
Solving ambiguity problems in phase based profilometry

M.Sc. Thesis

E. Schippers

University of Twente
Department of Electrical Engineering,
Mathematics & Computer Science (EEMCS)
Signals & Systems Group (SAS)
P.O. Box 217
7500 AE Enschede
The Netherlands

Report Number: SAS 01-10
Report Date: 15/01/2010
Period of Work: 01/04/2009 – 15/01/2010
Thesis Committee: Prof. Dr. ir. C.H. Slump
Dr. ir. F.van der. Heijden
Dr. ir. L.J. Spreeuwers

Abstract

This report is the result of a master thesis assignment at the Signals and Systems Group, University of Twente. Half of this project focuses on the creation of a development platform, which can be used for research on structured light systems. In the other half the platform is used to experimentally test a structured light approach called phase based profilometry.

The platform hardware was already available at the start of the project and consists of a digital projector, a digital photo camera and a stable rig. In this assignment, software was developed to enable accurate calibration of the devices, both geometrically and radiometrically. The software is created as a Matlab toolbox.

In phase based profilometry a projector is used to spatially modulate a light source with a periodic function. This light is used to illuminate an object of interest of which the 3D structure is required. Depending on the nature of the projected function, the phase of the function can be estimated by observing the object with a camera and analysing the deformed and shifted patterns.

This project concludes with the unambiguous 3D reconstruction of a scene at a distance of 1 meter with a 2mm standard deviation in depth, using two projections of a high frequency sine wave. The frequency has a low component in the phase direction (comparable with the epipolar line in stereo imaging). Due to this component, the distance between ambiguous solutions can be kept large. Thanks to fact that the component in the orthogonal direction is high, basic phase estimation schemes can still function properly.

Acknowledgements

A year and two months ago I started my internship at Fugro Intersite. Besides discovering that there was life after college, I gained experience in stereo vision and camera calibration. These experiences proved very useful during my master thesis.

Even though Fugro provided a master thesis assignment possibility as well, I decided to move to Enschede to celebrate my last months as a student with my roommates at the Calslaan. Thankfully this did not mean I had to permanently leave my colleagues in Leidschendam, as I will rejoin them professionally not long after completing this thesis.

At first difficult, multiple view geometry intrigued me and so did structured light when introduced to me by Ferdi van der Heijden, somewhere early 2009. Together with Luuk Spreeuwiers he formed the team to supervise me during my master thesis.

I thank Ferdi and Luuk as they did not force me to make the choice between research and development. I am very happy that I was able to put the mathematical research into practice by building a so-called development platform for structured light research. Keeping in touch with reality has always been very important to me. Also their confidence and useful hints and thoughts during meetings kept me going and motivated.

I do not thank my fellow students at the Signals and Systems chair for introducing me to and dragging me into playing “Achtung die Kurve”. The horrible computer game cost me up to several minutes per week of my precious time. However, they made up by being great company during coffee and lunch breaks, movie nights, trips to Cologne and even to the gravitational centre of the Netherlands – thanks after all!

Finally I thank my parents and my sister for supporting me unconditionally for all those years. Thanks to their faith in me I never even considered giving up.

Table of Contents

Abstract	i
Acknowledgements	iii
Table of Contents	v
1 Introduction	1
1.1 Project description.....	1
1.2 Outline.....	1
2 Structured light systems	5
2.1 Overview.....	5
2.1.1 One shot 3D reconstruction using instantaneous frequencies.....	6
2.1.2 One shot structured light range imaging using particle filters.....	7
2.1.3 Phase based profilometry.....	7
2.2 Camera model.....	8
2.2.1 Geometric model.....	8
2.2.2 Radiometric model.....	9
2.2.3 Colour images and Bayer tiles.....	11
2.3 Projector model.....	11
2.3.1 Geometric model.....	11
2.3.2 Radiometric model.....	12
2.4 3D reconstruction.....	13
2.4.1 Depth from corresponding points.....	13
2.4.2 Error propagation and device positions.....	15
2.5 Scenic influences.....	15
2.5.1 Additional light sources.....	16
2.5.2 Object texture.....	16
3 Development platform design and calibration	17
3.1 Available Hardware.....	17
3.1.1 Camera.....	17
3.1.2 Projector.....	17
3.1.3 LUX meter.....	17
3.1.4 Rig.....	18
3.1.5 Test objects.....	18
3.2 Geometric calibration.....	19
3.2.1 Camera calibration.....	19
3.2.2 Stereo calibration.....	21
3.2.3 Camera-Projector calibration.....	21
3.2.4 Calibration procedure.....	25
3.2.5 Calibration result example.....	25
3.2.6 Consistency check.....	26
3.3 Radiometric calibration.....	27
3.3.1 Linearity measurements.....	27
3.4 Pixel mapping and 3D reconstruction.....	29
3.4.1 Pixel mapping.....	29
3.4.2 Effects of inter reflections.....	32
3.4.3 Stereo triangulation and 3D reconstruction.....	32
3.5 Coordinate rectification.....	35

3.6	Implementation	37
3.7	Conclusion.....	37
4	Phase based profilometry	39
4.1	Introduction	39
4.2	Phase and orthogonal phase component	41
4.2.1	Image prediction	41
4.2.2	Projected function	41
4.2.3	Instantaneous frequency in x - direction	42
4.2.4	Phase in the y -direction	43
4.2.5	Phase and orthogonal direction.....	44
4.2.6	Phase in orthogonal direction	45
4.3	Phase demodulation	46
4.4	Hypothesis	47
4.5	Summary.....	47
5	Experiments	49
5.1	Experiment description	49
5.1.1	Test scene requirements	50
5.1.2	Picking the right solution	51
5.1.3	Experiment procedure	52
5.2	Measurements and results	54
5.2.1	Calibration result.....	54
5.2.2	Reference depth map	54
5.2.3	Examples of observed signals.....	54
5.2.4	Phase direction reconstruction results	56
5.2.5	Trouble using horizontal phase estimation.....	57
5.2.6	Final results.....	59
5.3	Comparison and discussion	60
5.3.1	Comparison with reference depth map.....	60
5.4	Conclusion.....	64
6	Conclusion	65
6.1	Achievements	65
6.2	Conclusions.....	65
6.3	Recommendations.....	66
6.3.1	Development platform	66
6.3.2	New phase based profilometry approach	67
A	Appendix A - Automatic corner finder	69
B	Appendix B – Structured Light Toolbox for Matlab Manual	71
B.1	Toolbox setup.....	71
B.2	Geometric calibration and coordinate rectification	72
B.3	Pixel mapping procedure and 3D reconstruction	73
B.4	Radiometric calibration	74
B.5	Additional functions	75
	Bibliography	77

1 Introduction

1.1 Project description

A structured light system consists of a projector and one or more cameras. In previous work, Berendsen [1] and Nijmeijer [2] used such a system to generate a depth map of an object from a certain point of view. They projected a known pattern onto an object and analysed the resulting image of the object. The work of Nijmeijer resulted in ambiguous solutions and Berendsen handled these by applying a particle smoother. Although this still did not fully solve the ambiguity problem, the smoother is at least able to reveal all possible ambiguous solutions. The work of Berendsen is not yet properly confirmed by experiments. To do so, the radiometric properties of the objects need to be considered and more practical details need to be taken into account.

These practical problems emphasise the need for a professional development platform for structured light systems. Therefore, the first part of the project is to deliver such a platform. It will help to analyse practical issues, compensate for distortions, validate proposed algorithms experimentally and determine the accuracy of new algorithms.

This platform will subsequently be used in the second part of the assignment to continue with the development of a one shot 3D reconstruction method. Phase based profilometry will be examined. The basic approach is to project a fringe pattern in the phase direction and analyse it in the same direction. The phase direction can be compared with the epipolar direction in terms of stereo vision. This thesis claims that projecting a sinusoid pattern in a different angle than (but not orthogonal to) the phase direction can increase the accuracy of the system, while maintaining or even increasing the space between ambiguous solutions.

1.2 Outline

As described in the project description, this thesis is separable in two parts; the design of a development platform and the development of a one shot 3D reconstruction method using structured light. The chronological order of the two parts is obvious as the development platform needs to be ready in order to be able to create and analyse a new structured light approach.

Chapter two gives an overview of the structured light system and presents relevant literature on existing structured light 3D reconstruction methods and profilometry. It introduces the models of the system and several variables are defined. Also the influences of the environment on the platform will be discussed.

Chapter three focuses on the development platform. First the design choices of the physical platform are substantiated. The main part of this chapter describes the calibration procedures. Finally an independent, multiple-shot structured light method is discussed that enables the user to verify his reconstruction results.

The fourth chapter illuminates the main research subject of this thesis; phase based profilometry. The projector and camera are assumed to be aligned parallel with a relative translation only in the y -direction. By means of coordinate rectification using the calibration by the development platform, this parallel alignment is possible.

When a sinusoid pattern $P(x, y) = \cos(\omega_o x + \omega_p y)$ is projected onto the scene, the observed phase can be used to estimate depth, but only if $\omega_p \neq 0$. This is illustrated in Figures 1.1 to 1.3. The y -direction is in this case the phase direction. The thesis claims that the frequency component ω_o has no influence on the sensitivity of the depth estimate w.r.t. phase estimation errors and has no influence on the distance between ambiguous solutions. By picking a low ω_p , the space between ambiguous solutions can be made large, while the phase estimation can still function properly thanks to a high ω_o . These claims are also elaborated in chapter four.

To confirm the claims, the development platform is used to do several experiments. Chapter five describes the experiments, presents the experimental results and discusses their meaning.

Finally the report is concluded by chapter six. The claims are briefly summarized as well as compared with the experimental data in a discussion. The work done in this project lead to new insights, dos and don'ts. The chapter includes this information as a summation of recommendations for future research.

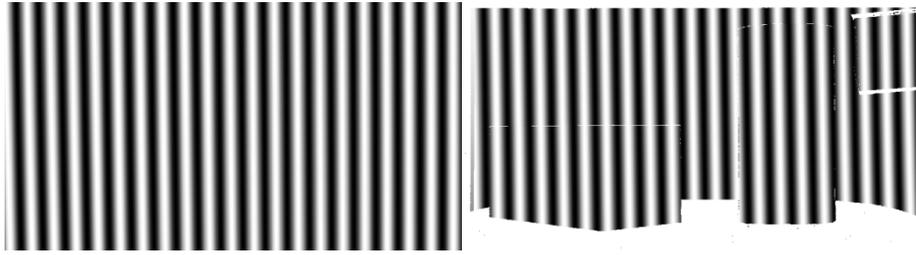


Figure 1.1 - In the projected pattern $P(x, y) = \cos(\omega_o x + \omega_p y)$, $\omega_p = 0$. Despite the presence of objects, the observed phase cannot be used to estimate depth. Left; the projected pattern. Right; the observed image.

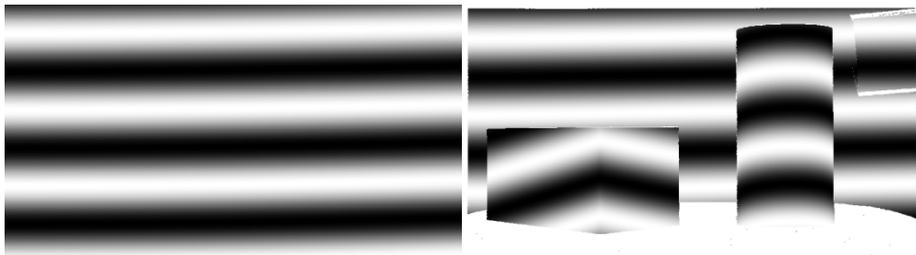


Figure 1.2 - In the projected pattern $P(x, y) = \cos(\omega_o x + \omega_p y)$, $\omega_o = 0$. The observed phase is distorted by the depth of the observed surfaces. Thanks to the low frequency, the objects can be unambiguously reconstructed, but the phase will be estimated poorly near edges.

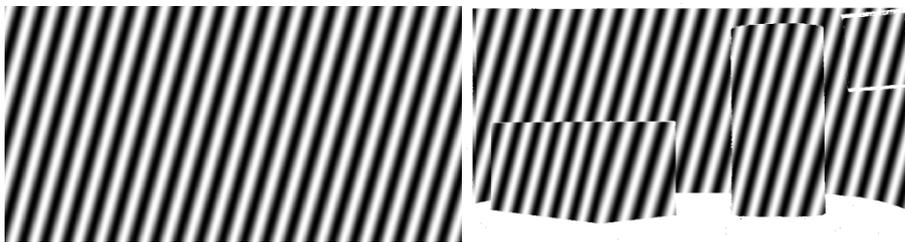


Figure 1.3 - In the projection $P(x, y) = \cos(\omega_o x + \omega_p y)$, neither of the frequency components is zero. ω_o and ω_p are as in the examples of Figure 1.1 and Figure 1.2 respectively. Due to the relatively high frequencies that are observed, the phase is disturbed less by edges and albedo than in the example in Figure 1.2.

2 Structured light systems

This chapter gives an introduction to structured light systems. The first section briefly describes existing methods. Sections 2.2 and 2.3 discuss the models of the two main components of a structured light system; the camera and projector. Subsequently, the mathematics on 3D reconstruction is introduced. Finally, in section 2.5 the main influences of the observed scene are discussed, like ambient light and surface albedo.

2.1 Overview

A structured light system can be compared with a stereo vision system. In stereo vision two cameras observe an object from a different point of view. Features on the object that can be detected in both images – corresponding points – can be used in a triangulation scheme to establish an estimate on the 3D position of that point. When enough corresponding points are available, a 3D point cloud can be created, which enables one to 3D reconstruct the observed object.

In a structured light system, a camera is replaced by a light source that can spatially modulate the emitted light rays. A range of devices can be used. A laser can be used to either emit one ray or a laser stripe. A slide projector or digital projector can be used to project an entire field of rays. As in a stereo camera setup, corresponding points are needed to triangulate and measure a 3D position. Somehow, the camera must be able to use the observed intensity at a certain image location to estimate by which ray from the light source that point was illuminated. In [3] a lot of codification strategies in order to do so are presented.

Three main strategies can be distinguished; time-multiplexing, spatial neighbourhood coding and direct coding. Time-multiplexing requires a static scene or object. By acquiring multiple images while projecting different binary patterns, each projected ray – in a digital projector each pixel modulates a ray – can be given a binary code, one bit per image. One such a system is described in [4] and is implemented as a reference method. Section 3.4 will go into more detail on this method.

Spatial neighbourhood coding will require only a single image and can thus be used to reconstruct moving objects. The prerequisite is that only continuous surface patches can be reconstructed, since the neighbourhood of the observed intensity value is needed to decipher which ray illuminated the observed spot.

