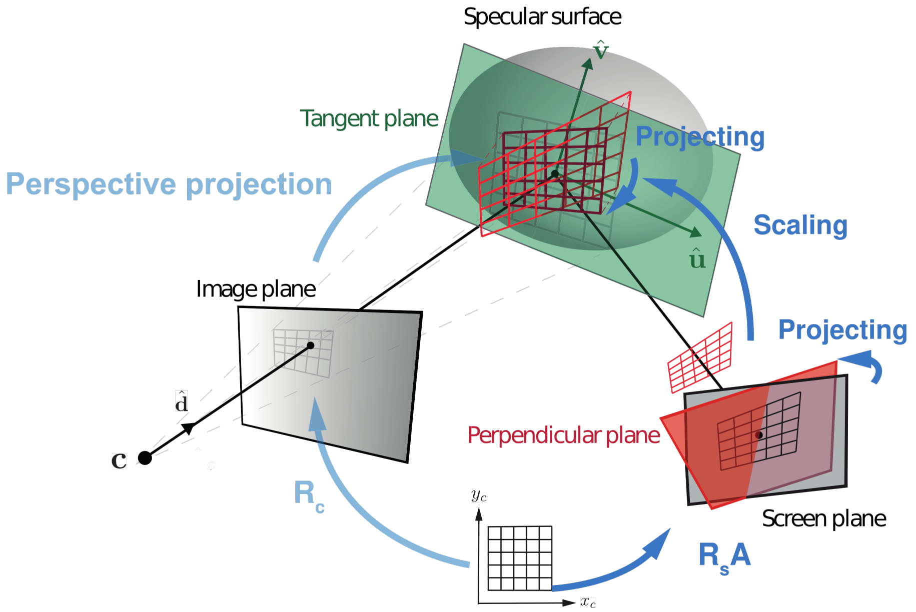


Fig. 5.14: The projection of a given patch onto the tangent plane in terms of the screen. A 2D patch related to the screen is given by the mapping matrix $\mathbf{A}$ (Chapter 5.2). The screen patch in 3D is obtained by using the screen rotation matrix $\mathbf{R}_s$. The 3D patch is then projected onto a plane perpendicular to the incident ray and then projected onto the tangent plane.

As shown in (5.11), the projection of $\mathbf{B}$ and $\dot{\mathbf{r}}$ on the $\hat{\mathbf{u}}\hat{\mathbf{v}}$ plane is mapped with $\mathbf{M}$. The mapping can be extended from one specific curve direction to the entire lateral area. Denoting the projection of $\mathbf{O}$ and $\mathbf{O}_s$ on the $\hat{\mathbf{u}}\hat{\mathbf{v}}$ plane as $\mathbf{O}_{uv}$ and $\mathbf{O}_{s,uv}$, the full area lateral mapping is expressed as:

$$\mathbf{O}_{uv} = \mathbf{M}\mathbf{O}_{s,uv}. \tag{5.35}$$

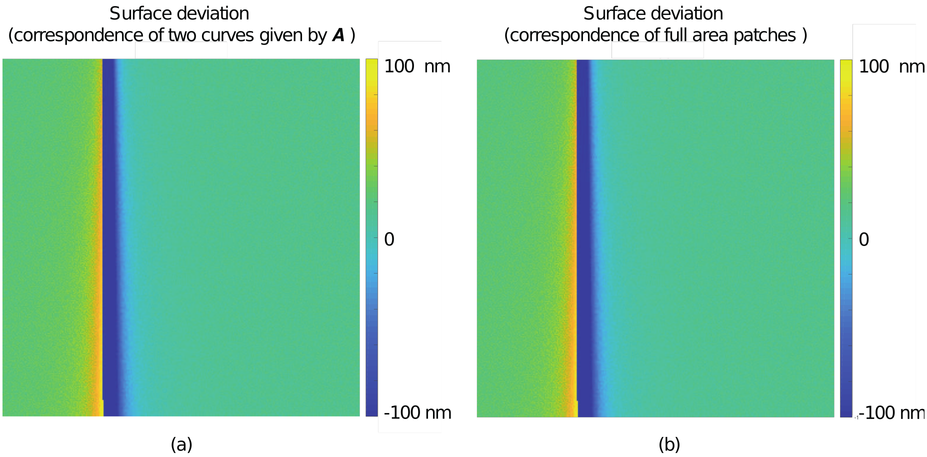
The mapping matrix $\mathbf{M}$ can be computed by $\mathbf{O}_{uv}\mathbf{O}_{s,uv}^{-1}$, which is also the polynomial fit of $(\frac{1}{|\mathbf{r}_{s0}|}\mathbf{P}\mathbf{R}_s\tilde{\mathbf{A}})_{uv}$ (from (5.34)) to $(\frac{s}{l}\langle\hat{\mathbf{d}},\hat{\mathbf{z}}_c\rangle(I - \frac{\hat{\mathbf{d}}\hat{\mathbf{w}}^T}{\langle-\hat{\mathbf{d}},\hat{\mathbf{w}}\rangle})\mathbf{R}_c)_{uv}$ (from (5.31)). Note that there is only one unknown variable s in $\mathbf{O}$ and $\mathbf{O}_s$. $\mathbf{M}$ is therefore a function only of s. As shown in (5.11), $\mathbf{M}$ muss be a symmetric matrix. The height of the object surface can therefore be obtained by s satisfying resp. imposing the condition that $\mathbf{M}$ is a symmetric matrix. However, whether a matrix is symmetric is a binary condition, which is disadvantageous in numerical computation. The symmetric criterion can be modified to be

$$\mathbf{U}^T\mathbf{V} \rightarrow \mathbf{I}, \tag{5.36}$$

with $\mathbf{U}$ and $\mathbf{V}$ being the left- and right-singular vectors of $\mathbf{M}$. If $\mathbf{M}$ is a symmetric matrix, it gives $\mathbf{M} - \frac{1}{2}(\mathbf{M} + \mathbf{M}^T) = 0$, which can also be written as $\mathbf{U}\Sigma\mathbf{V}^T - \frac{1}{2}(\mathbf{U}\Sigma\mathbf{V}^T + \mathbf{V}\Sigma\mathbf{U}^T) = 0$. Left multiplying the function with $\mathbf{U}^T$ and right multiplying the function with $\mathbf{V}$, it results in $\frac{1}{2}\Sigma - \frac{1}{2}\mathbf{U}^T\mathbf{V}\Sigma\mathbf{U}^T\mathbf{V} \rightarrow \mathbf{0}$. Therefore, if $\mathbf{M}$ is a symmetric matrix, it follows $\mathbf{U}^T\mathbf{V} \rightarrow \mathbf{I}$.

For comparison, the height of a simulated surface is computed using two specific curves given by the $\mathbf{A}$ matrix (Chapter 5.2) and computed without any preferential directions as introduced in this section. The evaluated heights are subtracted by the simulated surface and the deviations are shown in Fig. 5.15. The results show that there are almost no difference between these two methods. The outliers induced by the perpendicular configuration remain the same.

The open question of the perpendicular configuration discussed in Chapter 5.3.4 can now be explained by the system symmetry: Since the object height is computed by s satisfying the symmetry condition of $\mathbf{M}$, therefore, it is



(a)

(b)

Fig. 5.15: The height of a simulated surface is evaluated with two arbitrary curves obtained by **A** and evaluated without preferential directions. The height deviations from the simulated surfaces are displayed in (a) and (b) respectively.

expected that the computed **M** for the wrong object position is asymmetric. The “perpendicular configuration” where the screen plane is perpendicular to the principal plane, leads to system symmetry about the $\hat{\mathbf{u}}$ axis. The expected asymmetry of **M** while s does not equal the correct surface height is therefore getting weak or is lost completely. This is also demonstrated in Fig. 5.12. The maximal error (where the positive error switches to a negative error) appears exactly at the region where the perpendicular configuration is present(known from the simulation). The nearer the pixel locates to the area having a perpendicular configuration, the flatter the profile of the target function becomes. The sensitivity of locating the surface height is therefore getting weak.

Chapter 6

Data Fusion from Different Signal Sources

Chapter 5 presented how the information from a single camera position can be maximally exploited. In this chapter, the gathering of information will be extended to multi-camera positions, multi-object positions and even multiple measurement systems.

Multi position measurements are advantageous and competitive when measuring a complex object with large local curvatues which introduce a very large angular dynamic range and optical divergence. The pattern on the screen is, in this case, only reflected by a part of the object surface under test. For a full-field measurement (measuring the entire object surface at once), a much larger screen would be needed. However, a large screen introduces other problems, as it limits severely the available installation space for the camera and puts high demands on system calibration. This results in a slant viewing angle of the camera and a long working distance between the camera and the object.

This problem can be solved by using multi-sensor positions[1]. [Ole+14] also applied multi-sensor positions to solve the measuring field problem of astronomical mirrors. The 3D point cluster is firstly evaluated individually for every camera position. The 3D point clusters from different camera positions are then merged into a large point cluster. This method is straightforward and easy to be implemented. However, due to unavoidable calibration inaccuracies, edges are always present in the transition areas where the data acquired by one sensor is transitioning to the data acquired by another sensor. Therefore, it is advisable to merge the data in an earlier stage of the evaluation.

---

[1] Multi-camera positions are conventionally applied in PMD [KHL04a]. The purpose of using multiple cameras is primarily to solve the height problem introduced in Chapter 2 and Chapter 5. In this chapter, however, the goal is to utilize additional information from multiple cameras and / or positions in order to extend the field of measurement for complex object surfaces under test.

6.1 Information from Different Camera Positions

The object surface observed under multiple viewing directions can be evaluated by comparing the surface normals $\hat{\mathbf{n}}_i$ of an assumed height map under different viewing perspectives. The assumed surface point $\mathbf{r}$ does not have to be observed by all the available cameras. As long as $\mathbf{r}$ can be observed by at least one camera, the surface normal is obtained under this height assumption, through which the measuring field is enlarged.

As discussed in the introduction of this chapter, multi camera positions are conventionally used in PMD to solve the height problem [KHL04a]. In this work, the stereo method introduced in [KHL04a] is applied to solve the height ambiguity of one single surface point $\mathbf{r}_0$ with the surface normal $\hat{\mathbf{n}}_0$ around this point. This point serves as a start value for the following iterative evaluation. Further (new) evaluation is based on the assumption that the object surface is a plane perpendicular to $\hat{\mathbf{n}}_0$ and passing through $\mathbf{r}_0$. Projecting the points on this plane $\mathbf{r}$ onto the ith camera plane ($\mathbf{r}_{ci}$) and evaluating their corresponding screen points ($\mathbf{r}_{si}$), the "revised" surface normals $\hat{\mathbf{n}}_i$ can be evaluated by applying the law of reflection for each single camera. Taking the averaged surface normals from all the available camera perspectives, new surface normals $\hat{\mathbf{n}}$ can be evaluated. Performing an integration, a new surface $\mathbf{r}$ is obtained. Applying this new surface, the surface normals $\hat{\mathbf{n}}_i$ can be updated for every camera perspective. The surface can therefore be evaluated iteratively.