A naive approach would be projecting a gradient in greyscales and translating the observed intensities directly the corresponding projector coordinates. This is called direct coding. It is of course very sensitive to noise, texture, ambient light and the surface slope towards the camera. Colour coding [5] can be used to

be less sensitive to amplitude changes, or the gradients can be projected periodically to limit the error. The latter option, however, introduces ambiguity.

Fourier Transform Profilometry (FTP) is a structured light method that is more like the method examined in this research. [6] introduces the basics of the method and illuminates some of the used algorithms. A sinusoidal grating is projected onto a reference plane. In a second shot an object is placed on the reference plane. The scan lines of the observed image are Fourier transformed. Using the Fourier transforms of the reference image and the image with the object included, a phase difference can be obtained that holds the information on the object height. Unwrapping is needed to reconstruct the objects without jumps due to ambiguous phase jumps. Objects with discontinuities in height are thus difficult to properly reconstruct.

[7] combines FTP with a colour coding scheme to perform one shot reconstruction without ambiguity. It therefore can handle discontinuous heights. However, only simulated results are presented.

In the following subsections the work at the University of Twente on the development of a one shot 3D reconstruction method is discussed.

2.1.1 One shot 3D reconstruction using instantaneous frequencies

At the University of Twente at the chair "Signals and Systems," Nijmeijer was the first in the development of a one shot 3D reconstruction structured light system [2].

The first approach was to use the instantaneous frequency to estimate the depth of the scene. A projector at infinity was assumed is projecting vertical parallel lines. In a camera image, the distance between observed lines decreases as the distance of the camera to the observed surface increases. This spawned the idea of using instantaneous frequencies to reconstruct depth.

However, the slope of the observed surface influences the observed frequency as well as the depth. This implies there is no direct relation between the frequency and the depth, so, multiple, ambiguous solutions are possible.

A solution was found by measuring the derivative of instantaneous frequency, thereby acquiring an extra equation to solve for the slope of the surface. Since this estimation of the instantaneous frequency itself is already based on the derivative of the observed phase, the method is very sensitive to noise and distortions.

The project concluded with a simulation that validates the models for a 2D case, but experimental results were noisy and could not be used for reconstruction, as there were yet no means to calibrate the system.

2.1.2 One shot structured light range imaging using particle filters

Continuing the pursuit of a one shot reconstruction method, Berendsen [1] used a new approach by using a particle filter that estimates the depth and slope of the scene.

The observed intensity values are compared with the projected intensities. By means of a particle filter, the next corresponding position in the projector can be estimated and updated using an intensity measurement.

The particle filter is able to highlight the ambiguous solutions when a repeating pattern is projected. Especially the simulated results are convincing and can be used to see what kind of ambiguous solutions are generated by different types of patterns and frequencies.

Due to the influence of background illumination, inter reflections and other phenomena, the method could not be experimentally proven. Neither could real scenes be reconstructed due to the lack of calibration parameters.

2.1.3 Phase based profilometry

In this research, and to be precise in chapter four, a reconstruction method based on phase estimation of a sinusoid pattern is developed. This approach on structured light systems can be grouped with other phase based profilometry methods. More on existing phase based profilometry is discussed in section 4.1 while the rest of that chapter introduces the method that was implemented and tested here at the “Signals and Systems” group at the University of Twente.

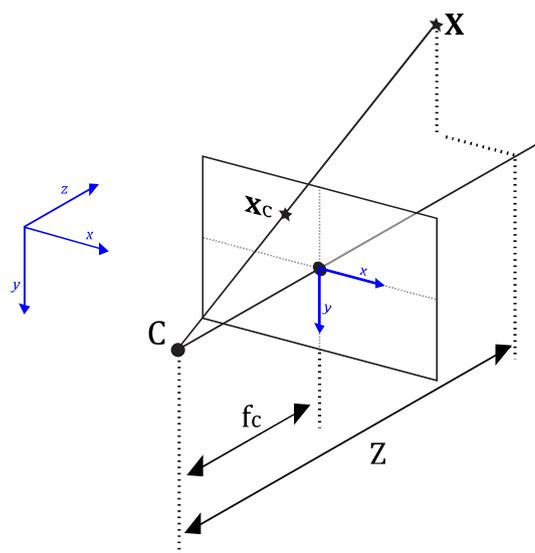


Figure 2.1 – Pinhole model of a camera. 3D coordinate X is projected onto the camera image plane at x_c . The camera centre C is also the origin of the world coordinate system. A ray perpendicular to the image plane through C is called the optical axis. The intersection of the optical axis and the image plane is called the principal point. This point is also the origin of the 2D coordinates on the image plane.

2.2 Camera model

2.2.1 Geometric model

As described in the first paragraph of this chapter the structured light system is modelled as a stereo camera system. A widely acknowledged model for a camera is the pinhole model with radial and tangential lens distortions. An illustration of this model is presented in Figure 2.1. An example is shown where a 3D coordinate is projected onto the image plane. The 3D coordinate \mathbf{X} is defined as $\mathbf{X} = [X, Y, Z]^T$. The coordinate of the projected point on the image plane can be computed as

$$\mathbf{x}'_c = \mathbf{K}_c \mathbf{X} \quad 2.1$$

where \mathbf{x}'_c is a 2D coordinate in the homogeneous form $\mathbf{x}'_c = [\alpha x'_c, \alpha y'_c, \alpha]^T$ and

$$\mathbf{K}_c = \begin{bmatrix} f_{cx} & & \\ & f_{cy} & \\ & & 1 \end{bmatrix} \quad 2.2$$

is called the camera calibration matrix. In this matrix the variables f_{cx} and f_{cy} represent the focal distance. Usually $f_{cx} = f_{cy}$, but in case of an asymmetric lens they can vary.

The acquired coordinate \mathbf{x}'_c is the ideal pinhole coordinate. In reality lens distortion will cause the 3D coordinate \mathbf{X} to be projected somewhere else. [8] presents a model for the radial and tangential distortion. The coordinate where \mathbf{X} will actually be projected on the image plane can be described as

$$\begin{aligned} x_c &= x'_c + \delta x_c^{(r)} + \delta x_c^{(t)} \\ y_c &= y'_c + \delta y_c^{(r)} + \delta y_c^{(t)} \end{aligned} \quad 2.3$$

with the contributions of radial distortion

$$\begin{aligned} \delta x_c^{(r)} &= x'_c (k_{c1} r^2 + k_{c2} r^4 + k_{c3} r^6) \\ \delta y_c^{(r)} &= y'_c (k_{c1} r^2 + k_{c2} r^4 + k_{c3} r^6) \end{aligned} \quad 2.4$$

and the contributions of tangential distortion

$$\begin{aligned} \delta x_c^{(t)} &= 2p_{c1} x'_c y'_c + p_{c2} (r^2 + 2x'_c) \\ \delta y_c^{(t)} &= p_{c1} (r^2 + 2y'_c) + 2p_{c2} x'_c y'_c \end{aligned} \quad 2.5$$

where $r = \sqrt{(x'_c)^2 + (y'_c)^2}$.

The coordinate \mathbf{x}_c is now the actual coordinate on the image plane. However, when a surface at the 3D coordinate emits a ray of light onto the image plane, it actually hits a sensor array. The sensor that detects the light ray is addressed with a row and column value with respect to the upper left corner of the image array. So, actually

$$\begin{aligned}\tilde{x}_c &= D_x x_c + x_{co} \\ \tilde{y}_c &= D_y y_c + y_{co}\end{aligned}\tag{2.6}$$

with D_x and D_y scaling factors to change from metric to pixels and (x_{co}, y_{co}) the centre point of the sensor array, i.e. the pixel location where the optical axis intersects the sensor array.

Up to this point, the focal distance has a metric unit, e.g. millimetres. By changing the unit to “pixel width”, the scaling factors D_x and D_y are no longer needed. The actual metric measure of the focal distance is not required for 3D reconstruction. When this value is desired for other reasons, the distance between pixels should be looked up in the specifications of the used camera.

To summarize, this model has nine parameters: f_{cx} , f_{cy} , $k_{c1\dots c3}$, $p_{c1\dots c2}$, x_{co} and y_{co} .

2.2.2 Radiometric model

The radiometric model concerns itself with the actual measured intensity values of the observed surface patches. Each camera pixel receives a certain amount of photons during its exposure. These photons are a portion of the number of photons that were emitted or reflected towards the camera centre. Instead of the number of photons the usual approach is to talk about power and energy.

Ideally, a pixel will intercept the flux from the direction of the ray that belongs to the pixel. The camera lens will accumulate the flux in a certain direction over the lens area and focus all that energy onto a certain pixel location. However, the area of the lens and diaphragm differs when seen from a different angle. This effect is called “vignetting” or radial falloff and can be described by a coordinate specific damping factor $M(x, y)$ of the incoming power.

For a thin lens model, vignetting can be modelled by a \cos^4 law [9], however, most cameras are built using more lenses and undergo more types of vignetting such as pixel vignetting (due to the angular sensitivity of the photo sensors) and optical vignetting (due to the lens casing and diaphragm). An example of the latter is shown in Figure 2.2. [10] states that the vignetting effects can be modelled properly by a 6th order even polynomial:



Figure 2.2 - [10] Two images of a wall, taken with different diaphragms. The aperture of the lens changes with the angle of incidence.

$$M(x, y) = M(r) = 1 + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6 \quad 2.7$$

with $r = \sqrt{x^2 + y^2}$.

The irradiance at the pixel location is measured by a photo sensor. The irradiance at the sensor surface causes a current to flow. For the duration of the exposure, the sensor integrates the current over time and will produce a certain output. The actual observed image will then be

$$C(x, y) = \int_0^{t_s} M(r) \cdot \hat{C}(x, y) dt = t_s M(r) \cdot \hat{C}(x, y) \quad 2.8$$

with $\hat{C}(x, y)$ the ideal image that would represent the radiance of the light in the direction of the ray that belong to the image plane coordinate (x, y) .

Not taken into account is the sensitivity of the sensor to a certain wavelength of light, i.e. colour. A pixel can be fitted with a spectral filter to limit its sensitivity to a certain spectral band.

The response of a photo sensor is in principle linear. Because this makes images “look too harsh”, most camera manufactures implement techniques to soften the image. The most common method is called gamma correction. Depending on the type of the camera, this effect is applied inside the camera and should be compensated afterwards. Most professional cameras are capable of presenting the raw sensor data.

A final step in the image acquisition process is of course the quantisation in order to process the image digitally. The signal-to-quantisation noise ratio can be computed by the well known equation

$$10 \log \left(\frac{S}{N} \right)_q \approx 1.8 + 6n \quad 2.9$$

with n the number of bits. In case of digital images where n is typically 8 or larger, this ratio will be 50dB or larger.

2.2.3 Colour images and Bayer tiles

In the radiometric model the pixel sensitivity to a certain wavelength of light was omitted. Most cameras are capable of taking colour images. The most common way is the implementation of so-called Bayer tiles. In that case three types of pixels are placed on the sensor array which are all sensitive to a different spectral range (red, green and blue). The types of pixels are placed in groups of four pixels. Such a group is called a Bayer tile. Since there are three types of pixels, one type is represented twice as often; green. This choice is based on the human vision system, which is most sensitive to green.

To create a full resolution output at the pixel location of a red or blue pixel, the green component at that location is computed by means of interpolation of the neighbouring green pixels.

Very comprehensive schemes exist that generate beautiful images, but are physically incorrect. It is therefore not appropriate to let the commercial software of a camera handle this so-called demosaicing.

In this research the demosaicing is omitted at all. Only one pixel per Bayer tiles is used (the lower left green pixel) while the other pixels are simply decimated. This reduces the image resolution by a factor 4.

2.3 Projector model

2.3.1 Geometric model

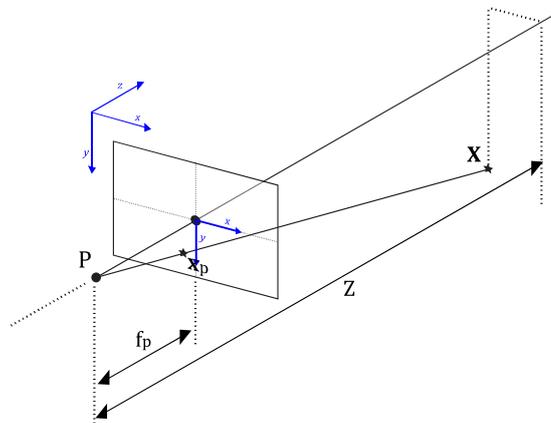


Figure 2.3 - Illustration of the pinhole model of the projector.

The projector more or less has the same configuration as a camera. Instead of receiving light on an image plane, an image from the projector plane is emitted into the world. Geometrically, the camera model described in section 2.2.1 is used to model the projector as well.

For clarity, the parameters of the projector model are equipped with a subscript p instead of a c for the camera. So we have f_{px} and f_{py} for the projector's focal

distance, $k_{p1..p3}$ and p_{p1}, p_{p2} for the projector lens distortion coefficients and \mathbf{x}_{p0} for the projectors centre point.

For the mathematical model we refer to section 2.2.1 where the camera model is elaborated.

2.3.2 Radiometric model

The projector is modelled by a pinhole model. This means that a point source of light is modulated by the projector image plane. This light modulation can be done by a Digital Micro mirror Device (DMD) or a Liquid Cristal Display (LCD). In the early days, slide projectors and overhead projectors have been used. Since the latter can only project one static image, their use is limited and is not discussed in this report.

The input of a projector consists of three colour values per pixel. The output should be a certain radiance for the rays that correspond to those pixels. Depending on the technology and the internal software settings of the projector, the input to output relation need not be linear:

$$P(x, y) = H(P_i(x, y)) \quad 2.10$$

in which $P_i(x, y)$ is the ideal projection and $H(I)$ is the projectors intensity response function. This response function is usually artificially implemented in the projector, which provides options to control gamma correction, white peaking, contrast, brightness and colour temperature. A general model is therefore beyond the scope of this research and correction should be performed by means of a look-up-table. This table can be created by a proper calibration scheme.

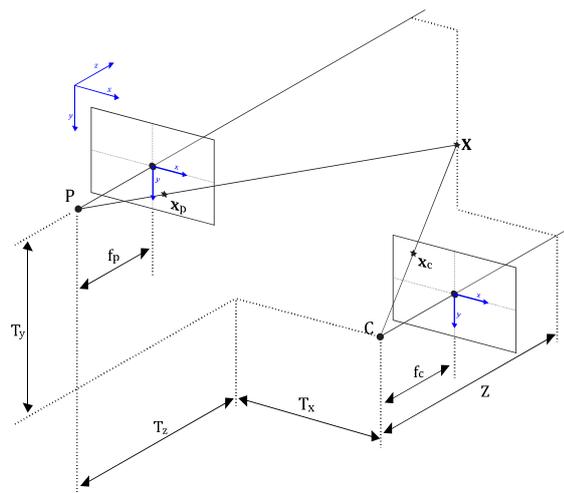


Figure 2.4 - Geometrical model of a structured light system. There is only a translation between projector and camera.

2.4 3D reconstruction

2.4.1 Depth from corresponding points

The camera and projector are considered to be oriented in the same direction (i.e. parallel setup). Only a translation \mathbf{T} is applied to the projector with respect to the camera. The situation is illustrated in Figure 2.4.

Every coordinate $\mathbf{x}_c = (\alpha x_p, \alpha x_c, \alpha)^T$ in the camera corresponds to a ray in space. This ray originates from a 3D coordinate $\mathbf{X} = (X, Y, Z)^T$ in space which in turn is lit by a projector ray modulated by the projector coordinate $\mathbf{x}_p = (\beta x_p, \beta y_c, \beta)^T$. The goal is to use the measured intensity value in the image at the specified camera coordinate to find out by which projector coordinate the observed point in space is lit. The coordinates are related to \mathbf{X} as follows:

$$\begin{aligned}\mathbf{x}_c &= \mathbf{K}_c \mathbf{X} \\ \mathbf{x}_p &= \mathbf{K}_p (\mathbf{X} - \mathbf{T})\end{aligned}\tag{2.11}$$

with

$$\mathbf{K}_c = \begin{bmatrix} f_{cx} & & \\ & f_{cy} & \\ & & 1 \end{bmatrix} \text{ and } \mathbf{K}_p = \begin{bmatrix} f_{px} & & \\ & f_{py} & \\ & & 1 \end{bmatrix}\tag{2.12}$$

\mathbf{x}_p and \mathbf{x}_c are related to each other through \mathbf{X} as follows:

$$\mathbf{x}_p = \mathbf{K}_p (\mathbf{K}_c^{-1} \mathbf{x}_c - \mathbf{T})\tag{2.13}$$

In this relation \mathbf{X} has disappeared and can only be found again by estimating \mathbf{x}_p for a known \mathbf{x}_c followed by solving the equation 2.11.

Because there are no rotations involved and the calibration matrices are diagonal, the computations for the x - and y -direction can be done separately. Equation 2.13 can now be split into

$$\begin{aligned}\alpha x_p &= f_{py} \cdot \left(\frac{\beta x_c}{f_{cx}} - T_x \right) \\ \alpha y_p &= f_{py} \cdot \left(\frac{\beta y_c}{f_{cy}} - T_y \right) \\ \alpha &= \beta - T_z\end{aligned}\tag{2.14}$$

while equation 2.11 dictates

$$\begin{aligned}
\beta x_c &= f_{cx} X \\
\beta y_c &= f_{cy} Y \\
\beta &= Z
\end{aligned}
\tag{2.15}$$

Now $\alpha = Z - T_z$ and equation 2.14 reduces to:

$$\begin{aligned}
(Z - T_z)x_{np} &= Zx_{nc} - T_x \\
(Z - T_z)y_{np} &= Zy_{nc} - T_y
\end{aligned}
\tag{2.16}$$

In which x_{nc}, y_{nc}, x_{np} and y_{np} are the normalized coordinates:

$$\begin{aligned}
x_{nc} &= \frac{x_c}{f_{cx}} \quad \text{and} \quad x_{np} = \frac{x_p}{f_{cy}} \\
y_{nc} &= \frac{y_c}{f_{py}} \quad \text{and} \quad y_{np} = \frac{y_p}{f_{py}}
\end{aligned}
\tag{2.17}$$

We can now establish the relation between x_{np} and x_{nc} which includes the depth of the point in space as observed by the camera at x_{nc} :

$$\begin{aligned}
x_{np} &= \frac{Zx_{nc} - T_x}{Z - T_z} \\
y_{np} &= \frac{Zy_{nc} - T_y}{Z - T_z}
\end{aligned}
\tag{2.18}$$

These functions allows us to predict the projector coordinates when the camera coordinates are known, as well as the 3D coordinate that links the two. The pairs x_{np}, x_{nc} and y_{np}, y_{nc} can both be used to estimate Z independently. This can be down by rewriting equation 2.18 into:

$$\begin{aligned}
Z &= \frac{T_z x_{np} - T_x}{x_{np} - x_{nc}} \\
Z &= \frac{T_z y_{np} - T_y}{y_{np} - y_{nc}}
\end{aligned}
\tag{2.19}$$

Now, with Z known, X and Y follow from equation 2.11 and 2.17:

$$\begin{aligned}
X &= x_{nc} \cdot Z \\
Y &= y_{nc} \cdot Z
\end{aligned}
\tag{2.20}$$

2.4.2 Error propagation and device positions

For both estimates the propagation of an error in x_{np} or y_{np} into an error in Z can be expressed as:

$$\frac{\delta Z}{\delta x_{np}} = \frac{(Z - T_z)^2}{T_x - x_{nc}T_z}$$

$$\frac{\delta Z}{\delta y_{np}} = \frac{(Z - T_z)^2}{T_y - y_{nc}T_z}$$
2.21

These error sensitivity figures need to be as low as possible. Small errors in the estimation of x_{np} or y_{np} should not cause large deviations in the estimate for Z . This should be kept in mind when choosing a relative translation for the two devices. A quick conclusion is that T_z should be chosen near the expected values for Z , i.e. close to the scene. On top of that, the difference between T_x (or T_y) and $x_{nc}T_z$ (or $y_{nc}T_z$) should be as large as possible. This implies that the object to reconstruct must not lay in the extend of the camera and projector centres. This is illustrated in Figure 2.5 which displays the error sensitivity as a function of Z and y_{nc} . For display purpose the logarithm of the error sensitivity is shown. The darker the intensity, the less sensitive Z will be for a measurement error in x_{np} . On the white line, which lays in the extend of P and C , the sensitivity is infinite.

A good choice for a relative position is thus to place one of the devices close to the scene and to make sure that in the x - or y -direction the object to reconstruct is in between the two devices, to prevent that part of the object is positioned in the extend of the camera and projector.

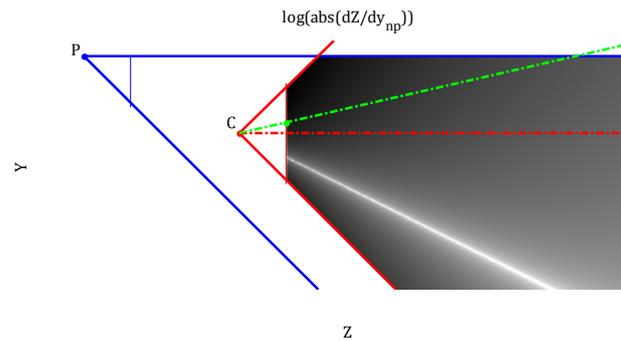


Figure 2.5 - A side view of a projector-camera setup. In the useful areas the intensity resembles the error sensitivity. The logarithm is shown for display purpose. The red dotted line covers the coordinates for $y_{nc} = 0$, the green dotted line indicates another choice for y_{nc} .

2.5 Scenic influences

In a structured light system the modulated light source, for example the projector, is ideally the only light source available. However, this is not always the case. Two types of additional lighting are important to be aware of and are discussed in the first subsection.

A second, scene and object dependant, influence is the texture of the observed surface. It can be imagined that a texture that is like the pattern that illuminates the scene will cause major problems when trying to analyse the observed surface patch. This will be discussed in the second subsection.

2.5.1 Additional light sources

As mentioned before, two types of additional lighting are important: ambient light and inter reflections.

Ambient light

Ambient light sources are sources of light that are not part of the modulated light source. An additional lamp or daylight might be such a source. In a multiple shot method, the ambient light can be detected by first observing the scene without the modulated light source and use that image as an offset. In a one shot method it is difficult to tell by what source an observed patch is illuminated. The observed intensity could differ due to added ambient light or a change in surface albedo.

Inter reflections

Inter reflections cause surface patches that are illuminated by the modulated light source to act as a light source themselves. Parts of the scene that would not have been lit by the modulated light source can then still be illuminated. Even in multiple shot methods, inter reflections can cause trouble since they cannot simply be subtracted like ambient light. The inter reflections differ for each projected pattern and thus for each shot.

A method exists that is able to separate the direct and global components [11], by shifting a checker pattern. The checkers are so small and dense that inter reflected light practically does not change while shifting the pattern. However, when a small surface patch is observed by a camera pixel, it can see the difference between the cases when the observed patch was directly lit by the projector and when not.

The direct component is the light from the modulated source. The global component is the light caused by all other sources. The separation can be used to enhance multiple shot reconstruction method that use binary coding.

2.5.2 Object texture

Structured light methods are meant for featureless surfaces, since the reconstruction of featureless surfaces using passive methods like stereo imaging is difficult or inaccurate. However, it can be the case that the observed objects have texture or sharp edges, which cause unwanted amplitude modulations of the observed signal.

3 Development platform design and calibration

In order to use the combination of a camera and a projector properly, all device properties need to be known (intrinsic parameters, radiometric properties and distortions). Furthermore information on the relative geometric orientation needs to be available. A development platform is a structured light system of which all these parameters are known or are compensated for.

The development platform consists of the actual hardware (a camera and projector) and software that enables the user to do calibrations and map the distortions that play a role. Where possible it should compensate for distortions and deviations from the used models so that the user can occupy himself solely with the development of a structured light method without having to comply to device specific nonlinearities and distortions.

This chapter introduces the available hardware, it shows the theory behind the created software and it presents experimental results to validate the implementations.

3.1 Available Hardware

3.1.1 Camera

The available camera is a Canon EOS 40D. It has a CMOS type of photo sensor and a “BGGR” Bayer tile layout. The resolution is 3908 by 2602 pixels. The camera images are taken in RAW mode and converted by ‘dcrw’ [12], a free tool to convert the undocumented “CR2” (Canon RAW II) format into an uncompressed TIFF file. This tool has the option to output the actual sensor data (14-bit AD converted) without the application of gamma curves, offsets, (usually gradient based-) demosaicing or other unwanted pre-processing.

3.1.2 Projector

The projector is an Optoma EP719 projector, which is based on DMD (Digital MicroMirror Device) technology. DMD devices project different intensities by changing the number of times per frame the source light is reflected into the lens instead of into a heat sink. Due to this very controlled nature of light modulation, DMD devices should be capable of linear projection. This typical projector requires the settings “Degamma: 9” and “White Balance: 0” for linear behaviour. The projector has a 1024 by 768 pixel resolution and 256 grey levels per pixel per colour plane.

3.1.3 LUX meter

A USB enabled LUX meter (DT-1309) is available for radiometric calibration measurements. However, it has some flaws. The device only starts measuring when asked for a measurement, while the meter shows significant start-up

behaviour. It takes several seconds for the meter to stabilize and it only stabilizes if the device is continuously probed for measurements in that period with a frequency of about 10Hz. So when doing a measurement, the first 20 to 30 values should be ignored. A single LUX measurement takes over 5 seconds this way.

A second flaw is the device's behaviour when it automatically switches measurement range. It again takes time for the device to stabilize after a switch and the several measurement ranges seem to have different offsets and even different slopes. Thus, for proper measurements this device is actually useless. The device is used in this project to check the linearity of the camera and projector, by operating in a single measurement range. It is not used to establish a radiometric calibration curve, since this turned out to be not necessary.

3.1.4 Rig



Figure 3.1 – Photographs of the structured light rig. In the left image the projector and camera can be seen. In the right image a platform can be seen that holds an object to reconstruct, in this case a cylinder. As an example, a grid of lines is projected onto the object.

A rig is built to stably hold the devices. The projector is mounted firmly on the “ceiling” of the rig, while the camera is placed accurately beneath the projector, on a rail. The rail only allows for translation in the z-axis. The rotation of the camera in the horizontal plane (heading) is not fixed. Due to this, repositioning the camera could cause minor misalignments in the rotation, but also in the position, since the axis of rotation does not contain the camera centre point.

The rig forms a structure in which the objects to reconstruct can be placed as well. By means of a curtain, the entire set-up can be darkened if necessary. The interior of the rig is shown in Figure 3.1.

3.1.5 Test objects

For testing new structured light methods or examining existing ones, the shape of test objects to reconstruct should be considered. Already available are a block (100 × 100 × 200mm) and a cylinder (200mm high with a radius of 50mm). These objects can provide several interesting surface profiles. The block

provides linear surfaces. The corners of the block provide a discontinuity in surface slope, but not in depth. The cylinder provides a smooth, but nonlinear slope to reconstruct. The edges of the objects of course cause discontinuities in depth.

The objects have a white paper surface, which only has considerable texture when looked at a very close range.

To test new principles, like in this project, simple objects to reconstruct with a white surface are sufficient for first tests.

3.2 Geometric calibration

This section explains the procedure and the theory behind geometric calibration, which is used to calibrate the camera-projector combination intrinsically and extrinsically. The procedure is based on the method described by Zhang [13] and implemented by Bouguet [14] in Matlab. Their method focuses on the calibration of cameras and stereo camera systems. Section 3.2.1 introduces the principles of camera calibration. Section 3.2.2 is on stereo camera calibration. The projector-camera setup is modelled as a stereo vision system and the toolbox is adjusted and extended to allow for projector-camera calibration. This is described in section 3.2.3. All the required steps to calibrate the system are combined in the calibration procedure explained in section 3.2.4. Finally, the stability and reliability of the results are examined in section 3.2.6.

3.2.1 Camera calibration

As explained in section 2.2.1, geometrical calibration of the camera entails the estimation of the following parameters: f_{cx} , f_{cy} , $k_{c1\dots c3}$, $p_{c1\dots c2}$, x_{co} , y_{co} . All these parameters are required to predict the projected coordinate on the image plane of a coordinate in 3D space. In the process of estimating the parameters, the 3D coordinates are known as well as their projected coordinates on the image plane. Each measurement consisting of a 3D coordinate and its projected coordinate, results in two equations that can be used in the estimation process. Since nine parameters are to be estimated, at least five of these pairs are required. However, when many more are available, a more accurate result can be obtained by means of maximum likelihood estimation, i.e. minimizing:

$$\sum_{i=1}^N \|\mathbf{x}_{c,i} - \hat{\mathbf{x}}_{c,i}(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co})\|^2 \quad 3.1$$

where N is the number of available coordinate pairs, $\mathbf{x}_{c,i}$ the actual projection of the known 3D coordinate \mathbf{X}_i and $\hat{\mathbf{x}}_{c,i}(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co})$ the modelled projection of \mathbf{X}_i using a certain set of model parameters.