The new algorithm merging data from different cameras can be divided into the following seven steps:

1. **Evaluate the 3D Positions of the observed Screen Points**

   Given the unwrapped phase[2] of the ith camera image $\phi_{xi}(m_{ci}, n_{ci})$ and $\phi_{yi}(m_{ci}, n_{ci})$, where m_{ci} and n_{ci} denotes the pixel index of the ith camera

---

[2] The unwrapped phase denotes the monotonic absolute phase. The phase jump after each period of 2π is removed and the screen coding is unique.

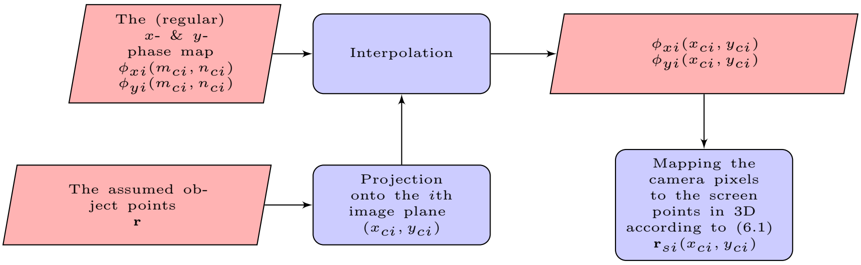


Fig. 6.1: The workflow of evaluating the corresponding screen points according to the phase map and given object points, denoted as Workflow 1.

with $m_{ci}, n_{ci} = 0, 1, 2, 3...$, the phase of the assumed object point $\mathbf{r}$ can be determined by projecting $\mathbf{r}$ onto the ith camera plane and performing an interpolation as shown in Fig. 6.1. The corresponding screen points in 3D space are then determined by

$$\mathbf{r}_{si}(x_{ci}, y_{ci}) = \mathbf{H}_s \begin{bmatrix} \frac{\mu_s n_{px}}{2\pi}\phi_{xi}(x_{ci}, y_{ci}) \\ \frac{\mu_s n_{py}}{2\pi}\phi_{yi}(x_{ci}, y_{ci}) \\ 0 \\ 1 \end{bmatrix}, \tag{6.1}$$

where μ_s denotes the metric screen pixel size, n_{px} and n_{py} denote the x- and y-pattern period given in number of screen pixels, $\mathbf{H}_s$ denotes the homogeneous matrix (Appendix A.5) of the screen and $\mathbf{r}_{si}(x_{ci}, y_{ci})$ denotes the corresponding 3D screen point of the camera point (x_{ci}, y_{ci}). This is a standard step in almost all the PMD evaluations.

2. Define an Evaluation Grid in the Common Coordinate System

A fixed xy-grid in the common coordinate system (x, y) must be defined before the iteration begins (Fig. 6.2). The height of the grid $z(x, y)$ (or the surface $\mathbf{r} = [x, y, z]$) will be adapted after every iteration step according to the assumed local surface normal. The grid can be arbitrarily chosen according to the application. The simplest option is to choose a Cartesian grid which is slightly larger than the object surface. The grid points outside the object surface can not be observed by any camera. They can be removed or masked appropriately after the first iteration.

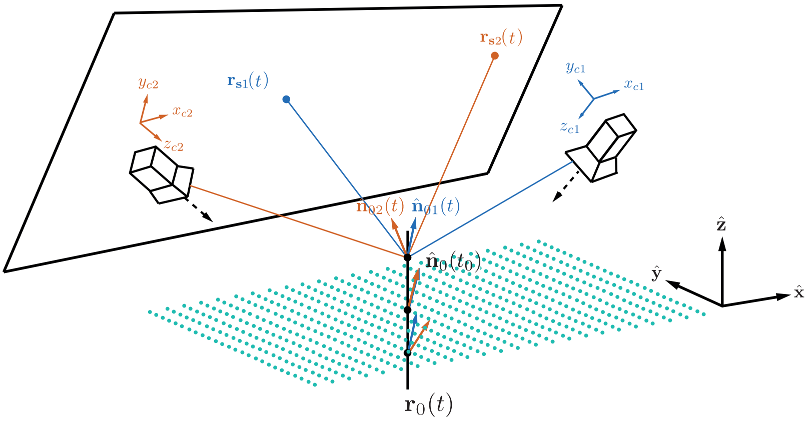


Fig. 6.2: A fixed xy-grid is defined in the common coordinate system $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$. A line segment is defined in the center of the grid and perpendicular to the grid. The real surface point is searched along the line segment by projecting the points on the segment onto every camera plane. Evaluating the corresponding screen points by interpolating the phase map, the surface normal can be determined by applying the law of reflection. The real surface point is found by comparing the surface normals under different camera perspectives and choosing the point along the line segment with the smallest variation of the different surface normals calculated for each camera.

3. Search for an Absolute Surface Point in 3D

A stereo surface point with a known absolute height is found as introduced in [KHL04a]. Define a line segment in the common coordinate system. This line segment, or part of the line segment must be observed by all the cameras. The simplest option is to define a line segment which is perpendicular to the xy-plane in the common coordinate and intersecting the center of the chosen grid (Fig. 6.2). This line segment is denoted as

$$\mathbf{r}_0(t) = [x_0\ y_0\ t]^T, t \in [t_{min}, t_{max}], \tag{6.2}$$

where (x_0, y_0) is the xy-position of the line segment, t is a parameter and $[t_{min}, t_{max}]$ defines the search range.

Projecting $\mathbf{r}_0(t)$ onto the ith camera image plane and denoting the landing point on this image plane as $\mathbf{r}_{c0i}(t)$. The screen point $\mathbf{r}_{s0i}(t)$ is obtained by applying the workflow shown in Fig. 6.1 and (6.1). For a given (assumed) surface point $\mathbf{r}_0(t)$, the direction of the corresponding surface normal "observed" by the ith camera at this point can be obtained by applying the law of reflection as

$$\mathbf{n}_{0i}(t) = \frac{\mathbf{r}_{s0i}(t) - \mathbf{r}_0(t)}{\mid \mathbf{r}_{s0i}(t) - \mathbf{r}_0(t) \mid} + \frac{\mathbf{r}_{c0i}(t) - \mathbf{r}_0(t)}{\mid \mathbf{r}_{c0i}(t) - \mathbf{r}_0(t) \mid}, \tag{6.3}$$

and in normalized form

$$\hat{\mathbf{n}}_{0i}(t) = \frac{\mathbf{n}_{0i}(t)}{\mid \mathbf{n}_{0i}(t) \mid}. \tag{6.4}$$

$\hat{\mathbf{n}}_{0i}(t)$ denotes the surface normal observed by the ith camera under the assumption that $[x_0\ y_0\ t]^T$ is an actual surface point. Under a wrong assumption, $\hat{\mathbf{n}}_{0i}(t)$ differs for different camera perspectives. The real surface point is obtained by searching for the t value where all the $\hat{\mathbf{n}}_{0i}(t)$ are maximally colinear. Denoting this t value as t_0, a real surface point is specified as $\mathbf{r}_0(t_0) = [x_0\ y_0\ t_0]^T$ and the surface normal around this point as $\hat{\mathbf{n}}_0(t_0)$ (obtained by averaging over all $\hat{\mathbf{n}}_{0i}(t_0)$).

4. Define a Surface According to the Absolute Surface Point and its Normal

In a first (rough) approximation, the object surface can be assumed to be a plane going through the point $\mathbf{r}_0(t_0)$ and perpendicular to $\hat{\mathbf{n}}_0(t_0)$. This plane is denoted as $\mathbf{r}_k$ with $k = 0$, where k denotes the kth iteration step (Fig. 6.3).

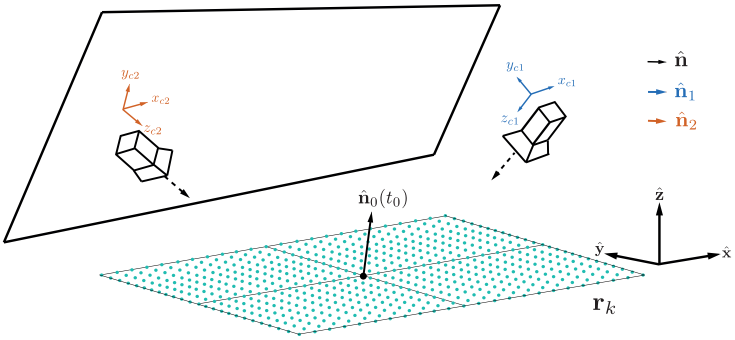


Fig. 6.3: The first assumption of the height of the grid points is obtained by a plane defined by the point $\mathbf{r}_0(t_0)$ and its surface normal $\hat{\mathbf{n}}_0(t_0)$.

5. Evaluate the Surface Normals by taking all Cameras into Account

The corresponding image point of the surface point $\mathbf{r}_k$ is obtained by projecting $\mathbf{r}_k$ onto the ith image plane using the camera calibration data. The image point is denoted as $\mathbf{r}_{cki}$ or presented in the ith local camera coordinate system as (x_{cki}, y_{cki}). If the projected point is outside the field of view of the ith camera, the surface point *under this camera perspective* $\mathbf{r}_{cki}$ must be removed. The corresponding screen point $\mathbf{r}_{ski}$ is obtained by applying Step. 1 and (6.1). The surface normal $\hat{\mathbf{n}}_{ki}$ observed by the ith camera under the height assumption of $\mathbf{r}_k$ can then be determined by 6.3 and 6.4.

Obviously, under a wrong height assumption, the surface normals $\hat{\mathbf{n}}_{ki}$ of the surface point $\mathbf{r}_k$ are different under diverse camera perspectives (Fig. 6.4). A new surface normal can be obtained by averaging the assumed normals from all the camera perspectives:

$$\hat{\mathbf{n}}_{k+1} = \frac{\sum\limits_i \hat{\mathbf{n}}_{ki}}{\mid \sum\limits_i \hat{\mathbf{n}}_{ki} \mid}. \tag{6.5}$$

Note that for a surface point $\mathbf{r}_k$, its normal vector $\hat{\mathbf{n}}_{ki}$ does not have to exist for all the camera perspectives. The new surface normal $\hat{\mathbf{n}}_{k+1}$ can be obtained if the assumed normal $\hat{\mathbf{n}}_{ki}$ is available for at least one camera perspective. The evaluation field is therefore not limited by any single camera. The evaluation field is in this stage a combined field of all the cameras (Fig. 6.4).

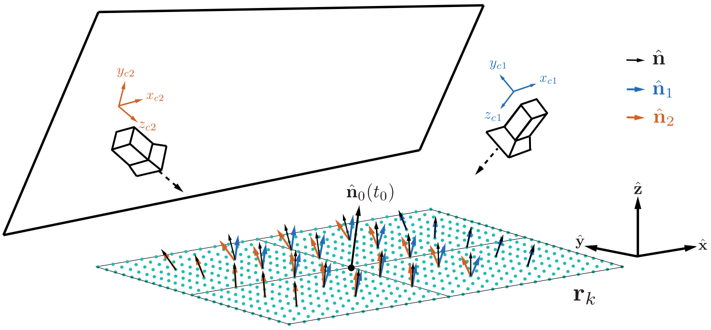


Fig. 6.4: The local surface normals for each grid point can be evaluated under the height assumption introduced in Step. 4 for each camera (blue and orange). New "consolidatedßurface local normal $\hat{\mathbf{n}}(x, y)$ (black) are obtained for each grid point by averaging over the different surface normals calculated for each camera perspective.

6. Evaluate the Surface According to the New Surface Normals

The shape of the surface can now be newly reconstructed according to the normals determined in the Step. 5 by applying a $2D$ integration (Fig. 6.5). Reconstructing a surface through the normal field / the gradient field is a frequently posed mathematical question that has been studied intensively in the field of 3D metrology. This can be done by different mathematical methods. One possible method is based on the global least squares solution with Tikhonov regularization [HO13a][HO13b][HO11][HO08]. A second option is to fit a radial basis function (RBF) in local patches and then to reduced to height distances of the overlapping points sets [FC88][Ett+08][EKH07][LOW02][LK05][Low05].

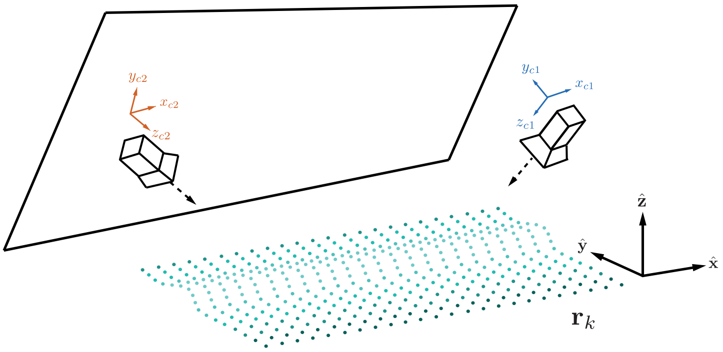


Fig. 6.5: The height of the object surface is obtained by performing a 2D-integration of the normal vector field calculated in Step. 5.

7. Iteration Loop: Repeat Step. 5 and Step. 6

Use the updated surface points $\mathbf{r}_{k+1}$ determined in Step. 6 as input data and perform Step. 5 again, until the stopping criterion is fulfilled. A possible criterion is for example surface normal $\max|\,\mathbf{r}_{k+1} - \mathbf{r}_k\,| < \textit{threshold}$ or does not change[3].

The entire multi cameras evaluation is introduced in the workflow displayed Fig. 6.6.

---

[3] Another criterion could be that the surface normals $\hat{\mathbf{n}}_{ki}$ from the different camera views differ slightly enough - this would correspond to a "(multi-)stereo criterion"

6.2 Information from Different Object Positions

In addition to multiple camera positions, the measuring field can also be extended by combining data obtained from multiple object positions. Obtaining / acquiring the data from different object positions is in principle easy and can be done without any additional hardware if the object is moving anyway. The object only needs to be captured several times by all the available cameras during object movement as displayed in Fig. 6.7. The measurement field is extended by the multiple views which is generated by the object movement.

However, data fusion of multiple object positions is very difficult due to the reflectivity of the object surface. A reflective surface is invisible and has to be considered as a part of the imaging system, effectively imaging the (distorted) screen pattern used for illumination. The exact location and shape of the object, is unknown until the 3D point cloud of the object surface has been fully evaluated. Up to this point, the surface normals are only assumed under a height assumption - the height ambiguity. In the case of multiple object positions, an additional lateral ambiguity is introduced: The surface normal correspondence between the jth position to the $j+1$th position is not traceable, therefore a normal comparison or normal fusion is not possible in this early stage of the evaluation.

To be able to compare and combine the assumed surface normals from different object positions, the surface normals must be considered in the same coordinate system in which the object does not move, so that no further ambiguity is added. The previous $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$-coordinate system is no longer suitable since the object has a relative motion in these coordinates. A new coordinate system fixed to the object muss be introduced. The fixed coordinate can be achieved by using a target carrier with fixed markers on it (Fig. 6.8). Since the target carrier moves with the object, there is no relative movement between the markers and the object. Such a fixed "object coordinate system" $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ can be defined basing on the markers (Fig. 6.8). As the markers are not reflective, the positions of the markers in the previous $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$-coordinates are easy to evaluate using triangulation [HS97]. The transform between the moving coordinate $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ and the common coordinate $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$ are performed by the homogeneous matrix (s. Appendix A.5),

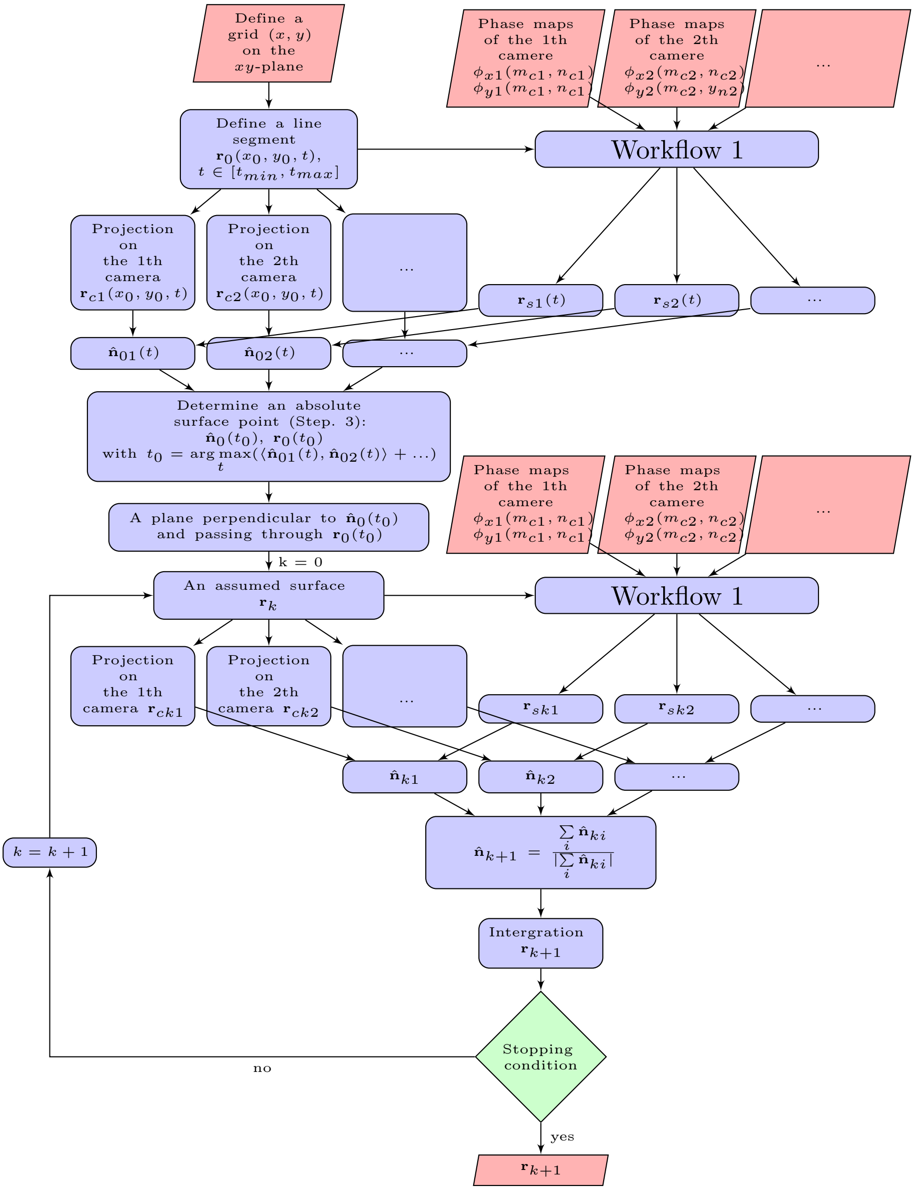


Fig. 6.6: The workflow of evaluating the object surface by applying multiple camera positions.

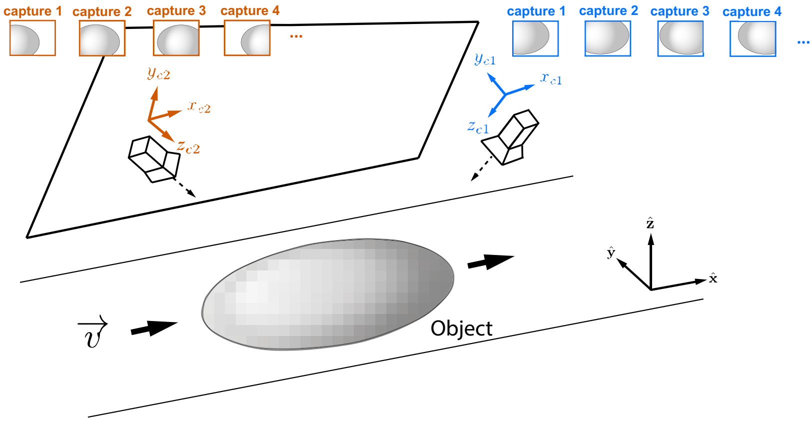


Fig. 6.7: Multi position PMD.

where j denotes the jth camera image exposure.

Fig. 6.8: target carrier with fixed markers to define a coordinate $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ that moves together with the object. (courtesy of Simon Hartel)

By applying the coordinate system - $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ moving with the object, the camera image captured in the jth object position can be considered as an image captured by the jth slightly shifted virtual camera and screen (as shown in Fig. 6.9), with the object always remaining in the same position. The algorithm introduced in Chapter 6.1 comparing the surface normals from multiple camera perspectives can therefore now be applied for multiple object positions as well while all the screen points $\mathbf{r}_{sj}$ and image points $\mathbf{r}_{cj}$ are presented in the $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$-coordinate by using the inverse[4] of $\mathbf{H}_{tj}$.