In [13] the minimization is done using the Levenberg-Marquardt algorithm. In the implementation of Bouguet a gradient decent method is used.

Providing known 3D coordinates

A set of 3D coordinates can be generated by building a calibration object. On this object, features must be visible that can be positioned and identified accurately when observed by a camera. For a stable result, the 3D coordinates are not allowed to be co-planar [15]. Since the exact position and orientation of the camera relative to the calibration object is not known, these parameters also need to be estimated. The minimization problem will then include these parameters as well:

$$\min_{(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{r}, \mathbf{t})} \left(\sum_{i=1}^N \|\mathbf{x}_{c,i} - \hat{\mathbf{x}}_{c,i}(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{r}, \mathbf{t})\|^2 \right) \quad 3.2$$

in which \mathbf{r} is a rotation vector and \mathbf{t} a translation vector.

Since the number of parameters is large, many calibration points are needed. Either a very elaborate calibration object is built, or another way to provide coordinates is needed. Zhang's procedure uses a checkerboard pattern, printed on a planar object. By presenting the checkerboard in different orientations at different locations and taking multiple images, a very large number of coordinates can be provided, distributed over the entire 3D space. However, for every image j the relative orientation and position of the checkerboard needs to be estimated. The minimization problem can now be formulated as:

$$\min_{(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{r}_1, \dots, \mathbf{r}_j, \mathbf{t}_1, \dots, \mathbf{t}_j)} \left(\sum_{j=1}^M \sum_{i=1}^N \|\mathbf{x}_{c,i} - \hat{\mathbf{x}}_{c,i}(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{r}_j, \mathbf{t}_j)\|^2 \right) \quad 3.3$$

The minimization problem has increased in complexity, but the calibration procedure itself is now rather simple. The required checkerboard is easy to fabricate and the checkerboard corners are easy to detect and identify. Each new image will provide N more calibration points, while the number of parameters only increases with six (\mathbf{r}_j and \mathbf{t}_j). As N can easily be $\gg 100$ (e.g. a 10x10 checkerboard), it is possible to generate more than enough points with ease.

Note that the checkerboard needs to be presented in independent orientations. As explained in [16], parallel presentation of the checkerboard in a second image will not provide additional constraints and will not aid during minimization.

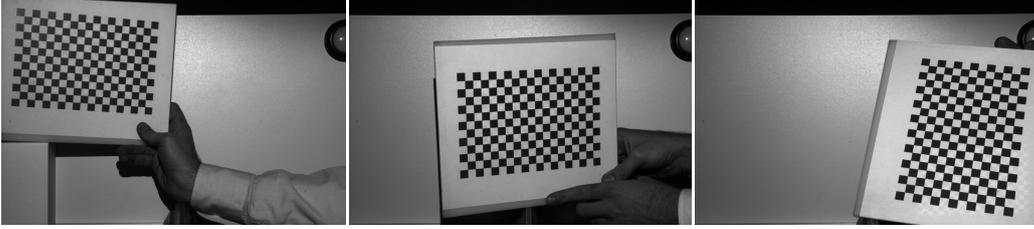


Figure 3.2 – Examples of a printed checkerboard presented to the camera to calibrate.

3.2.2 Stereo calibration

Stereo calibration or extrinsic calibration can be done by the same procedure as intrinsic calibration. The same calibration object must be entirely visible to both cameras in the same shot. The following expression needs to be minimized for $(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{f}_p, \mathbf{k}_p, \mathbf{p}_p, \mathbf{x}_{po}, \mathbf{r}_{1\dots j}, \mathbf{t}_{1\dots j}, \mathbf{r}_p, \mathbf{t}_p)$:

$$\sum_{j=1}^M \sum_{i=1}^N \left\| \begin{bmatrix} \mathbf{x}_{c,i} \\ \mathbf{x}_{p,i} \end{bmatrix} - \begin{bmatrix} \hat{\mathbf{x}}_{c,i}(\mathbf{f}_c, \mathbf{k}_c, \mathbf{p}_c, \mathbf{x}_{co}, \mathbf{r}_j, \mathbf{t}_j) \\ \hat{\mathbf{x}}_{p,i}(\mathbf{f}_p, \mathbf{k}_p, \mathbf{p}_p, \mathbf{x}_{po}, (\mathbf{r}_j + \mathbf{r}_p), (\mathbf{t}_j + \mathbf{t}_p)) \end{bmatrix} \right\|^2 \quad 3.4$$

where the subscript p indicates the second camera. Note that the calibration object has been rotated with the angles in vector \mathbf{r}_j and translated with \mathbf{t}_j with respect to the first camera. The relative orientation and translation for the second camera with respect to the calibration objects equals $\mathbf{r}_j + \mathbf{r}_p$ and $\mathbf{t}_j + \mathbf{t}_p$, where \mathbf{r}_p and \mathbf{t}_p are the orientation and translation of the second camera relative to the first camera.

By first calibrating the cameras separately, an accurate first estimate can be made for all parameters. This goes even for \mathbf{r}_p and \mathbf{t}_p , as long as the calibration images are shot while the camera pair is fixed in the stereo setup. It is obvious that a minimization problem with this amount of unknowns in a highly non-linear set of equations can only be solved in reasonable time when a proper first estimate is available.

The fact that the second camera is this stereo camera pair was indicated with the subscript p is because in this project the second camera actually is a projector. Hence, the subscript c for the coordinates on the camera image plane and the subscript p for the coordinates on the projector image plane.

3.2.3 Camera-Projector calibration

A checkerboard was chosen, so that the checkerboard corners can be easily detected in a camera image. The second camera is, however, a projector and cannot take an image of the presented checkerboard. Van Koten and Keemink [17] propose to use a projected calibration grid together with a printed grid. By using a different colour for the printed grid and the projected grid, the observation of the two grids can be separated when observed with the camera. A homography can be computed between the camera and projector coordinates.

They use this homography to transform the image of the printed checkerboard and use it to feed Bouguets calibration toolbox.

The transformation of the image is not necessary. Also the choice of colours to separate the projection from the printed grid can be improved. This section will discuss how to incorporate a projector into the existing stereo camera calibration toolbox of Bouguet, based on the approach of Van Koten and Keemink.

Colour separation

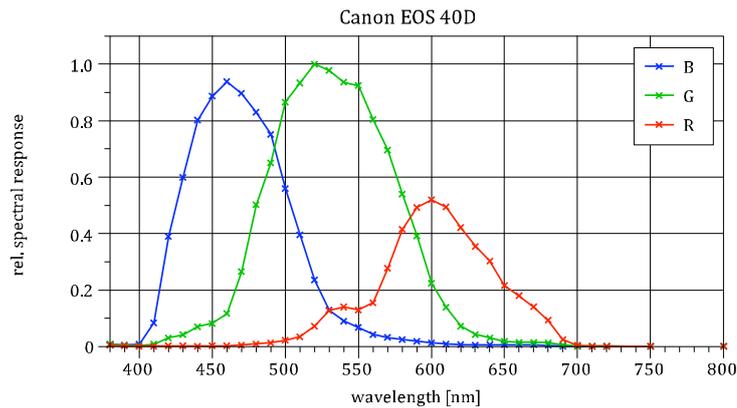


Figure 3.3 – Spectral response measurement [18].

An indication of the spectral response of the camera is presented in Figure 3.3. Although a measurement of the projector spectral response is not available, a likewise response is assumed for the moment. Green light will manifest itself also in the red and blue bands. The red and blue bands still overlap, but much less. A good choice is thus to not use the green channel of the projector. The projector will therefore project the RGB colour $[1,0,1]$, where the intensity range is $[0,1]$ for each channel.

When a red checkerboard is presented, which is printed on white paper and is observed with the camera, the checkerboard will mainly be visible in the blue channel. The white paper will reflect practically as much red light as the red squares. The squares, however, absorb the blue light from the projector and will show black in the blue channel of the colour image.

In the red channel of the projector, the projector can project a checkerboard onto the plane which was fitted with the printed, red checkerboard. Any pattern in the red channel of the projector should not disturb the blue channel of the camera and thus the observation of the printed grid. The projected pattern in the red channel will show up in the red channel of the camera and should not be disturbed by the red printed pattern as white paper and red ink should reflect equally in that channel.

The proposed separation is demonstrated in Figure 3.4. The blue channel is acquired by using only the blue pixels from the Bayer tiles of the camera sensor. Interpolation is omitted since it will not introduce more information. The red channel is acquired likewise by using only the red pixels from the Bayer tiles.

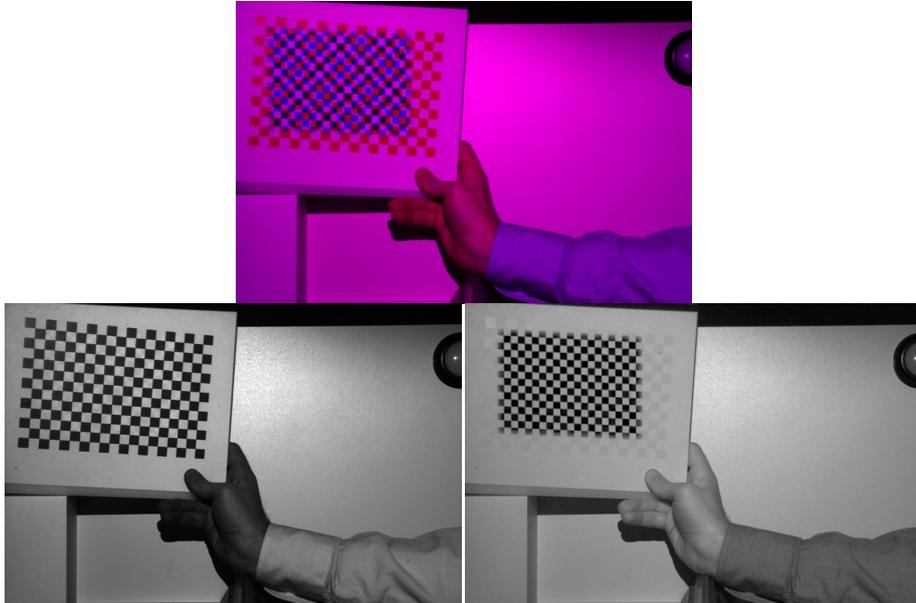


Figure 3.4 - Separation of projected grids. Top: image of blue grid projected onto a red calibration pattern. Bottom-left: the blue channel. Bottom-right: the red channel.

Camera calibration and preparing for projector calibration

First, the camera is calibrated using the blue channels of the images. This will result in the intrinsic camera parameters as well as the relative pose of the calibration grid in each image j , \mathbf{r}_j and \mathbf{t}_j .

Next, the grid corner coordinates are extracted from the red channel. These coordinates correspond with known projector coordinates. Since the grid is projected onto a planar surface before it is observed, the relation between the observed coordinates and the projected coordinates can be described by a homography. The lens distortion of the camera is compensated for, since the camera is already calibrated. The lens distortion due to the projector lens cannot yet be compensated for. To justify the modelling of the projector coordinates to camera coordinates by homography alone, the grid is projected in a small portion of the projection plane. In this local area, the effect of lens distortion is small. Also, as illustrated in Figure 3.5, the projector does not suffer much from lens distortion, while the camera does.

Using the found homography, the coordinates of the printed grid corners in the image are transformed to projector coordinates.

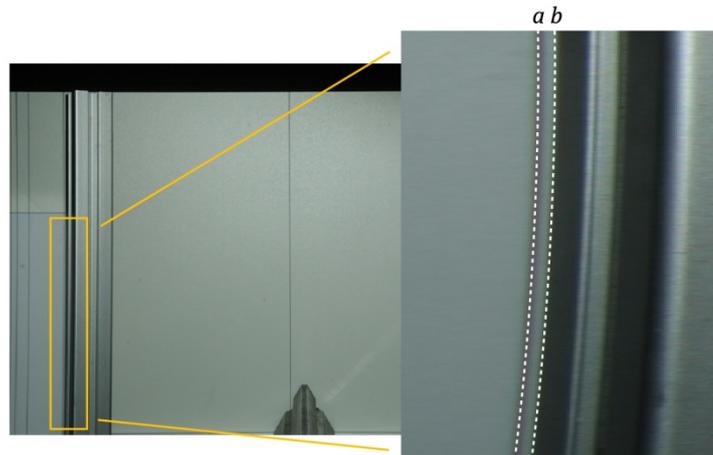


Figure 3.5 – Course analysis of lens distortions. Straight horizontal lines are projected on a flat surface. The actual straightness is compared with a straight metal beam. A part of the image is enlarged and stretched to enhance the effect of lens distortion. The white dotted lines indicate: *a*: a projected line and *b*: the edge of the metal beam. Though bent by camera lens distortion, the projected line is practically equally bent as the metal beam; an indication that projector lens distortion is small.

Projector calibration

Using the transformed coordinates, the projector is calibrated. Now the intrinsic parameters of the projector are known as well as $\mathbf{r}_{j,p}$ and $\mathbf{t}_{j,p}$; the relative rotation and translation of the printed grid with respect to the projector for each image j . Since $\mathbf{r}_{j,p} = \mathbf{r}_j + \mathbf{r}_p$ and $\mathbf{t}_{j,p} = \mathbf{t}_j + \mathbf{t}_p$, each calibration image also gives an estimate of \mathbf{r}_p and \mathbf{t}_p .

System calibration

All the required initial estimates are available to perform the stereo calibration as described earlier. It will refine the estimates for both the intrinsic device parameters as well as for \mathbf{r}_p and \mathbf{t}_p .

Automatic corner finder

Bouguet's toolbox does not come with an automatic corner finder. An initial estimate of a corner is needed before its location can be located with sub pixel accuracy. This first estimate must be given by hand.

Since two grids need to be detected per calibration image and the procedure consists of taking 18 calibration images, this manual grid indication scheme will require much time. Therefore an automatic corner finder is thought of that will generate these rough estimates. Appendix A describes the algorithm.

3.2.4 Calibration procedure

This subsection will introduce the calibration procedure by discussing the steps to follow.