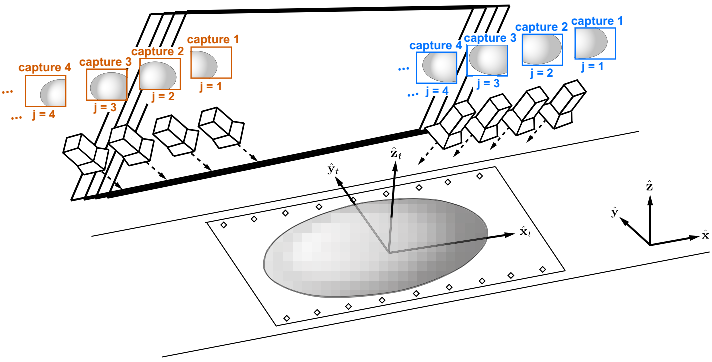


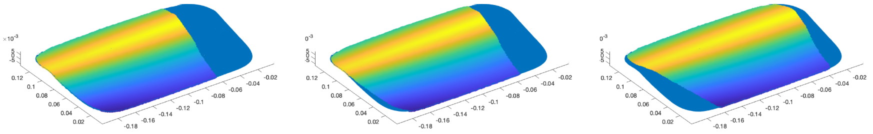
Fig. 6.9: Fixed coordinate with respect to the object. Cameras and screen are moving relative to the object.

The petrol cap displayed in Fig. 2.12 is under test. The petrol cap is captured five times during its movement through the sensor. It is evaluated by combining and comparing the assumed surface normals from all the object positions by applying the homogeneous matrix $\mathbf{H}_{tj}$. The measuring result is shown in Fig. 6.10 at the bottom right. The object is also evaluated in every position independently and the independently evaluated point clouds are demonstrated in Fig. 6.10 as well. As shown in Fig. 6.10, only a part of the object surface is acquired in every single object position.

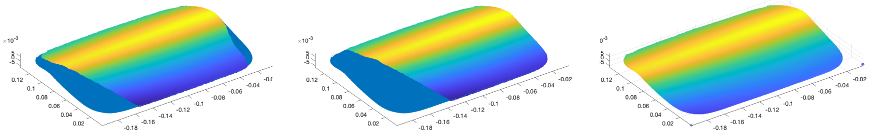
The topography of the petrol cap was measured stationary with a high-

---

[4] To perform a numerically stable inverse of $\mathbf{H}_{tj}$, the following approach is applied: $\mathbf{H}_{tj}^{-1} = \begin{bmatrix} \mathbf{R}_{tj}^T & -\mathbf{R}_{tj}^T\mathbf{t}_{tj} \\ 0 & 1 \end{bmatrix}$.



Position 1 Position 2 Position 3



Position 4 Position 5 Surface evaluated by combined normals

Fig. 6.10: A petrol cap is measured in motion. The petrol cap is captured five times in five different positions during the movement. Only a part of the patrol cap is measured in each measuring position. The entire surface is evaluated by combining the normals acquired in every position and the result is displayed at the bottom right.

precision confocal sensor as well. The measurement result from the confocal sensor is viewed as the ground truth. The comparison of the multi-position deflectometry measurement in motion at a speed of 200 mm/s and the time consuming stationary confocal measurement is shown in Fig. 6.11. For the large measuring object of about 100 mm wide and 160 mm long, which is substantially uncooperative for a deflectometric measurement due to the large curvature, the multi-position deflectometric measurement in motion delivers only height deviations (sometimes even significantly) smaller than 50 μm.

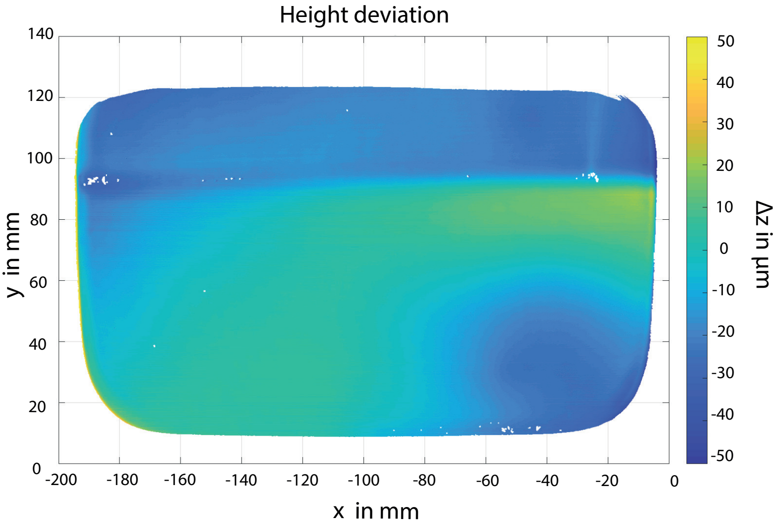


Fig. 6.11: Height deviation of a multi-position deflectometry measurement in motion of a petrol cap against a ground truth measured with a precise confocal sensor.

6.3 Information from Different Measuring Systems

The multi-view and multi-position-measurements created by a moving object can increase the measurement field. However, the (lost) information of the object side facing the screen itself, which is observed by the camera with a large angle, cannot be captured by adding observation views from the same side. To recover this information, the object must be removed and placed back on the target carrier, but with this side of the object surface facing the camera view, and then measured again.

Alternatively, the object can also be measured by two sensors oriented in opposite directions and located one after the other. Such a combined sensor system is shown in Fig. 6.12. The combined sensor system consists of two identical PMD sensors, each with two cameras. The two PMD sensor units are mounted one after the other in the object moving direction - along the $\hat{\mathbf{x}}_t$-axis. Each sensor has its own coordinate system $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$, marked in red and blue. The $\hat{\mathbf{x}}$-axes of the two sensors are directed in the opposite directions, as well as the $\hat{\mathbf{y}}$-axes. The object is placed on the target carrier. The associated moving coordinate system, defined by the markers on the target carrier, is shown in green. The object moves along the $\hat{\mathbf{x}}_t$-axis and both sides of the object surface are now observed in the respective sensor units.

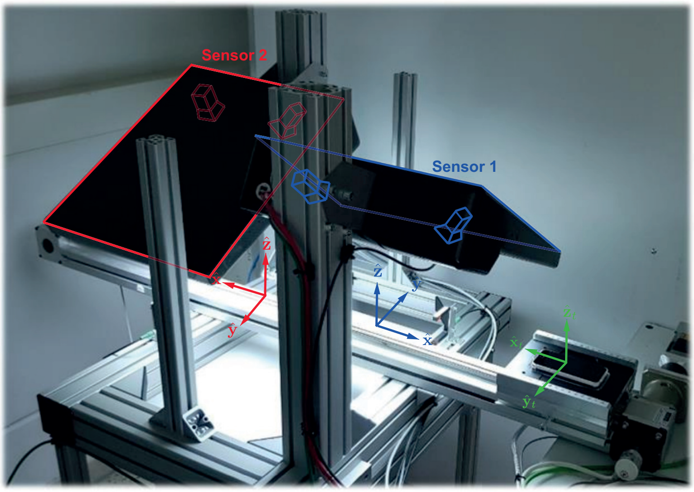


Fig. 6.12: The combined sensor system consists of two identical PMD sensors. (blue and red). The two PMD sensor units are mounted one after the other in the object moving direction and facing the opposite direction. (courtesy of Simon Hartel)

Like the multi-position PMD, the data from different measuring stations can be also evaluated in a merged way by applying the same algorithm as introduced above. The sensor 1 and the sensor 2 must be calibrated in their own coordinates

$\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$, respectively. The object moves together with the markers on the object carrier. The position of the markers can be evaluated in each measurement by triangulation, thus, the position of the coordinate $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ relative to $\hat{\mathbf{x}}\hat{\mathbf{y}}\hat{\mathbf{z}}$ can be determined. The transformation matrix is denoted by $\mathbf{H}_{tjm}$, where j and m stand for the jth camera exposure in the mth measuring station. Representing the data points (screen points and camera points) of different measuring stations in $\hat{\mathbf{x}}_t\hat{\mathbf{y}}_t\hat{\mathbf{z}}_t$ by applying the inverse of the homogeneous matrix $\mathbf{H}_{tjm}$, the surface normals from different measuring stations can be considered in a common coordinate system. Thus the object surface can be evaluated with the algorithm introduced in Chapter 6.1.

Chapter 7

Conclusion and Outlook

Measuring the topography of specular surfaces with strong surface structures in motion was impossible before this research. A new method based on single-shot phase-measuring deflectometry (SSPMD) and combining different solution aspects has been presented.

Specular surfaces can be measured in motion using SSPMD, which uses the additive superposition of two perpendicular fringe patterns as a diffuse light source, and then uses single-sideband demodulation to separate the fringes of different directions and determine the phases. However, this approach only works for sufficiently smooth object surfaces without strong surface structures, since the optical effect of the varying radii of curvature associated with strong surface structures results in heavily changing magnification factors being applied to the spatial fringe frequency observed via the surface. The camera image of such object surfaces is not a narrowband signal anymore. Therefore, single-sideband demodulation cannot be applied. The broadband problem arising in conventional SSPMD has been solved in this research by an adapted localized Ridgelet transform, in which the varying frequency components are considered locally. The additively superimposed fringes, which encode the x- and y-positions of the screen respectively, are separated by localized ridgelets in different directions. Using a petrol cap as a test object, which has strong surface structures and can not be measured with conventional SSPMD, the geometric positions on the illuminating screen observed via the object surface by the different camera pixels (forming the primary measurand of deflectometry) could be determined by applying the localized ridgelet transform in screen pixel accuracy (screen pixel size: 297 μm), which corresponds to a surface slope variation of 5 *arcmin* in the laboratorial PMD setup. Such an object was impossible to be measured with only one camera image before this research. A limitation of the localized ridgelet transform is still the *local* broadband problem, where the signal frequency varies strongly within only a few fringe periods in local areas. A possible improvement for these kinds of situations using the chirplet transform was discussed in Chapter 4.3.

Since the screen - the encoding light source of PMD - is diffusely illuminating, the direction of incident ray is unknown. For each point along the reflection ray, a surface normal which satisfies the law of reflection can always be found. The height and the normal of the surface can not be determined simultaneously and unambiguously. This problem is called the height problem of PMD. The height problem has been solved in a new way by adaption and substantial extension of an existing discrete differential approach based on two lateral curves on the continuous fullfield PMD phase data. Not only the screen point to camera point correspondence, but also the existing (but up to now not exploited - at least not for the purpose of solving the height problem) spatial context information of the local $\mathbb{R}^2 \to \mathbb{R}^2$ mapping from the PMD screen plane to the camera plane (vice versa) has been utilized, which makes an absolute monocular PMD possible. Simulations and practical test results have been shown in Chapter 5.2. Simulations showed that in principle, a height measuring accuracy of up to 10 nm is achievable for extremely high SNRs. However, like most differential approaches, the proposed method is also sensitive to signal noise. Despite this, a measurement accuracy of ± 500 μm for a planar mirror has been reached. This allows this method to be used for a start point search in certain situations (with the final surface still being evaluated by a proper integration of the deflectometric slope data determined under this height assumption). Limitations of this method have been shown and a physical interpretation of those limitations has been given in Chapter 5.3.4. Furthermore, a new generalized solution without the necessity of pre-defining specific curves was presented and discussed in Chapter 5.4.

In order to increase the robustness and to enlarge the measurement field when measuring complex objects, a novel multi-position resp. multi-station-PMD has been developed. Furthermore, for object surfaces with extremely high dynamical curvature variations, two fringe patterns with different frequencies could be needed. These different patterns should be displayed in different screens (stations) due to the time consuming switching between patterns in only one screen. Therefore, not only data from different camera views, but also from different measuring stations should be merged. However, the specular object surface is invisible. A (further) lateral ambiguity is introduced when gathering data acquired in motion or from different measuring stations. To avoid the additional ambiguity, a coordinate system defined by matt markers moving together with the object has been applied. The camera position and the screen position can be determined in this (moving) coordinate system in every single measurement by detecting these markers. Every further acquisition is therefore contributing an additional (virtual) measuring system, which is completely described in this moving coordinate system. A new algorithm has been developed based on this approach, which combines the data from all virtual measurement systems and determines the 3D shape of the object surface. Since all measuring units are considered equally and only the averaged measured values are considered, the new evaluation algorithm is not limited by the number of measuring stations. The measuring system can be flexibly extended as required.

Results from simulations as well as practical experiments have shown that the combination of the presented approaches allows for the measurement of specular objects even with strongly curved surface structures in motion, which was not the case so far. A petrol cap as device under test, which has a height dynamic of about 10 mm, was measured in motion with a speed of up to 200 mm/s. A measurement error of only ± 30 μm was achieved.

This research enables automated measurement of objects with specular surfaces in motion, through which the costs for quality control can be considerably reduced. This approach can be applied to various inspection scenarios, such as inspection systems for smartphones, tablets, painted car bodies, lenses, mirrors or porcelain. Both scientific and economic benefits can be achieved through this research.

Bibliography

[Nyq28] H. Nyquist. „Certain Topics in Telegraph Transmission Theory“. In: *Transactions of the American Institute of Electrical Engineers* 47.2 (1928), S. 617–644. DOI: `10.1109/T-AIEE.1928.5055024`.

[Wig32] E. Wigner. „On the Quantum Correction For Thermodynamic Equilibrium“. In: *Phys. Rev.* 40 (5 Juni 1932), S. 749–759. DOI: `10.1103/PhysRev.40.749`.

[Gab46] D. Gabor. „Theory of communication. Part 1: The analysis of information“. In: *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering* 93 (26 Nov. 1946), 429–441(12). ISSN: 0367-7540. DOI: `10.1049/ji-3-2.1946.0074`. URL: `https://digital-library.theiet.org/content/journals/10.1049/ji-3-2.1946.0074`.

[HS81] Berthold Horn und Brian Schunck. „Determining Optical Flow“. In: *Artificial Intelligence* 17 (Aug. 1981), S. 185–203. DOI: `10.1016/0004-3702(81)90024-2`.

[LK81] Bruce D. Lucas und Takeo Kanade. „An Iterative Image Registration Technique with an Application to Stereo Vision“. In: *Proceedings of 7th International Joint Conference on Artificial Intelligence (IJCAI '81).* Aug. 1981, S. 674–679.

[TIK82] Mitsuo Takeda, Hideki Ina und Seiji Kobayashi. „Fourier-transform method of fringe-pattern analysis for computer-based topography and interferometry“. In: *J. Opt. Soc. Am.* 72.1 (1982), S. 156–160. DOI: `10.1364/JOSA.72.000156`. URL: `http://www.osapublishing.org/abstract.cfm?URI=josa-72-1-156`.

[Kos83] A. I. Kostrikin. „Introduction to algebra“. In: *Proceedings of the Edinburgh Mathematical Society* 26.3 (1983). DOI: `10.1017/S0013091500004557`.

[Cre87] Katherine Creath. „Comparison Of Phase-Measurement Algorithms“. In: *Surface Characterization and Testing*. Hrsg. von Katherine Creath. Bd. 0680. International Society for Optics und Photonics. SPIE, 1987, S. 19–28. DOI: `10.1117/12.939587`. URL: `https://doi.org/10.1117/12.939587`.

[FC88] R. T. Frankot und R. Chellappa. „A method for enforcing integrability in shape from shading algorithms“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10.4 (1988), S. 439–451. DOI: `10.1109/34.3909`.

[Kin+88] K. Kinnstaetter u. a. „Accuracy of phase shifting interferometry“. In: *Appl. Opt.* 27.24 (Dez. 1988), S. 5082–5089. DOI: `10.1364/AO.27.005082`.

[Gru91] M. Gruber. „Ein phasenmessendes Triangulationsverfahren zur parallelen optischen Erfassung der Form dreidimensionaler Objekte“. Diplomarbeit. University of Erlangen-Nuremberg, 1991.

[JB94] T.R. Judge und Peter Bryanston-cross. „A review of phase unwrapping techniques in fringe analysis“. In: *Optics and Lasers in Engineering* 21 (Dez. 1994), S. 199–239. DOI: `10.1016/0143-8166(94)90073-6`.

[Sch94] Christophe Schlick. „An Inexpensive BRDF Model for Physically-based Rendering“. In: *Computer Graphics Forum* 13.3 (1994), S. 233–246. ISSN: 1467-8659. DOI: `10.1111/1467-8659.1330233`.

[MH95] S. Mann und S. Haykin. „The chirplet transform: physical considerations“. In: *IEEE Transactions on Signal Processing* 43.11 (1995), S. 2745–2761. DOI: `10.1109/78.482123`.

[Wen95] Holger Wendland. „Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree.“ In: *Adv. Comput. Math.* 4.1 (1995), S. 389–396.

[SML96] R. G. Stockwell, L. Mansinha und R. P. Lowe. „Localization of the complex spectrum: the S transform“. In: *IEEE Transactions on Signal Processing* 44.4 (1996), S. 998–1001. DOI: `10.1109/78.492555`.

[FS97] G.B. Folland und A. Sitaram. „The uncertainty principle: A mathematical survey“. In: *Fourier Analysis and Applications* 3 (1997), S. 207–238. DOI: `10.1007/BF02649110`.

[HS97] Richard I. Hartley und Peter Sturm. „Triangulation“. In: *Computer Vision and Image Understanding* 68.2 (1997), S. 146–157. ISSN: 1077-3142. DOI: `https://doi.org/10.1006/cviu.1997.0547`.

[PB97] Denis Perard und Juergen Beyerer. „Three-dimensional measurement of specular free-form surfaces with a structured-lighting reflection technique“. In: *Three-Dimensional Imaging and Laser-based Systems for Metrology and Inspection III.* Hrsg. von Kevin G. Harding und Donald J. Svetkoff. Bd. 3204. International Society for Optics und Photonics. SPIE, 1997, S. 74–80. DOI: `10.1117/12.294443`.

[Can98] Emmanuel Jean Candes. „Ridgelets: Theory and Applications“. Diss. Stanford University, 1998.

[Hor98] Christoph Horneber. „Entwicklung eines Lichtmodulators zur Streifenprojektion in der optischen Messtechnik“. Diplomarbeit. University of Erlangen-Nuremberg, 1998.

[Häu99] Gerd Häusler. „Verfahren und Vorrichtung zur Ermittlung der Form oder der Abbildungseigenschaften von spiegelnden oder transparenten Objekten “. DE19944354A1. 1999.

[SCR99] Giovanna Sansoni, Matteo Carocci und Roberto Rodella. „Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors“. In: *Appl. Opt.* 38.31 (Nov. 1999), S. 6565–6573. DOI: `10.1364/AO.38.006565`.

[SH00] Rainer Seßner und Gerd Häusler. „Verfahren zum optischen Messen der Form spiegelnder Oberflächen “. DE10014964C2. 2000.

[PR01] Marcus Petz und Reinhold Ritter. „Reflection grating method for 3D measurement of reflecting surfaces“. In: *Optical Measurement Systems for Industrial Inspection II: Applications in Production Engineering.* Hrsg. von Roland Hoefling, Werner P. O. Jueptner und Malgorzata Kujawinska. Bd. 4399. International Society for Optics und Photonics. SPIE, 2001, S. 35–41. DOI: `10.1117/12.445587`. URL: `https://doi.org/10.1117/12.445587`.

[Add02] P.S. Addison. „The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science“. In: *Engineering, Medicine and Finance* (Jan. 2002).

[LOW02] SVENJA LOWITZSCH. „APPROXIMATION AND INTERPOLATION EMPLOYING DIVERGENCE FREE RADIAL BASIS FUNCTIONS WITH APPLICATIONS“. Diss. Texas AM University, 2002.

[Lam03] Ralf Lampalzer. „Physikalische und informationstheoretische Eigenschaften und Grenzen von Systemen zur optischen Formerfassung und Inspektion nach dem Prinzip der phasen-messenden Triangulation“. Diss. University of Erlangen-Nuremberg, 2003.

[Bot+04] Thorsten Bothe u. a. „High-resolution 3D shape measurement on specular surfaces by fringe reflection“. In: *Optical Metrology in Production Engineering.* Hrsg. von Wolfgang Osten und Mitsuo Takeda. Bd. 5457. International Society for Optics und Photonics. SPIE, 2004, S. 411–422. DOI: `10.1117/12.545987`.

[Kam04a] J. Kaminski. „Geometrische Rekonstruktion spiegelnder Oberflächen aus deflektometrieschen Messdaten“. Diss. University of Erlangen-Nuremberg, 2004.

[Kam04b] J. Kaminski. „Geometrische Rekonstruktion spiegelnder Oberflächen aus deflektometrieschen Messdaten“. Diss. University of Erlangen-Nuremberg, 2004.

[KHL04a] Markus C. Knauer, Gerd Häusler und R. Lampalzer. *Verfahren und Vorrichtung zur Bestimmung der Form und der lokalen Oberflächennormalen spiegelnder Oberflächen.* DE102004020419 B3. 2004.

[KHL04b] Markus Knaver, Gerd Haeusler und Ralf Lampalzer. „Verfahren und Vorrichtung zur Bestimmung der Form und der lokalen Oberflächennormalen spiegelnder Oberflächen “. DE102004020419B3. 2004.

[SCP04] Silvio Savarese, Min Chen und Pietro Perona. „Recovering Local Shape of a Mirror Surface from Reflection of a Regular Grid“. In: *Computer Vision - ECCV 2004*. Hrsg. von T. Pajdla und J. Matas. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, S. 468–481. ISBN: 978-3-540-24672-5.