1. Start the geometric calibration script.
The script will aid in the image acquisition and handle the grid projection. When all images are available, it will automatically transfer all data to the calibration toolbox of Bouguet.
2. Supply the script with the size of the printed checkerboard (in number of squares, not corners) and the actual size of the printed squares.
18 grids will be projected on nine different location. Per location, present the printed calibration grid in two different poses for two different shots.
3. The script will project a blue calibration grid and will count down to acquire an image. Make sure to 'catch' the projected grid with the printed grid. The projected grid does not have to (fully) overlap, but it needs to be on the same plane as the printed grid.
The script will acquire an image and will try to auto-detect the corners. If successful, the next grid will be projected. If failed, the same grid will be presented once more. Until all 18 grids are captured, step 3 will be repeated.
4. The script will access Bouguets toolbox and start the camera calibration. The result will be stored.
5. The script will now use the camera calibration parameters to undistort the grid coordinates and transform them to projector plane coordinates.
6. Now the projector is calibrated and the results are stored.
7. Finally the script performs the stereo camera calibration and the final results are again stored.

3.2.5 Calibration result example

A calibration result is presented as follows:

```
Stereo calibration parameters:

Intrinsic parameters of left camera:
Focal Length:      fc_left = [ 6891.55348  6891.04117 ] ± [ 4.59712  4.69314 ]
Principal point:   cc_left = [ 1963.73403  1303.13267 ] ± [ 7.18655  7.06727 ]
Distortion:        kc_left = [ 0.14750  -0.00833  -0.00159  0.00148  0.00000 ]
                  ± [ 0.00298  0.03811  0.00049  0.00051  0.00000 ]

Intrinsic parameters of right camera:
Focal Length:      fc_right = [ 2032.01710  2032.00491 ] ± [ 1.70522  1.75285 ]
Principal point:   cc_right = [ 505.75452  -104.55116 ] ± [ 2.14116  2.98075 ]
Distortion:        kc_right = [ -0.08476  0.08006  -0.00099  -0.00177  0.00000 ]
                  ± [ 0.00373  0.01034  0.00060  0.00023  0.00000 ]

Extrinsic parameters (position of right camera wrt left camera):
Rotation vector:   om = [ 0.00606  -0.00560  0.00401 ] ± [ 0.00111  0.00119  0.00012 ]
Translation vector: T = [ -0.15555  205.55128  0.80863 ] ± [ 0.05676  0.14349  0.45926 ]
```

Note: The numerical errors are approximately three times the standard deviations (for reference).

The variables with the addition `_left` refer to the camera properties. The variables with the addition `_right` refer to the projector properties. `fc` refers to

the focal distance. *cc* refers to the centre point or principle point. Finally, the vector *kc* contains the distortion parameters as follows: $kc = [k_1 k_2 p_1 p_2 k_3]$.

The relative rotation is presented in vector form. The first element concerns the rotation from *y* to *z* (pitch). The second element concerns the rotation from *z* to *x* (heading). The third element is the rotation from *x* to *y* (roll). The unit is radians.

The unit of the translation vector is in millimeters.

By re-projecting all found checkerboard corners in all images and comparing them with the measured checkerboard corners, an estimate can be made of the errors of the estimations. These error figures are presented by mentioning the value of three times the standard deviation.

3.2.6 Consistency check

In order to check whether the geometric calibration procedure generates consistent results, the same set-up is calibrated five times in a row. The results are shown in Table 3.1. Of the five measurements, the mean and standard deviations are computed and compared with the error that is indicated by the toolbox' results. The error figures presented by the toolbox are on average 4.5 times as low as the standard deviation of the 5 measurements.

Table 3.1 – Calibration results of five different calibration of the same set-up. Also given are the mean and standard deviation of the 5 measurements. The “error indication” is the error as indicated by the calibration toolbox. These figures are compared by their ratio to the standard deviation of the 5 measurements.

		1	2	3	4	5	mean	stdev	error indication	ratio
fc_left	x	3,079E+03	3,081E+03	3,081E+03	3,082E+03	3,078E+03	3,080E+03	1,541E+00	5,865E-01	2,6272
	y	3,078E+03	3,080E+03	3,080E+03	3,082E+03	3,078E+03	3,080E+03	1,667E+00	5,766E-01	2,8907
fc_right	x	2,025E+03	2,024E+03	2,024E+03	2,026E+03	2,020E+03	2,024E+03	2,301E+00	5,232E-01	4,3973
	y	2,025E+03	2,023E+03	2,026E+03	2,027E+03	2,021E+03	2,024E+03	2,256E+00	5,434E-01	4,1516
cc_left	x	1,940E+03	1,944E+03	1,950E+03	1,941E+03	1,949E+03	1,945E+03	4,543E+00	6,270E-01	7,2455
	y	1,318E+03	1,324E+03	1,317E+03	1,316E+03	1,320E+03	1,319E+03	3,143E+00	7,854E-01	4,0017
cc_right	x	5,070E+02	5,098E+02	5,168E+02	5,084E+02	5,063E+02	5,097E+02	4,194E+00	6,742E-01	6,2206
	y	-9,305E+01	-8,889E+01	-9,160E+01	-9,909E+01	-9,063E+01	-9,265E+01	3,906E+00	8,916E-01	4,3811
kc_left	k1	-1,733E-01	-1,752E-01	-1,735E-01	-1,762E-01	-1,744E-01	-1,745E-01	1,197E-03	3,661E-04	3,2712
	k2	1,291E-01	1,377E-01	1,413E-01	1,407E-01	1,351E-01	1,368E-01	4,943E-03	1,444E-03	3,4225
	p1	-9,971E-05	1,544E-04	-1,420E-04	-2,934E-04	-1,930E-04	-1,147E-04	1,669E-04	4,174E-05	3,9989
	p2	-6,178E-04	-3,624E-04	-3,264E-04	-3,456E-04	-2,266E-04	-3,758E-04	1,452E-04	3,801E-05	3,8206
	k3	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	
	k3	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	
kc_right	k1	-6,466E-02	-7,242E-02	-6,523E-02	-7,416E-02	-6,866E-02	-6,903E-02	4,228E-03	1,084E-03	3,9016
	k2	3,588E-02	4,638E-02	5,257E-02	5,778E-02	4,191E-02	4,690E-02	8,620E-03	3,006E-03	2,8677
	p1	-1,528E-03	4,194E-04	-1,964E-03	-9,366E-04	-1,014E-03	-1,005E-03	8,980E-04	1,824E-04	4,9248
	p2	-1,222E-03	-1,184E-03	-4,385E-04	-1,019E-03	-1,565E-03	-1,086E-03	4,128E-04	7,691E-05	5,3666
	k3	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	
	k3	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	0,000E+00	
R	Rx	1,388E-02	1,340E-02	1,427E-02	1,150E-02	1,346E-02	1,330E-02	1,067E-03	3,017E-04	3,5348
	Ry	-5,427E-03	-5,337E-03	-6,840E-03	-5,695E-03	-2,148E-03	-5,089E-03	1,751E-03	3,154E-04	5,5508
	Rz	3,493E-03	3,557E-03	3,675E-03	3,306E-03	3,214E-03	3,449E-03	1,872E-04	3,696E-05	5,0643
T	Tx	1,732E+00	1,161E+00	-2,310E-01	1,335E+00	1,840E+00	1,167E+00	8,299E-01	1,317E-01	6,2992
	Ty	2,018E+02	2,007E+02	2,010E+02	2,029E+02	2,007E+02	2,014E+02	9,196E-01	1,753E-01	5,2450
	Tz	3,656E+02	3,655E+02	3,649E+02	3,658E+02	3,641E+02	3,652E+02	7,034E-01	1,749E-01	4,0209

For a better comparison, more measurements are required. The estimation of the standard deviation is not very accurate when only five measurements are available. However, the calibration procedure takes about ten minutes and requires constant attention. Over time, due to fluctuating temperatures of the projector and calibration rig, the parameters can actually drift. The fluctuation of the calibration results over time could thus in fact be caused by fluctuating parameters instead of calibration errors.

At the moment, processing is performed parallel to the image acquisition, so that a failed image can be re-taken immediately. This causes the procedure to take more time. For the purpose of this consistency check, a better way is to first acquire hundreds of images in one sequence. Then take for example 30 or 40 permutations of 18 images to perform the calibration algorithm. The time frame in which the images are taken does not allow considerable parameters changes, while much more calibrations results can be generated and compared.

For now it can be concluded that the calibration procedure produces a consistent output, but the error indications by the toolbox are probably somewhat optimistic. The cause for this “optimism” is yet unknown.

3.3 Radiometric calibration

3.3.1 Linearity measurements

Procedure

The linearity measurement of the camera and projector are combined in one procedure. The lux meter is positioned so that the camera is able to observe it while the projector can illuminate it.

The lux meter position in the projector plane can be indicated by the user by clicking on the lux meter with the mouse cursor – the cursor can be dragged onto the secondary display of the computer (i.e. the projection screen) and click on the actual lux meter.

The calibration script will now project a spot onto the lux meter when started. To prevent the lux meter from clipping or switching measurement range, this “spot method” is chosen over the global illumination method used in [2]. An image is captured and the spot is automatically located by finding the location of the global maximum in the blurred image. This coordinate is stored in memory.

For a certain number of projector input intensities, a spot with this intensity is projected and an image is captured by the camera. At the stored spot location in the image, the value of the blurred image is extracted. Each shot will now result in three measurements: a projected intensity value, a lux measurement and an image intensity value.

The lux measurement is proportional to the radiance of the rays from the projector and the radiance of the rays towards the camera. It strongly depends on the orientation and position of the lux meter what ratio there is between these values. Therefore, the measurements can only be used to determine the nature of the transfer up to an unknown factor.

Measurements

The default settings of the projector are chosen, except for the ‘degamma’ option. The degamma option is used to enhance the contrast of the projected image by amplifying higher intensities more than lower intensities. This non-linear behaviour is unwanted and is thus disabled by setting this option to “setting 9”.

In Figure 3.6 and Figure 3.7 the measurement results of the procedure are presented.

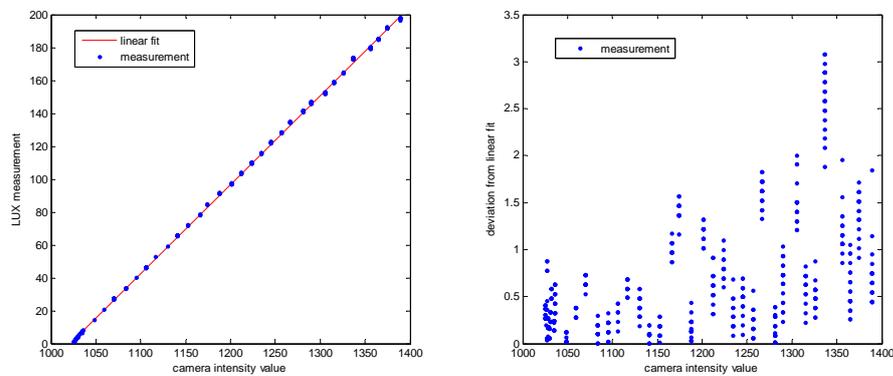


Figure 3.6 – Camera intensity response measurements.

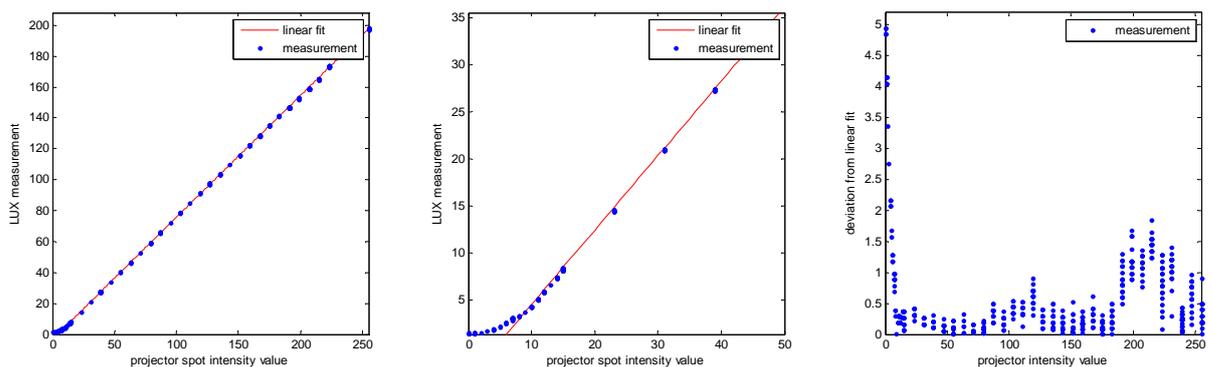


Figure 3.7 – Projector intensity response measurements. The centre image is a close-up of the lower intensity range.

Linearity compensation

Part of the structured light toolbox would be a feature to correct any nonlinear behaviour. However, after the measurements were done, the devices already showed linear behaviour. Any deviations presented in the figures are mainly due

to noise or fluctuations of projector output in time. Only for the lower ten intensity values, the projector shows nonlinear behaviour. Correcting this tiny deviation was omitted and the bottom 10 intensity values of the projector are simply not used. When, in the future, non-linear devices will be used, this part of the toolbox needs revisiting.

3.4 Pixel mapping and 3D reconstruction

This chapter introduces the pixel mapping procedure. The procedure is able to trace every pixel from the projector to a projected position on the camera image plane. Combined with the geometric calibration data the pixel mapping can be used to do 3D reconstruction and produce a depth map.

The first section describes the pixel mapping procedure, while the second section explains the generation of the depth map by stereo triangulation.

3.4.1 Pixel mapping

The implemented procedure is based on [4]. Multiple projections and images are needed to complete the pixel mapping. Therefore, analysing moving objects is not possible. Every pixel in the projector plane is assigned an identical binary code. This will require $N = \lceil \log_2(\text{number of pixels}) \rceil$ bits. Per projection and image, one bit of this code can be “transmitted”. The projector pixel is turned on or off according to the bit to transmit and the camera will determine per camera pixel if it observes a 1 or a 0. The observed bits are concatenated and decoded to see what projector pixel lit that part of the scene. An example is given in Figure 3.8.

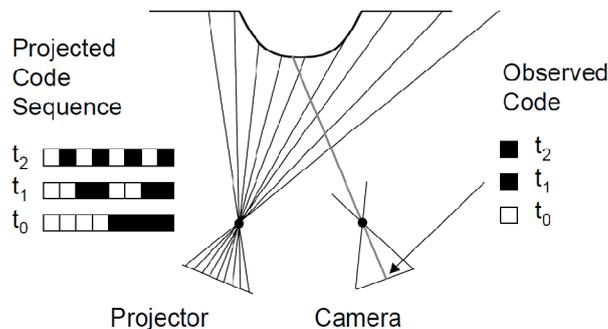


Figure 3.8 [4] – Pixel mapping using binary codes.

Per image pixel it needs to be decided whether a 1 is received or a 0. A possibility is to use a threshold, which has to be pixel specific due to the albedo of the observed surface. A more robust way is to use two images to project the bit pattern and its inverse. The two received images are compared and if a pixel has a higher intensity value in the first image than in the second, it is decided that the received bit is a 1. Nevertheless, bit errors can be made, especially when inter reflections are present. This will be discussed in section 3.4.2.

Gray codes

To limit the effect of bit errors, a proper coding scheme can be used. A gray code is such a scheme and widely used in structured light systems. First, the row indices of the projector pixels are encoded and transmitted, then the column indices. An example of a Gray code is given in Figure 3.9. The advantage of using a Gray coding scheme is that when only one bit is flipped due to an error, the decoded number (e.g. the row or column index that was encoded) will only differ by 1.

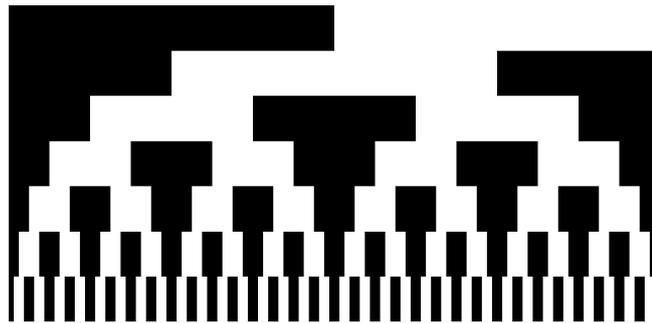


Figure 3.9 – Gray codes. Each column represents a code, each row represents a bit.

Line shifts

When every individual pixel is encoded and mapped with the Gray code method, the bit patterns that encode the least significant bits will be very dense. That means that consecutive pixels in the image will be turned on and off. When the camera is slightly out-of-focus, the surface texture exhibits some subsurface scattering or has a large slope towards the camera's (or projector's) point of view, most of these bits will be wrongfully decoded.

A line shift method is proposed that allows for pixel mapping with subpixel resolution. If enough space is left between the projected lines, inter reflections do no longer play a role. Gray codes can be replaced totally if only one line is shifted all over the screen. In that case 768 lines need to be projected to encode all the projector lines (and this 768 images need be shot) as well as 1024 more projections and images to map all columns, given 1024x768 projector screen resolution.

However, projecting multiple lines at once will do as well. Assuming that sub surface scattering and other sources or blurring are limited to one or two pixels, it is safe to say that a line can be projected every six lines. However, when a line is detected in an image, its origin is ambiguous. The ambiguity can be solved by a course version of the Gray code mapping scheme. Instead of identifying all the projector pixels, the regions where a certain line will be shifted over is identified.

[4] detects the actual line position with subpixel accuracy and does so for all columns and rows of the projector. This way the subpixel mapping is found from projector to camera, i.e. for each projector pixel a sub-pixel position on the camera image plane is found.

In our implementation, the detection is modified in order to achieve a corresponding pair per camera pixel, instead of per projector pixel. This way a depth map at camera resolution is created, making the result compatible with the profilometry results.

For simplicity we consider one camera pixel and just the mapping of a projector row. The projector screen is totally black (off), however, every S rows an entire row is white (on), starting with the first row. An image of the scene, lit by the projector is then shot. Subsequently the lines are shifted one pixel downwards, so that every S rows a row is lit, starting at row 2. An image is acquired and this entire procedure is repeated until S images are shot.

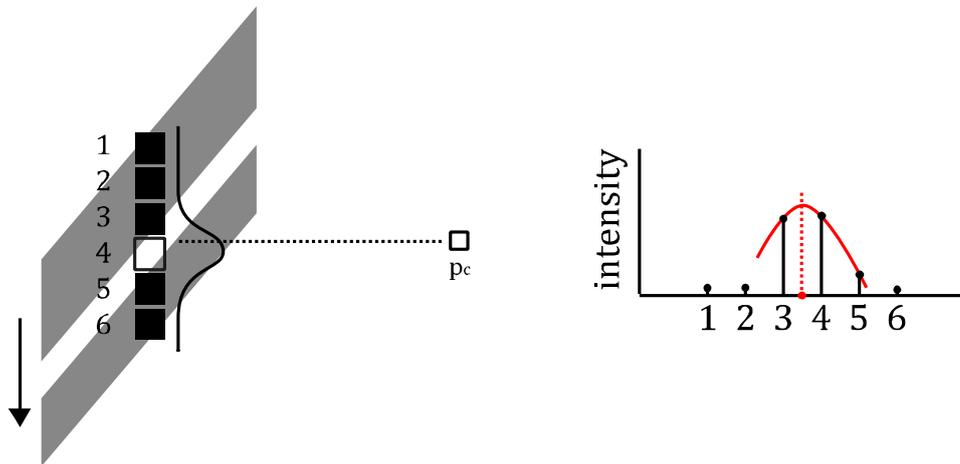


Figure 3.10 - Adapted line shift method. A camera pixel indicated by p_c observes a surface. In six consecutive images, projector row 1 to 6 is lit, while the rest of the projector is black. The situation shown in the left image illustrates image 4 - the fourth row is on. However, due to blurring, the observed intensity is nonzero in images 3 and 5 as well. The red curve resembles a fitted quadratic to the three observed intensity values around the maximum value.

Figure 3.10 illustrates the procedure for $S = 6$. The goal to find out what row is observed by the pixel p_c . It observes row 4 as can be seen in the illustration, however, not in the middle. A sub pixel estimate of e.g. "row 3.6" would be more accurate. The maximal intensity will be observed in image 4, corresponding to row 4. However, the intensity in images 3 and 5 are non zero due to some blurring of the projector and subsurface scattering of the observed surfaces. The intensity values surrounding the maximal value are used to fit a quadratic function. The location of the quadratic's top indicates the observed row with sub pixel resolution.

3.4.2 Effects of inter reflections

Inter reflections of light cause a problem in multiple shot pixel mapping algorithms. In the earlier described pixel mapping procedure, an observed surface patch is labelled “1” when it is lit by the projector and “0” when it is not. To determine this, two images are shot. In the first image the projector illuminates the patches it should, in the second image it will project the inverse and thus illuminate only the patches it should not. If the observed intensity value in the first image pixel is larger than in the second, it was probably lit in the first image.

Inter reflection can cause bit errors when using this method. An example is shown in Figure 3.11. A cylinder and a block are placed next to each other. However, the block is reflecting light onto the cylinder. The amount of light that the block reflects is bigger than the amount of light that will directly illuminate the edge of the cylinder. This will cause an error in the proposed pixel classification.

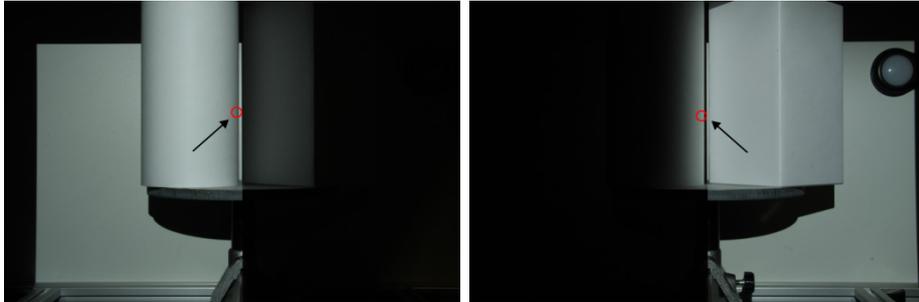


Figure 3.11 - Inter reflection. On the edge of the cylinder, in the indicated spot, the image intensity is lower if the projector is illuminating the spot than if not directly illuminating the spot, but illuminating the block on the right of the cylinder.

The method described in [19] combines basic pixel mapping schemes (like projecting Gray codes) with the results of the light separation procedure described in [11]. Even though it will not solve the problem, it will now be able to classify these pixels as unknown instead of classifying them falsely. In [20] this method is used in an iterative process that performs the pixel mapping procedure multiple times. By turning off the already classified pixels in a next iteration, the amount of light decreases and thus the amount of inter reflecting light decreases. Imagine the case of Figure 3.11; if all pixels illuminating the block are classified, they will not illuminate the cylinder edge in the next iteration, allowing the cylinder edge to be mapped properly in that iteration.

3.4.3 Stereo triangulation and 3D reconstruction

The pixel mapping procedure has resulted in a camera-projector correspondence for each camera pixel. The camera pixels that did not observe a scene patch that could have been lit by the projector will of course result in a false correspondence. However, when the pixel mapping procedure in [19] or

[20] is applied, most of these false correspondences will be eliminated by marking them as “unknown”.

When the geometrical calibration is performed and all intrinsic and extrinsic parameters of the structured light system are known, the pixel coordinates can be transformed into the normalized format. They need to be scaled down with the focal distances, lens distortions needs to be compensated for and the camera pixel coordinates need to be rotated in order to fit the parallel model in Figure 2.4.

Undoing the lens distortion entails the application of the inverse of equations 2.3, 2.4 and 2.5. This is analytically not possible, so an iterative process is required.

The normalized coordinates in equation 2.19 can be used to solve for the depth of the observed surface patches. This is called stereo triangulation. Once the depth is known, the real world X - and Y -coordinates can be computed with equation 2.20.

The lens distortion compensation, coordinate normalization and stereo triangulation are implemented by Bouguet. The final result of the pixel mapping procedure followed by 3D reconstruction can be presented as a depth map, a 3D point cloud or a profile, which is practically a plot of a row or column from the depth map.

In a depth map, the shown image intensity is representative for the depth Z of the observed surface patch per pixel. An example result is shown in Figure 3.12. Combining the depth map Z with equation 2.20 results in a 3D coordinate for each pixel. This allows the presentation of the 3D point cloud in Figure 3.13. An example of a profile is given in Figure 3.14.

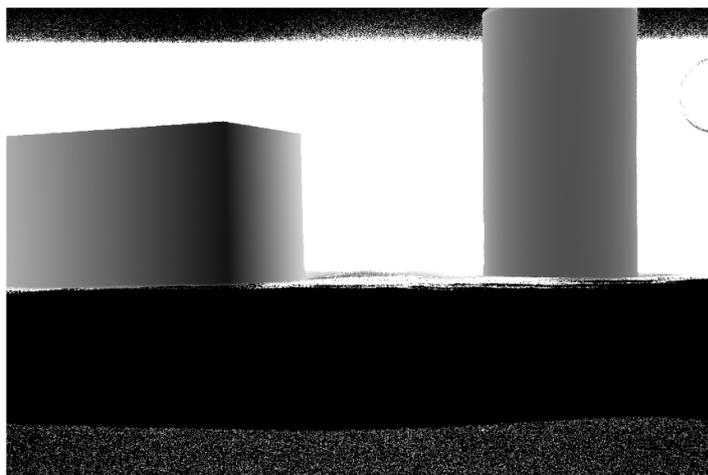


Figure 3.12 – Depth map of the scene as determined by the reference method.

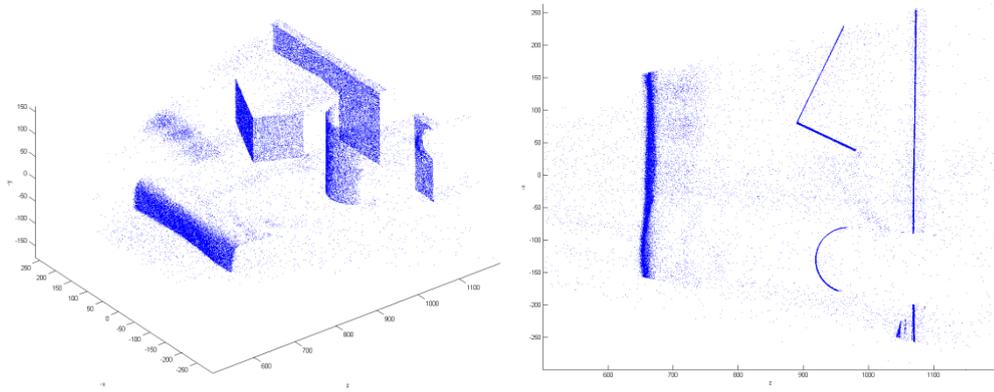


Figure 3.13 – 3D point cloud of the depth map. Left: 3D view, Right: Top view. Note that no efforts are yet undertaken to remove outliers.

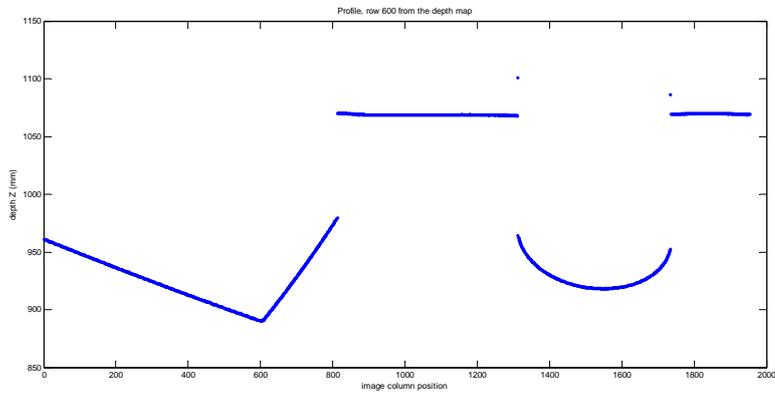


Figure 3.14 – Profile of the scene at row 600 in the depth map.

Accuracy

If no efforts are made to establish a sub-pixel estimate of the corresponding point in the projector plane, the errors in the corresponding coordinate will be up to half a pixel width. Larger errors are possible when the decoding of the observed pixel fails. The latter type of error is left be for now, as this kind of error is not encountered in the scene used in this report.

For simplicity let us assume $\mathbf{T} = [0 \ T_y \ 0]^T$. In that case, equation 2.16 simplifies to:

$$Z = \frac{T_y}{y_{nc} - y_{np} - e} \quad 3.5$$

In which e is the error in the corresponding projector coordinate. This error leads to the following error in Z :

$$\frac{\delta Z}{\delta e} = \frac{T_y}{(y_{nc} - y_{np} - e)^2} = \frac{Z^2}{T_y} \quad 3.6$$

Since a typical value of f_p is about 2000 “pixel widths”, the maximal error would be $\frac{0.5}{2000} = 2.5 \cdot 10^{-4}$. At a depth of 850mm and having $T_y \approx 200mm$, the maximal error in Z would be about $\frac{850^2}{200} \cdot 2.5 \cdot 10^{-4} \approx 1mm$.

A close-up of a piece of profile is shown in Figure 3.15. This piece of profile belongs to a flat plane, thus it should be a line. In this figure the profile that results when no sub pixel estimation is performed is shown in comparison to the profile with sub pixel estimation. As predicted, the resolution is about 2mm and the error between the real depth and the discrete depth is maximally 1mm. In the result with sub pixel estimation, the discrete steps can still be seen slightly, but it is safe to say that the error is well below 1mm.