[SH04] Rainer Seßner und Gerd Häusler. „Richtungscodierte Deflektometrie“. In: *DGaO*. 2004, A16.

[BWS05] Andres Bruhn, Joachim Weickert und Christoph Schnörr. „Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods“. In: *International Journal of Computer Vision* 61 (Feb. 2005), S. 211–231. DOI: `10.1023/B:VISI.0000045324.43199.43`.

[Car05] P Carré. „Installation et utilisation du comparateur photoélectrique et interférentiel du Bureau International des Poids et Mesures“. In: *Metrologia* 2 (Jan. 2005), S. 13. DOI: `10.1088/0026-1394/2/1/005`.

[Low05] Svenja Lowitzsch. „Matrix-valued radial basis functions: stability estimates and applications“. In: *Advances in Computational Mathematics* 23 (2005), S. 299–315.

[LK05] Svenja Lowitzsch und M. Knauer. „Vision and Modeling of Specular Surfaces“. In: 2005.

[SCP05] Silvio Savarese, Min Chen und Pietro Perona. „Local Shape from Mirror Reflections“. In: *International Journal of Computer Vision* 64 (Aug. 2005), S. 31–67. DOI: `10.1007/s11263-005-1086-x`.

[BSG06] Thomas Bonfort, Peter Sturm und Pau Gargallo. „General Specular Surface Triangulation“. In: *Computer Vision – ACCV 2006*. Hrsg. von P. J. Narayanan, Shree K. Nayar und Heung-Yeung Shum. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, S. 872–881. ISBN: 978-3-540-32432-4.

[RKW06] Ron Rykowski, Doug Kreysar und Sipke Wadman. „9.3: The Use of an Imaging Sphere for High-Throughput Measurements of Display Performance — Technical Challenges and Mathematical Solutions“. In: *Sid Symposium Digest of Technical Papers* 37 (Juni 2006). DOI: `10.1889/1.2433146`.

[Dea07] S.R. Deans. *The Radon Transform and Some of Its Applications*. Dover Books on Mathematics Series. Dover Publications, 2007. ISBN: 9780486462417. URL: `https://books.google.de/books?id=xSCc0KGi0u0C`.

[EKH07] S. Ettl, J. Kaminski und G. Häusler. „Generalized Hermite interpolation with radial basis functions considering only gradient data“. In: 2007.

[Kic07] Reiner Kickingereder. „Optische Vermessung partiell reflektierender Oberflächen“. Diss. University of Passau, 2007.

[Bal08] Jonathan Balzer. „Regularisierung des Deflektometrieproblems - Grundlagen und Anwendung“. Diss. University of Karlsruhe, 2008.

[CXJ08] Xiaobo Chen, Juntong Xi und Ye Jin. „Phase error compensation method using smoothing spline approximation for a three-dimensional shape measurement system based on gray-code and phase-shift light projection“. In: *Optical Engineering* 47.11 (2008), S. 1–9. DOI: `10.1117/1.3026075`.

[Ett+08] Svenja Ettl u. a. „Shape reconstruction from gradient data“. In: *Appl. Opt.* 47.12 (Apr. 2008), S. 2091–2097.

[HO08] M. Harker und P. O'Leary. „Least squares surface reconstruction from measured gradient fields“. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. 2008, S. 1–7. DOI: `10.1109/CVPR.2008.4587414`.

[Küh08] Wolfgang Kühnel. *Differentialgeometrie*. Vieweg+Teubner Verlag, 2008. ISBN: 978-3-8348-9453-3.

[Mal08] Stephane Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. 3rd. USA: Academic Press, Inc., 2008. ISBN: 0123743702.

[SB09] Eike Slogsnat und Karl-Heinz Brenner. „Non-stereoscopic Method for Deflectometric Measurement of Reflecting Surfaces“. In: *DGaO*. Juli 2009, A11.

[YG09] Pochi Yeh und Claire Gu. *Optics of Liquid Crystal Displays, 2nd Edition*. Wiley Sons Ltd, 2009. ISBN: 978-0-470-18176-8.

[Zei+10] Angela Zeiler u. a. „Empirical Mode Decomposition - an introduction“. In: Juli 2010, S. 1–8. DOI: `10.1109/IJCNN.2010.5596829`.

[HO11] M. Harker und P. O'Leary. „Least squares surface reconstruction from gradients: Direct algebraic methods with spectral, Tikhonov, and constrained regularization“. In: *CVPR 2011*. 2011, S. 2529–2536. DOI: `10.1109/CVPR.2011.5995427`.

[HNA11] Lei Huang, Chi Seng Ng und Anand Krishna Asundi. „Dynamic three-dimensional sensing for specular surface with monoscopic fringe reflectometry“. In: *Opt. Express* 19.13 (Juni 2011), S. 12809–12814. DOI: `10.1364/OE.19.012809`.

[BLC12] R.-F Bai, Bing-Zhao Li und Q.-Y Cheng. „Wigner-Ville Distribution Associated with the Linear Canonical Transform“. In: *Journal of Applied Mathematics* 2012 (Jan. 2012). DOI: `10.1155/2012/740161`.

[HO13a] Matthew Harker und Paul O'Leary. *Regularized Reconstruction of a Surface from its Measured Gradient Field*. 2013. eprint: `1308.4292`.

[HO13b] Matthew Harker und Paul O'Leary. „Direct regularized surface reconstruction from gradients for Industrial Photometric Stereo“. In: *Computers in Industry* 64.9 (2013). Special Issue: 3D Imaging in Industry, S. 1221–1228. ISSN: 0166-3615. DOI: `https://doi.org/10.1016/j.compind.2013.03.013`.

[Liu+14] Yuankun Liu u. a. „Fast and accurate deflectometry with crossed fringes“. In: *Advanced Optical Technologies* 3.4 (2014). DOI: `10.1515/aot-2014-0032`. URL: `https://doi.org/10.1515/aot-2014-0032`.

[LCZ14] William Lohry, Vincent Chen und Song Zhang. „Absolute three-dimensional shape measurement using coded fringe patterns without phase unwrapping or projector calibration“. In: *Opt. Express* 22.2 (Jan. 2014), S. 1287–1301. DOI: `10.1364/OE.22.001287`.

[Ole+14] Evelyn Olesch u. a. „Deflectometric measurement of large mirrors“. In: *Advanced Optical Technologies* 3 (Juni 2014). DOI: `10.1515/aot-2014-0023`.

[Lok15] Firdous Ahmad Shah Lokenath Debnath. *Wavelet Transforms and Their Applications*. Birkhäuser Boston, 2015. ISBN: 978-0-8176-8418-1.

[Lia+16] Hanning Liang u. a. „Single-shot phase-measuring deflectometry for cornea measurement“. In: *Advanced Optical Technologies* 5.5-6 (2016), S. 433–438. DOI: `doi:10.1515/aot-2016-0049`.

[SEV16] Narasimman Sundararajan, A. Ebrahimi und N. Vasudha. „Two Dimensional Short Time Hartley Transforms“. In: *Sultan Qaboos University Journal for Science [SQUJS]* 21 (Nov. 2016), S. 41. DOI: `10.24200/squjs.vol21iss1pp41-47`.

[SKS17] Kamal Sindhu, Vikas Kumar und Ajay Shankar. „A Review on Phase Unwrapping Techniques in Interferometry“. In: *International Journal of Advanced Research in Electronics and Communication Engineering* 6 (Juli 2017), S. 798–802.

[Gro18] Herbert Gross. *Lens Design II.* 2018. URL: `https://www.iap.uni-jena.de/iapmedia/de/Lecture/Lens+Design+II1519858800/LDII_Lens+Design+II+_+12+Mirror+systems-p-20002953.pdf`.

[LF18] Hanning Liang und Christian Faber. „Deflektometrie zur schnellen 3D-Inline-Inspektion in der Bewegungung“. In: *DGaO.* 2018, P001. Poster presented at DGaO, Aalen.

[Zha18] Song Zhang. „Absolute phase retrieval methods for digital fringe projection profilometry: A review“. In: *Optics and Lasers in Engineering* 107 (2018), S. 28–37. ISSN: 0143-8166. DOI: `https://doi.org/10.1016/j.optlaseng.2018.03.003`.

[Coh19] Michael X Cohen. „A better way to define and describe Morlet wavelets for time-frequency analysis“. In: *NeuroImage* 199 (2019), S. 81–86. ISSN: 1053-8119. DOI: `https://doi.org/10.1016/j.neuroimage.2019.05.048`.

[Lia+19a] Hanning Liang u. a. „A new method for solving the height problem in deflectometry“. In: *Applied Optical Metrology III.* Hrsg. von Erik Novak und James D. Trolinger. Bd. 11102. International Society for Optics und Photonics. SPIE, 2019, S. 173–179. URL: `https://doi.org/10.1117/12.2527747`.

[Lia+19b] Hanning Liang u. a. „Eine neue Methode zur Lösung des Höhenproblems in der Deflektometrie“. In: *DGaO.* 2019, B14.

[Lia+19c] Hanning Liang u. a. „Verfahren und System zum optischen Vermessen eines Objekts mit spiegelnder und/oder teilspiegelnder Oberfläche sowie entsprechende Messanordnung“. DE102019208474 / WO2020249166 , (the international patent procedure not yet completed by the time of submission of the dissertation). 2019.

[Wil+19] Florian Willomitzer u. a. *Uncalibrated Deflectometry with a Mobile Device on Extended Specular Surfaces.* 2019. arXiv: `1907.10700 [cs.CV]`.

[LSF20] Hanning Liang, Tomas Sauer und Christian Faber. „Using wavelet transform to evaluate single-shot phase measuring deflectometry data“. In: *Applications of Digital Image Processing XLIII.* Hrsg. von Andrew G. Tescher und Touradj Ebrahimi. Bd. 11510. International Society for Optics und Photonics. SPIE, 2020, S. 404–410. URL: `https://doi.org/10.1117/12.2567301`.

[Wil+20] Florian Willomitzer u. a. „Hand-guided qualitative deflectometry with a mobile device“. In: *Opt. Express* 28.7 (März 2020), S. 9027–9038.

[Vil] J. Ville. *Theorie et Applications de la notion de signal analytique.* URL: `https://books.google.de/books?id=JVp1twAACAAJ`.

Acknowledgement

This work would not have been accomplished without the support and help of many people during my time as a PhD student. I am deeply grateful to each and every one of you.

First of all, my biggest thanks go to my supervisor, Prof. Dr. Christian Faber, who gave me the opportunity to work at Landshut University of Applied Sciences as a scientific associate and to participate in the FlyFlect3D project. The knowledge and experience I have learned and gained from you over the past five years are tremendous. I am grateful for your supportive encouragement, expert insights and helpful discussions.