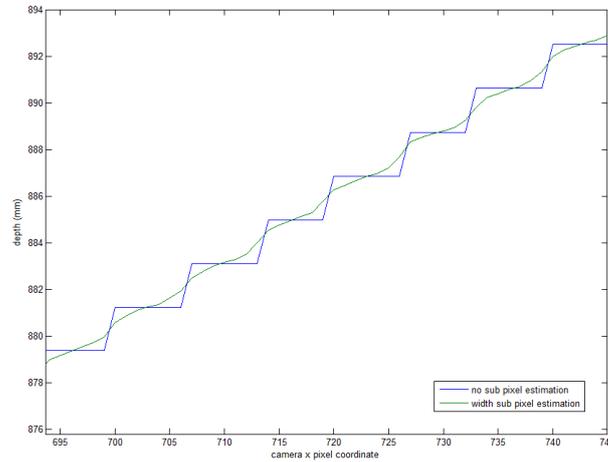


Figure 3.15 – Close-up of the profile at a row 820 in the camera. The result is shown for both cases where the corresponding projector coordinate is estimated either with or without sub pixel precision.

Some pixels in areas where inter reflections play an important role, decoding errors can be made. Another cause of errors, which is more difficult to analyse, is the systematic error introduced by propagation of the errors in the calibration parameters.

3.5 Coordinate rectification

In the previous section, the normalized pixel coordinates are used to reconstruction depth. The stereo triangulation method provided by the toolbox of Bouguet takes the relative translation and rotation of the two devices into account when producing the depth map.

However, in many reconstruction schemes like the proposed method in chapter 4, a perfectly aligned setup is expected. This means that the devices have no relative difference in orientation and are solely translated in one dimension. It is

extremely difficult to achieve this type of setup in practice. The rectification of the image plane coordinates solves this problem.

Any reasonable setup, where only some effort is taken to align the devices, can be rectified. The idea is illustrated in Figure 3.16.

Figure 3.17 shows a 2D view of the rectification principle. A 3D coordinate X generates the corresponding camera and projector coordinates $\mathbf{x}_c = [x_c, 1]^T$ and $\mathbf{x}_p = [x_p, 1]^T$ on the planes C and P , w.r.t. to axes c and p . By rotating these coordinates with $-r_2$ and $-r_1$, the coordinates $R_2\mathbf{x}_c$ and $R_1\mathbf{x}_p$ are obtained. These coordinates can be projected onto the planes C_{new} and P_{new} and $\mathbf{x}_{c,new} = [x_{c,new}, 1]^T$ and $\mathbf{x}_{p,new} = [x_{p,new}, 1]^T$ are computed.

The new, rectified coordinates can be used in the framework that requires $T_z = 0$ and $T_x = 0$. For the new value of T_y , the value of $|\mathbf{T}|$ should be used.

The reference method reconstructs the coordinate X with respect to the original pose of the camera. The reconstruction of X using the rectified coordinates should be rotated back to the original orientation for comparison. In Figure 3.17 this means rotating the reconstructed result with r_2 .

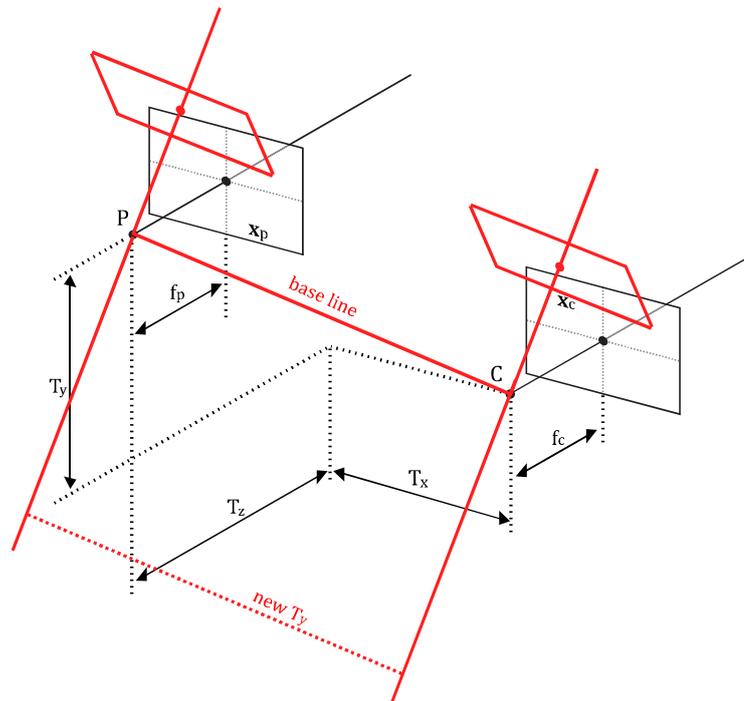


Figure 3.16 - Example of the result of rectified coordinates. For the proposed method, T_x and T_z are supposed to be zero. In reality they might not be, like shown in this image. However, two new, parallel image planes can be defined by rotating the original image plane coordinates. These new coordinates belong to a stereo set-up with $T_x = 0$ and $T_z = 0$ and a new value for T_y . An example is overlaid in red.

Measurements of the devices radiometric behaviour were performed. Their intensity response turns out to be linear and a method to correct for nonlinearity proved to be not necessary.

A pixel mapping procedure is implemented that estimates the corresponding projector plane coordinate per camera pixel. This mapping is used together with the calibration data to form a reference depth map. Depending on the accuracy of the calibration data, the depth estimates are accurate within a millimetre.

With this toolbox, the claims and methods described in the next chapter can be validated.

4 Phase based profilometry

This chapter leads to a hypothesis on a new phase based profilometry method. After presenting some existing methods and introducing the terms “phase direction” and “orthogonal direction” in the first section, the mathematics of phase projections are elaborated. Based on the acquired equations a hypothesis is formed.

4.1 Introduction

In stereo imaging a pair of corresponding points results in the estimation of a 3D coordinate. For a certain location in one image, the corresponding point can be found on the epipolar line in the other image.

In a structured light system, the first camera is replaced by a projector. The location in the projector screen can be seen as the phase within the projector signal. The projector is able to project a periodic function based on the phase of the projector screen. When the camera is able to estimate the observed phase of the used periodic function, it has an estimate for the location within the projector screen. Like in stereo images, this location can be used together with the known geometry of the system to estimate a 3D coordinate. Using phase while scanning over an epipolar line in the image to estimate a depth profile is called phase based profilometry.

Estimating the phase of the observed signal is difficult when not only the projected signal is observed, but the signal is distorted by the albedo of the observed surfaces and has offsets due to inter reflections and global illumination sources. Furthermore, phase estimations are always ambiguous. In case of a sine wave, the estimated phase can always be an integer times 2π larger or smaller.

A lot of work is done to overcome these difficulties. A common practice is to perform a phase shifting method as described in [4]. A sinusoid pattern is projected and observed. In a second image, the sinusoid is shifted slightly in phase and observed once more. Using at least three or more images, a complete period is stepped through. Each camera pixel has now observed one entire period in those three or more, say N images. The phase of the period can be easily determined by examining the first harmonic from the N -point FFT of the observed values. This method is called phase measuring profilometry (PMP).

As explained earlier, a method that uses multiple images is not capable of analysing moving objects. Guan e.a. [21] propose a method that enables the user to merge multiple PMP images to project into one shot. First, the devices are aligned parallel. In the image and projector planes two directions are defined: the phase direction and orthogonal direction. The phase direction is the direction that is parallel to the baseline (the vector from camera centre to

projector centre). The orthogonal direction is the direction orthogonal to the phase direction. To illustrate the effect of projecting and observing an orthogonal frequency, we refer to Figure 1.1.

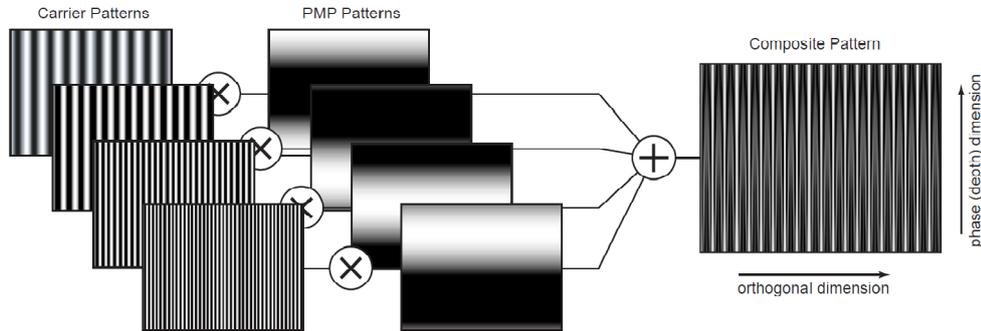


Figure 4.1 – Illustration from [21]; Purely orthogonal frequencies are modulated by PMP patterns.

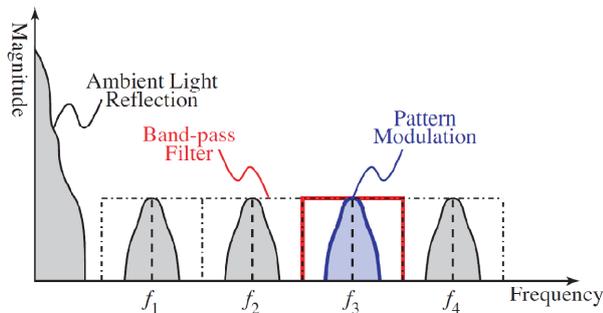


Figure 4.2 – Figure from [21]; When the spectrum of the observed image is analysed along the orthogonal direction, the four projected images are positioned in different bands.

Frequencies that only have an orthogonal component are not transformed by the scene when observing the projection. It is illustrated in Figure 4.1 how these frequencies are used to merge multiple PMP patterns into one image. In the observed image, the PMP image can be extracted by means of band filters, as shown in Figure 4.2.

To overcome ambiguities, a low frequency pattern should be projected, so that the range between ambiguous solutions remains large. However, the low frequency results in inaccurate phase estimation. Using multiple frequencies can combine the accuracy of the higher frequency with the uniqueness of the low frequency solution, which is done in [21].

Some methods, such as in [6], use phase unwrapping techniques to solve for the ambiguity. However, the unwrapped result requires a known starting point and is possibly invalid after steps in the phase due to steps in depth.

All encountered literature focuses on estimating the phase in the so called phase direction which is along the epipolar lines in the camera image. In this report a new approach is presented, which, like the work in [21], uses the fact that

patterns that are repeated only in the orthogonal direction are not affected by the depth of the scene when projected and observed. More on the new method is discussed in section 4.2 and further.

4.2 Phase and orthogonal phase component

4.2.1 Image prediction

The relation between \mathbf{x}_{np} and \mathbf{x}_{nc} is known with given \mathbf{X} and so the image that is observed with the camera can be predicted by sampling the intensity of the projected image at \mathbf{x}_{np} for a given \mathbf{x}_{nc} and \mathbf{X} .

The projected intensities are $P(x_{np}, y_{np})$. The observed intensities are denoted by $C(x_{nc}, y_{nc})$. At a certain image coordinate x_{nc}, y_{nc} a 3D coordinate $\mathbf{X}(x_{nc}, y_{nc})$ is observed. This coordinate is in turn illuminated by a ray from the projector which is modulated by the intensity value of $P(x_{np}(x_{nc}, y_{nc}), y_{np}(x_{nc}, y_{nc}))$. The relation between x_{nc}, y_{nc} and x_{np}, y_{np} is given by equation 2.18. Now the observed image can be predicted:

$$C(x, y) = P\left(\frac{Z(x, y)x - T_x}{Z(x, y) - T_z}, \frac{Z(x, y)y - T_y}{Z(x, y) - T_z}\right) \quad 4.1$$

For clarity, the subscript nc that indicates normalized coordinates is omitted in this equation and in the rest of this entire chapter.

Note that Z is now a function of x and y as it resembles the depth of the coordinate that is observed on the camera image plane at (x, y) .

4.2.2 Projected function

The projector is capable to spatially modulate the emitted light rays as follows:

$$P(x, y) = f(\omega_x x + \omega_y y) \quad 4.2$$

with $f(t)$ a periodic function over 2π and ω_x and ω_y two components of the projected frequency.

The observed image will then be

$$C(x, y) = f\left(\omega_x \frac{Z(x, y)x - T_x}{Z(x, y) - T_z} + \omega_y \frac{Z(x, y)y - T_y}{Z(x, y) - T_z}\right) \quad 4.3$$

For simplicity we assume T_x to be zero from now on. When $T_x \neq 0$ the setup can be rotated around the z -axis in such a way that in the new situation T_x is zero and the axes on the image and projector plane have to be rotated to suit the new situation (rectification).

T_z and T_y can be freely chosen, however, they cannot both be zero; in that case the camera would be inside the projector. Choosing $T_y = 0$ will also cause the

projector to be in front or behind the camera, causing one of the devices to not be able to observe or project onto the scene, so $T_y \neq 0$.

The remaining parameters are now T_z , ω_x and ω_y . Furthermore, the actual function $f(t)$ could be for example $f(t) = \cos(t)$ but might as well be any other arbitrary periodic function. For simplicity we analyse the choices for T_z , ω_x and ω_y with

$$f(t) = \cos(t) \quad 4.4$$

In this case the observed image intensities will be

$$C(x, y) = \cos\left(\omega_x \frac{Z(x, y)x}{Z(x, y) - T_z} + \omega_y \frac{Z(x, y)y - T_y}{Z(x, y) - T_z}\right) \quad 4.5$$

To find $Z(x, y)$, the goal is to estimate the phase of the observed signal. The phase can be estimated in two directions. For that purpose two scan line signals can be defined:

$$\begin{aligned} s^x(y) &= C(x, y) = \cos(\phi^x(y)) \\ s^y(x) &= C(x, y) = \cos(\phi^y(x)) \end{aligned} \quad 4.6$$

Note that the signals have either x or y fixed (indicated by a superscript). I.e. while $s^x(y) = s^y(x)$ and thus $\phi^x(y) = \phi^y(x)$, the phases are estimated by a one dimensional estimation scheme that considers either $s^x(y)$ or $s^y(x)$ as a one dimensional signal.