My greatest thanks also go to my PhD supervisor Prof. Dr. Tomas Sauer for having accepted me as a PhD student at the University of Passau in the Faculty of Mathematics and Computer Science. I thank you for offering me the opportunities to participate in the seminars in different thematic groups and for the promising and productive discussions with you. Your enthusiasm for mathematics has always been an inspiration for me.

Huge thanks to the people I worked with: Dr. Michael Strohmeier, Simon Hartel for the numerous discussions, supports and inspirations. I would also like to thank Dr. Alexander Zimmermann for the help in optimizing the algorithm and the discussions. I don't think I would have achieved these results in this project without the supports from all of you.

I also thank the financial support of the Bavarian Research Foundation (BFS, AZ-1273-17) and Micro-Epsilon Messtechnik GmbH & Co. KG. I thank Micro-Epsilon Messtechnik GmbH & Co. KG for providing the PMD setup and all the helping hands. I would like to thank the Bavarian Science Forum (BayWiss) for their generous scholarship and the offer for a wide variety of workshops.

Thank you mom, dad and my brother for standing firmly behind me whenever I look back. You have supported me all the way to this day. My dearest love goes to my husband Li Ang. Thank you for always being by my side in all the ups and downs, sharing my pain and joy.

Thank you all.

Anhang A

Appendix

A.1 Calculation of R_2

According to (4.37), $R_2 = \int \frac{1}{2} b(x) e^{i\phi(x)} \psi^*_{a,b}(x) dx$, with:
$\phi(x) = \phi(b) + \phi^{'}(b)(x-b) + \frac{\phi^{''}(b)}{2!}(x-b)^2 + ...,$
and $\psi_{a,b}(x) = \frac{1}{a\sqrt{\pi F_b}} e^{i2\pi F_c(\frac{x-b}{a})} e^{-\frac{1}{F_b}(\frac{x-b}{a})^2}$.

Since $b(x)$ varies only slightly, $b(x)$ is considered as a constant b_0. R_2 is now rewritten as:
$R_2 = \frac{b_0}{2a\sqrt{\pi F_b}} \int e^{i\phi(b)} e^{i\phi^{'}(b)(x-b)} e^{i\frac{\phi^{''}(b)}{2}(x-b)^2} e^{-i2\pi F_c(\frac{x-b}{a})} e^{-\frac{1}{F_b}(\frac{x-b}{a})^2} dx.$
Combining the $(x-b)$ - terms and the $(x-b)^2$ - terms, it gives:
$R_2 = \frac{b_0}{2a\sqrt{\pi F_b}} e^{i\phi(b)} \int e^{-i2\pi(\frac{F_c}{a} - \frac{\phi^{'}(b)}{2\pi})(x-b)} e^{-(\frac{1}{F_b a^2} - i\frac{\phi^{''}(b)}{2})(x-b)^2} dx.$

Note that, the integration in R_2 is the Fourier transform of a complex Gaussian function. Therefore, it can be rewritten as:
$\int e^{-i2\pi(\frac{F_c}{a} - \frac{\phi^{'}(b)}{2\pi})(x-b)} e^{-(\frac{1}{F_b a^2} - i\frac{\phi^{''}(b)}{2})(x-b)^2} dx$
$= \sqrt{\frac{\pi}{\alpha_r}} e^{-\frac{(\pi f_r)^2}{\alpha_r}},$
with $\alpha_r = \frac{1}{F_b a^2} - i\frac{\phi^{''}(b)}{2}$, and $f_r = \frac{F_c}{a} - \frac{\phi^{'}(b)}{2\pi}$.

$$\sqrt{\frac{\pi}{\alpha_r}}$$

$$= \sqrt{\frac{2\pi F_b a^2}{2 - iF_b a^2 \phi''(b)}}$$

$$= \sqrt{\frac{2\pi F_b a^2 (2 + iF_b a^2 \phi''(b))}{4 + F_b^2 a^4 \phi''^2(b)}}$$

$$= \sqrt{\frac{2\pi F_b a^2}{(4 + F_b^2 a^4 \phi''^2(b))^{\frac{1}{2}}}} e^{i\frac{\theta}{2}}, \text{ with } \theta = \arctan \frac{F_b a^2 \phi^{''}(b)}{2}.$$

$$-\frac{(\pi f_r)^2}{\alpha_r}$$

$$= -\frac{\pi^2 (\frac{F_c}{a} - \frac{\phi^{'}(b)}{2\pi})^2}{\frac{1}{F_b a^2} - i\frac{\phi^{''}(b)}{2}}$$

$$= -\frac{\frac{1}{2} F_b a^2 (2\pi \frac{F_c}{a} - \phi^{'}(b))^2}{4 + F_b^2 a^4 \phi''^2(b)} (2 + iF_b a^2 \phi^{''}(b))$$

Then,

$$R_2 = \frac{b_0}{\sqrt{2}(4 + F_b^2 a^4 \phi''^2(b))^{\frac{1}{4}}} e^{-\frac{F_b a^2 (2\pi \frac{F_c}{a} - \phi^{'}(b))^2}{4 + F_b^2 a^4 \phi''^2(b)}} e^{i(\phi(b) + \frac{\theta}{2} - \frac{\frac{1}{2} F_b^2 a^4 \phi^{''}(b)(2\pi \frac{F_c}{a} - \phi^{'}(b))^2}{4 + F_b^2 a^4 \phi''^2(b)})}.$$

Let

$M = a^2 F_b \phi''(b)$,

$N_2 = F_b (2\pi F_c - a\phi'(b))^2$,

$\theta = \arctan(\frac{M}{2})$,

then

$$R_2 = \frac{b_0}{\sqrt{2}(4 + M^2)^{\frac{1}{4}}} e^{-\frac{N_2}{4 + M^2}} e^{i(\phi(b) + \frac{\theta}{2} - \frac{N_2 M}{2(4 + M^2)})}.$$

Let $N_3 = F_b (2\pi F_c + a\phi'(b))^2$,

then $R_3 = \frac{b_0}{\sqrt{2}(4 + M^2)^{\frac{1}{4}}} e^{-\frac{N_3}{4 + M^2}} e^{-i(\phi(b) + \frac{\theta}{2} - \frac{N_3 M}{2(4 + M^2)})}$.

A.2 $\mathbf{B}(t, \mathbf{r}, \lambda) = \frac{\partial \mathbf{F}(t, \mathbf{r}, \lambda)}{\partial t}$ represented in the principal coordinate system

According to Eq.5.3, **B** can be represented as $\begin{bmatrix} \frac{\partial \hat{\mathbf{x}}_r + \partial \hat{\mathbf{x}}_i + \partial \lambda \nabla g(\mathbf{r})}{\partial t} \\ \frac{\partial g(\mathbf{r})}{\partial t} \end{bmatrix}$, since

$\frac{\partial g(\mathbf{r})}{\partial t}=0$, $\frac{\partial \hat{\mathbf{x}}_r}{\partial t}=0$ and $\frac{\partial[\lambda\nabla g(\mathbf{r})]}{\partial t}=0$, therefore, **B** can be written as $\begin{bmatrix}\frac{\partial\hat{\mathbf{x}}_i}{\partial t}\\ 0\end{bmatrix}$.

$$\begin{aligned}
\frac{\partial\hat{\mathbf{x}}_i}{\partial t} &= \frac{\partial\frac{\mathbf{r}-\mathbf{r}_s(t)}{|\mathbf{r}-\mathbf{r}_s(t)|}}{\partial t}\\
&= \frac{\partial(\mathbf{r}-\mathbf{r}_s(t))}{\partial t}\frac{|\mathbf{r}-\mathbf{r}_s(t)|}{|\mathbf{r}-\mathbf{r}_s(t)|^2}-\frac{\partial|\mathbf{r}-\mathbf{r}_s(t)|}{\partial t}\frac{\mathbf{r}-\mathbf{r}_s(t)}{|\mathbf{r}-\mathbf{r}_s(t)|^2}\\
&= \frac{\partial(\mathbf{r}-\mathbf{r}_s(t))}{\partial\mathbf{r}_s(t)}\frac{\partial\mathbf{r}_s(t)}{\partial t}\frac{1}{|\mathbf{r}-\mathbf{r}_s(t)|}-\frac{\hat{\mathbf{x}}_i}{|\mathbf{r}-\mathbf{r}_s(t)|}\frac{\partial|\mathbf{r}-\mathbf{r}_s(t)|}{\partial(\mathbf{r}-\mathbf{r}_s(t))}\frac{\partial(\mathbf{r}-\mathbf{r}_s(t))}{\partial t}\\
&= -\frac{\delta\mathbf{r}_s}{|\mathbf{r}-\mathbf{r}_s(t)|}+\frac{\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T\delta\mathbf{r}_s}{|\mathbf{r}-\mathbf{r}_s(t)|}\\
&= -\frac{(\mathbf{I}-\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T)\delta\mathbf{r}_s}{|\mathbf{r}-\mathbf{r}_s(t)|}.\ \text{(Eq.(21) in [SCP05])}
\end{aligned}$$

Applying Eq. 5.7 and representing **B** in [$\hat{\mathbf{u}}$, $\hat{\mathbf{v}}$, $\hat{\mathbf{w}}$], $\mathbf{B}=\begin{bmatrix}B_u\\ B_v\\ B_w\\ 0\end{bmatrix}$, with

$B_u=\frac{\delta r_{s,w}\cos\theta\sin\theta-\delta r_{s,u}\cos^2\theta}{|\mathbf{r}_{s0}|}$;(Eq. (10) in [SCP05])
$B_v=-\frac{\delta r_{s,v}}{|\mathbf{r}_{s0}|}$; (Eq. (10) in [SCP05])
$B_w=\frac{\delta r_{s,u}\cos\theta\sin\theta-\delta r_{s,w}\sin^2\theta}{|\mathbf{r}_{s0}|}$. (Eq. (24) in [SCP05])

A.3 $\mathbf{J}(t,\mathbf{r},\lambda)=\frac{\partial\mathbf{F}(t,\mathbf{r},\lambda)}{\partial(\mathbf{r},\lambda)}$ represented in the principal coordinate system

According to Eq. 5.3, **J** can be written as

$$\frac{\partial\mathbf{F}(t,\mathbf{r},\lambda)}{\partial(\mathbf{r},\lambda)}=\begin{bmatrix}\frac{\partial\hat{\mathbf{x}}_r+\partial\hat{\mathbf{x}}_i+\partial\lambda\nabla g(\mathbf{r})}{\partial\mathbf{r}} & \frac{\partial\hat{\mathbf{x}}_r+\partial\hat{\mathbf{x}}_i+\partial\lambda\nabla g(\mathbf{r})}{\partial\lambda}\\ \frac{\partial g(\mathbf{r})}{\partial\mathbf{r}} & \frac{\partial g(\mathbf{r})}{\partial\lambda}\end{bmatrix}=\begin{bmatrix}\frac{\partial\hat{\mathbf{x}}_r+\partial\hat{\mathbf{x}}_i+\partial\lambda\nabla g(\mathbf{r})}{\partial\mathbf{r}} & \nabla g(\mathbf{r})\\ \nabla(g(\mathbf{r}))^T & 0\end{bmatrix}.$$

(Eq.(20) and Eq. (21) in [SCP05])

According to Eq.5.2, $\frac{\partial\hat{\mathbf{x}}_i}{\partial\mathbf{r}}$ can be expressed as

$$\begin{aligned}
&\frac{\partial\frac{\mathbf{r}-\mathbf{r}_s(t)}{|\mathbf{r}-\mathbf{r}_s(t)|}}{\partial\mathbf{r}}\\
&= \frac{\partial(\mathbf{r}-\mathbf{r}_s(t))}{\partial\mathbf{r}}\frac{|\mathbf{r}-\mathbf{r}_s(t)|}{|\mathbf{r}-\mathbf{r}_s(t)|^2}-\frac{\partial|\mathbf{r}-\mathbf{r}_s(t)|}{\partial\mathbf{r}}\frac{\mathbf{r}-\mathbf{r}_s(t)}{|\mathbf{r}-\mathbf{r}_s(t)|^2}\\
&= \frac{I}{|\mathbf{r}-\mathbf{r}_s(t)|}-\frac{\hat{\mathbf{x}}_i}{|\mathbf{r}-\mathbf{r}_s(t)|}\frac{\partial|\mathbf{r}-\mathbf{r}_s(t)|}{\partial(\mathbf{r}-\mathbf{r}_s(t))}\frac{\partial(\mathbf{r}-\mathbf{r}_s(t))}{\partial\mathbf{r}}
\end{aligned}$$

$= \frac{I}{|\mathbf{r}-\mathbf{r}_s(t)|} - \frac{\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T}{|\mathbf{r}-\mathbf{r}_s(t)|}$

$= \frac{\mathbf{I}-\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T}{|\mathbf{r}-\mathbf{r}_s(t)|}$,

and analogous $\frac{\partial\hat{\mathbf{x}}_r}{\partial\mathbf{r}} = \frac{\mathbf{I}-\hat{\mathbf{x}}_r\hat{\mathbf{x}}_r^T}{|\mathbf{r}-\mathbf{c}|}$.

Rewriting $\frac{\partial\lambda\nabla g(\mathbf{r})}{\partial\mathbf{r}} = \lambda\frac{\partial\nabla g(\mathbf{r})}{\partial\mathbf{r}}$ as $\lambda\mathbf{H}_g$ with

$\mathbf{H}_g = \begin{bmatrix} -a-eu-fv & -c-fu-gv & 0 \\ -c-fu-gv & -b-gu-hv & 0 \\ 0 & 0 & 0 \end{bmatrix}$, (Eq. (22) in [SCP05])

$\mathbf{J}$ can be represented by

$\mathbf{J} = \begin{bmatrix} \frac{\mathbf{I}-\hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T}{|\mathbf{r}-\mathbf{r}_s(t)|} + \frac{\mathbf{I}-\hat{\mathbf{x}}_r\hat{\mathbf{x}}_r^T}{|\mathbf{r}-\mathbf{c}|} + \lambda\mathbf{H}_g & \nabla g(\mathbf{r}) \\ (\nabla g(\mathbf{r}))^T & 0 \end{bmatrix}$. (Eq. (20) and Eq. (21) in [SCP05])

Note that $\lambda = 2cos\theta$. Considering the surface parameter up to the second order, $\mathbf{J}$ can therefore be written as:

$\begin{bmatrix} J_u - 2a\cos\theta & -2c\cos\theta & J_w & 0 \\ -2c\cos\theta & J_v - 2b\cos\theta & 0 & 0 \\ J_w & 0 & J_v\sin^2\theta & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$, (Eq.(23) in [SCP05])

with

$J_u = J_v\cos^2\theta$, (Eq. (11) in [SCP05])
$J_v = \frac{1}{s} + \frac{1}{|\mathbf{r}_{s0}|}$, (Eq. (11) in [SCP05])
$J_w = \frac{s-|\mathbf{r}_{s0}|}{|\mathbf{r}_{s0}|s}\cos\theta\sin\theta$. (Eq. (25) in [SCP05])

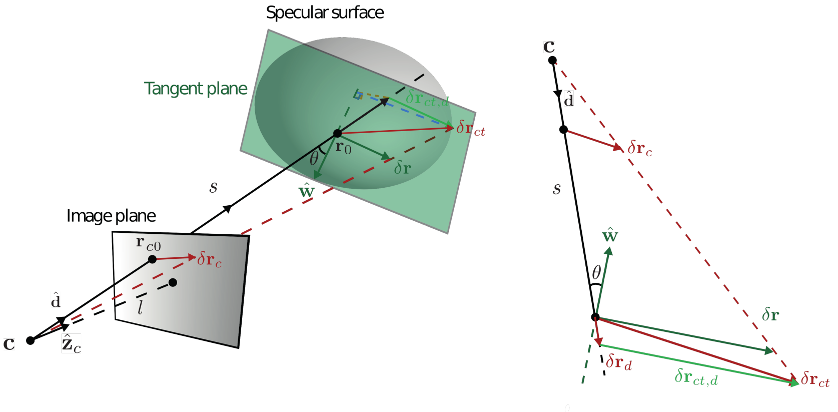


Fig. A.1: The perspective projection of $\delta \mathbf{r}_c$ on the camera plane onto $\delta \mathbf{r}$ on the tangent plane of the object surface.

A.4 The perspective projection of a vector on the camera plane onto the tangent plane of the object surface

$\hat{\mathbf{z}}_c$ is the camera axis and l is the distance between the projection center $\mathbf{c}$ and the camera plane. $\hat{\mathbf{d}}$ is the directional vector of a camera ray, which passes through point $\mathbf{r}_{c0}$ on the camera plane and lands on point $\mathbf{r}_0$ on the object surface. The distance between $\mathbf{c}$ and $\mathbf{r}_{c0}$ is therefore

$$d_{\mathbf{c}\rightarrow\mathbf{r}_{c0}} = \frac{l}{\langle \hat{\mathbf{z}}_c, \hat{\mathbf{d}} \rangle}. \tag{A.1}$$

$\delta \mathbf{r}_c$ is a vector beginning from $\mathbf{r}_{c0}$ and lying on the camera plane. Shifting $\delta \mathbf{r}_c$ with its beginning from $\mathbf{r}_{c0}$ to $\mathbf{r}_0$ along $\hat{\mathbf{d}}$ under the perspective projection in the respect of $\mathbf{c}$ and denoting the shifted vector as $\delta \mathbf{r}_{ct}$, it gives

$$\delta \mathbf{r}_{ct} = \frac{s\langle \hat{\mathbf{z}}_c, \hat{\mathbf{d}} \rangle}{l} \delta \mathbf{r}_c. \tag{A.2}$$

Denote a vector on the $\hat{\mathbf{d}}$ axis starting from $\mathbf{r}_0$ as $\delta\mathbf{r}_d = \alpha\hat{\mathbf{d}}$ with α being a scaling factor, which allows $\delta\mathbf{r}_{ct,d} = \delta\mathbf{r}_{ct} - \delta\mathbf{r}_d$ to be parallel to the tangent plane. It follows that the projection of $\delta\mathbf{r}_{ct}$ and $\delta\mathbf{r}_d$ on the $\hat{\mathbf{w}}$ axis (the normal of the tangent plane) are equal:

$$\langle\delta\mathbf{r}_{ct}, \hat{\mathbf{w}}\rangle = \langle\delta\mathbf{r}_d, \hat{\mathbf{w}}\rangle = \langle\alpha\hat{\mathbf{d}}, \hat{\mathbf{w}}\rangle. \tag{A.3}$$

α can be therefore written as

$$\alpha = \frac{\hat{\mathbf{w}}^T}{\langle\hat{\mathbf{d}}, \hat{\mathbf{w}}\rangle}\delta\mathbf{r}_{ct}, \tag{A.4}$$

and it follows

$$\delta\mathbf{r}_d = \alpha\hat{\mathbf{d}} = \frac{\hat{\mathbf{d}}\hat{\mathbf{w}}^T}{\langle\hat{\mathbf{d}}, \hat{\mathbf{w}}\rangle}\delta\mathbf{r}_{ct}. \tag{A.5}$$

Since the norm of $\delta\mathbf{r}_d$ is negligible compared to the s - the distance between $\mathbf{c}$ and $\mathbf{r}_0$, $\delta\mathbf{r}_{ct,d}$ can be approximated as $\delta\mathbf{r}$ - the The perspective projection of $\delta\mathbf{r}_c$ on the camera plane onto the tangent plane. Therefore,

$$\delta\mathbf{r} \approx \delta\mathbf{r}_{ct,d} = \delta\mathbf{r}_{ct} - \delta\mathbf{r}_d = \frac{s}{l}\langle\hat{\mathbf{z}}_c, \hat{\mathbf{d}}\rangle(I - \frac{\hat{\mathbf{d}}\hat{\mathbf{w}}^T}{\langle\hat{\mathbf{d}}, \hat{\mathbf{w}}\rangle})\delta\mathbf{r}_c. \tag{A.6}$$

A.5 Homogeneous Matrix

The homogeneous matrix is applied for affine transformation including translation, scaling, rotation and shearing. To transform a 3-vector $[x\ y\ z]^T$ presented in the original coordinate system XYZ into a new coordinate system $X'Y'Z'$, the homogeneous matrix $\mathbf{H}$ is a $4 \cdot 4$ matrix and can be written as:

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}, \tag{A.7}$$

where $\mathbf{t} = [t_x\ t_y\ t_z]^T$ is the translation vector (the origin of the coordinate system XYZ presented in $X'Y'Z'$), and the scaling, rotation and shearing are all included in the $3 \cdot 3$ matrix $\mathbf{R}$. In the application of coordinate transformation discussed in this thesis, only rotation and translation are applied. Therefore, the $\mathbf{R}$ matrix is a (orthogonal) rotation matrix here and can be written as:

$$\mathbf{R} = [\hat{\mathbf{x}}\ \hat{\mathbf{y}}\ \hat{\mathbf{z}}], \tag{A.8}$$

where $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$ and $\hat{\mathbf{x}}$ denote the x-, y- and z-axis of the original coordinate XYZ presented in the new coordinate $X'Y'Z'$.

To apply the homogeneous transformation, the $3D$ vector $[x\ y\ z]^T$ muss be extended to $4D$ by adding the 4th axis with 1 and the vector $[x\ y\ z]^T$ presented in $X'Y'Z'$ is obtained by

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}. \tag{A.9}$$