4.2.3 Instantaneous frequency in x- direction

In [2] the depth is estimated by estimating the instantaneous frequency of a scan line signal in the x -direction, while having $\omega_y = 0$:

$$s^y(x) = \cos\left(\omega_x \frac{Z(x, y)x}{Z(x, y) - T_z}\right) = \cos(\phi^y(x)) \quad 4.7$$

Note that if $T_z = 0$ the dependence of the signal on depth is lost. The phase of the signal is denoted by $\phi^y(x)$ and the instantaneous frequency as $F(x)$:

$$F(x) = \frac{\delta\phi^y(x)}{\delta x} = \omega_x \frac{Z(x, y) \frac{\delta Z(x, y)}{\delta x} (1 - x)}{(Z(x, y) - T_z)^2} \quad 4.8$$

In this equation $\frac{\delta Z(x, y)}{\delta x}$ is called the slope of the profile. In [2] the projector is supposed to be infinitely far away so that the projected rays are parallel. This is equivalent to limiting the case to $T_z \rightarrow \infty$ as well as $\omega_x \rightarrow \infty$, while keeping a constant ratio T_z/ω_x . However, the problem remains that the estimation of the instantaneous frequency is not enough to estimate the depth as it is influenced

by the slope as well. Another type of measurement is needed and in [2] it is decided to use the derivative of $F(x)$ while requiring the surface to be locally linear.

4.2.4 Phase in the y-direction

A better approach would be to estimate the phase, since it is influenced by depth alone and not by the surface slope:

$$\phi^y(x) = \omega_x \frac{Z(x, y)x}{Z(x, y) - T_z} \quad 4.9$$

$$Z(x, y) = \frac{T_z \phi^y(x)}{\phi^y(x) - \omega_x x}$$

With $T_x = 0$, the basic approach of profilometry is analysing the scan line in the y-direction, using $\omega_x = 0$ instead of $\omega_x \neq 0$:

$$s^x(y) = \cos(\phi^x(y)) = \cos\left(\omega_y \frac{Z(x, y)y - T_y}{Z(x, y) - T_z}\right) \quad 4.10$$

$$\phi^x(y) = \omega_y \frac{Z(x, y)y - T_y}{Z(x, y) - T_z} \quad 4.11$$

$$Z(x, y) = \frac{\phi^x(y)T_z - \omega_y T_y}{\phi^x(y) - \omega_y y}$$

A remaining problem is that the estimated phase has ambiguous solutions, so that the estimate of $Z(x, y)$ has ambiguous solutions as well.

The number of solutions can be limited by choosing a low frequency ω_y . However, this frequency also affects the sensitivity to noise. This can clearly be seen in the derivative of $Z(x, y)$ to $\phi^x(y)$:

$$\frac{\delta Z(x, y)}{\delta \phi^x(y)} = \frac{(Z(x, y) - T_z)^2}{(T_y - T_z y)\omega_y} \quad 4.12$$

Now let $T_z = 0$. This simplifies the problem to

$$\phi^x(y) = \omega_y \left(y - \frac{T_y}{Z(x, y)}\right) \quad 4.13$$

$$Z(x, y) = \frac{-\omega_y T_y}{\phi^x(y) - \omega_y y}$$

and

$$\frac{\delta Z(x, y)}{\delta \phi^x(y)} = \frac{Z(x, y)^2}{T_y \omega_y} \quad 4.14$$

All that is left to do is to actually estimate the phase from the observed signal. When writing

$$\begin{aligned} \phi^x(y) &= \omega_y \left(y - \frac{T_y}{Z(x, y)} \right) = \omega_y y - \frac{\omega_y T_y}{Z(x, y)} = \omega_y y - m^x(y) \\ \text{with } m^x(y) &= \frac{\omega_y T_y}{Z(x, y)} \end{aligned} \quad 4.15$$

the scan line signal can be treated as a phase modulated wave with carrier frequency ω_y . The signal is modulated by the message signal $m^x(y)$, which holds the information on $Z(x, y)$ for the running variable y .

A proper phase demodulation scheme that can handle the wide band nature of the message signal should now be able to estimate the message signal and thereby the depth.

4.2.5 Phase and orthogonal direction

With $T_z = 0$ and $T_x = 0$ and no relative rotation between the devices, the setup is parallel. The terms phase direction and orthogonal direction can now be used. The phase direction, comparable with the epipolar direction in stereo vision, is parallel to the base line. In this case, this is the y -direction. The orthogonal direction is thus the x -direction. Up to now the superscripts x and y were used to indicate the static variable; an extra subscript will now be used to indicate in which direction a signal is analysed. Also, ω_x will now be denoted by ω_o and ω_y by ω_p .

These terms and new variable names make the equations more general. In a setup where the projector is placed next to the camera instead of on top, the phase direction is the x -direction. When $T_x \neq 0$ and $T_y \neq 0$, the phase direction is a vector. However, after rectification, the formulas that are introduced here will remain valid.

Equation 4.15 can thus be written as

$$\begin{aligned} \phi_p^x(y) &= \omega_p \left(y - \frac{T_y}{Z(x, y)} \right) = \omega_p y - \frac{\omega_p T_y}{Z(x, y)} = \omega_p y - m_p^x(y) \\ \text{with } m_p^x(y) &= \frac{\omega_p T_y}{Z(x, y)} \end{aligned} \quad 4.16$$

The same phase can also be estimated by considering the scan line in the orthogonal direction. The phase of the signal on that scan line is given by

$$\phi_o^y(x) = \omega_p \left(y - \frac{T_y}{Z(x,y)} \right) \quad 4.17$$

Even though the signal appears to be the same as $\phi_p^x(y)$, the running variable is in this case x . Only the depth itself influences the phase. In fact, the signal is phase modulated without a carrier frequency. A carrier frequency can be introduced for this scan line by setting $\omega_o \neq 0$. The question is now how this influences the scan line signal in the phase direction and what the influence is of the carrier frequency on the performance of the phase demodulation technique that is used.

4.2.6 Phase in orthogonal direction

So let $\omega_o \neq 0$, now

$$\phi_p^x(y) = \omega_o x + \omega_p \left(y - \frac{T_y}{Z(x,y)} \right) = \omega_p y + m_p^x(y) \quad 4.18$$

$$Z(x,y) = \frac{\omega_p T_y}{\phi_p^x(y) - \omega_p y - \omega_o x}$$

$$\frac{\delta Z(x,y)}{\delta \phi_p^x(y)} = \frac{Z(x,y)^2}{T_y \omega_p} \quad 4.19$$

The choice for ω_o does not influence the error sensitivity of $Z(x,y)$ to estimation errors in $\phi_p^x(y)$. However, the scan line in the orthogonal direction has gained a carrier frequency ω_o :

$$\phi_o^y(x) = \omega_o x + \omega_p \left(y - \frac{T_y}{Z(x,y)} \right) = \omega_o x + m_o^y(x) \quad 4.20$$

Both scan lines now provide information on the depth in their message signals, which can be extracted by a proper phase demodulation scheme.

$$m_p^x(y) = -\frac{T_y \omega_p}{Z(x,y)} + \omega_o x \quad 4.21$$

$$Z(x,y) = \frac{T_y \omega_p}{\omega_o x - m_p^x(y)}$$

$$m_o^y(x) = -\frac{T_y \omega_p}{Z(x,y)} + \omega_p y \quad 4.22$$

$$Z(x,y) = \frac{T_y \omega_p}{\omega_p y - m_o^y(x)}$$

4.3 Phase demodulation

The general approach to phase demodulation starts with a phase modulated signal

$$s(t) = a(t) \cos(\omega_c t + m(t)) = x(t) \cos(\omega_c t) - y(t) \sin(\omega_c t) \quad 4.23$$

with $m(t)$ the message signal, $a(t)$ the amplitude modulation and

$$\begin{aligned} x(t) &= a(t) \cos(m(t)) \\ y(t) &= a(t) \sin(m(t)) \end{aligned} \quad 4.24$$

$x(t)$ and $y(t)$ can be obtained by first multiplying the signal with $\cos(\omega_c t)$ and $-\sin(\omega_c t)$ respectively:

$$\begin{aligned} s^x(t) &= s(t) \cos(\omega_c t) \\ &= \frac{1}{2}(x(t) + \cos(2\omega_c t) x(t) - y(t) \sin(2\omega_c t)) \\ s^y(t) &= -s(t) \sin(\omega_c t) \\ &= \frac{1}{2}(y(t) - \cos(2\omega_c t) y(t) - x(t) \sin(2\omega_c t)) \end{aligned} \quad 4.25$$

After low pass filtering to remove the band around the double carrier frequency, we have

$$\begin{aligned} s^x(t) &= \frac{1}{2}x(t) \\ s^y(t) &= \frac{1}{2}y(t) \end{aligned} \quad 4.26$$

The message can be retrieved from $x(t)$ and $y(t)$ by

$$m(t) = \arctan\left(\frac{y(t)}{x(t)}\right) \quad 4.27$$

and thus be estimated using the low passed filtered $s^x(t)$ and $s^y(t)$:

$$\hat{m}(t) = \arctan\left(\frac{s^y(t)}{s^x(t)}\right) \quad 4.28$$

The fact that a low pass filter is required to obtain the message implies that $m(t)$ must be band limited to remain intact. The width of this band depends on the carrier frequency. In the phase based profilometry approach, the message signal is caused by the depth profile of scene. This is usually not a band limited signal due to discontinuities in depth.

Near discontinuities transient effects cause the message signals and thus the depth estimate to be distorted. These effects will diminish as the carrier frequency becomes very high. This fact is exploited when projecting a sine wave with a high orthogonal frequency component.

4.4 Hypothesis

Projecting a sine pattern in the phase direction with radial frequency ω_p and estimating the observed phase can be used to estimate the depth of the observed coordinates. A high frequency is needed for accuracy, while a low frequency is required to limit the ambiguity of the estimated phase.

Adding a frequency component in the orthogonal direction does not influence the accuracy of the depth estimation using the scan line in the phase direction. However, the scan line in the orthogonal direction can now be used to estimate the phase and can have high frequency carrier.

Near discontinuities in the profile, discontinuities in phase can be expected. Naive phase estimation schemes, such as discussed in section 4.3, expect a certain carrier to be phase modulated by a band pass signal and will fail near these edges.

Even in this naive approach, choosing a high frequency allows for a wider band of message signal and thus a better estimation near profile edges. For this reason it is plausible that adding a high frequency component in the orthogonal direction and using $s_o^y(x)$ instead of $s_p^x(y)$ will improve performance near profile discontinuities.

Hypothesis: By adding an extra high frequency component in the orthogonal direction by setting $\omega_o \neq 0$ in equation 4.2, $s_o^y(x)$ will provide a more accurate phase estimate which is less disturbed by discontinuities. This way ω_p can remain low to keep the space between ambiguous solutions large.

4.5 Summary

Equations 4.21 and 4.22 are the main result of the mathematical endeavours in this chapter. They describe the forward and backward relation between the depth map $Z(x, y)$ and the demodulated message signals $m_p^x(y)$ and $m_o^y(x)$.

The hypothesis was formed that the scan signal in the orthogonal direction could provide a more robust phase estimate when a high orthogonal frequency component ω_o is used for the projected sine wave.

5 Experiments

The goal of the experiments described in this chapter is to validate the hypothesis formed in the previous chapter. First the experimental setup and the procedure are discussed. Next the results are presented and compared.

5.1 Experiment description

The hypothesis is based on a system where the camera has the same orientation as the projector and is only translated in the y -direction. Due to the proper construction of a rig in which the devices can be mounted, the requirements on the relative translation can be met within a millimetre. The way the camera is mounted leaves some room for relative rotation, mainly in the xz -plane (heading). The other rotations are fixed and close to zero. However, the equations that enable the 3D reconstruction require that corresponding points between the devices share the same x -coordinate. Even a slight rotation causes this to not be the case. This can be compensated for in two ways. Either the observed image is transformed or the coordinates per pixel are rectified as discussed in section 3.5.

The first option requires re-interpolation of the image. Since the images contain high frequency information, re-interpolation may introduce errors there. The second option causes the image to remain as is and only changes the coordinates of a pixel.

The hypothesis claims that the results of a basic profilometry approach based on phase estimation can be improved. Therefore, the results of a basic approach need to be available for comparison. This requires some experiments with $\omega_o = 0$. For each experiment with a certain frequency in the phase direction, extra experiments will be done with a selection of frequencies in the orthogonal direction. We expect bad estimation results for low frequencies, so only relatively high frequencies will be chosen for the orthogonal frequencies. High frequencies in the phase direction will cause the space between ambiguous solutions to be small, so some low frequencies are required there. Table 5.1 indicates more precisely which experiments will be done. For some of the lower frequencies in the phase direction, experiments with $\omega_o \neq 0$ are done. The numbers in the table are used to identify an experiment.

As this project did not focus on the actual phase estimation yet, the phase estimation is done by a basic phase estimation scheme that expects a band limited message even though this is not the case. Errors near depth discontinuities are therefore expected. To aid the phase estimation, for each experiment two images are shot with a π phase shift, instead of the ultimate goal of one image.

Table 5.1 – Table indicating for which combinations of ω_o and ω_p experiments will be done. Instead of presenting the actual frequencies (in rad/mm on the projection plane), the wave period is presented number of pixels.

		Period in orthogonal direction (px)					
		Inf	50	40	30	20	10
Period in phase direction (px)	10	1					
	25	2					
	50	3					
	75	4					
	100	5	12	13	14	15	16
	125	6					
	150	7					
	175	8	17	18	19	20	21
	200	9					
	225	10					
	250	11	22	23	24	25	26
	Inf		27	28	29	30	31

To compare the results of the reconstructions with $\omega_o = 0$ and $\omega_o \neq 0$ a reference is required. As described in section 3.4, the development platform offers such a reference method. A reconstruction based on this method is used as “ground truth”.

5.1.1 Test scene requirements

Phase estimation errors will be mostly present near discontinuities. For a fair comparison of the $\omega_o = 0$ and $\omega_o \neq 0$ methods, discontinuities need to be present in both directions. Figure 5.1 presents the proposed test scene and introduces three scan lines that are of interest.

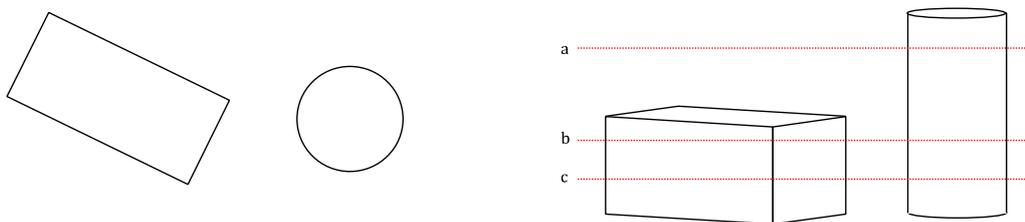


Figure 5.1 – Proposed test scene. Left: a top view of the objects that form the scene. Right: a front view and three possible scan lines in the orthogonal direction. Line (a) is clear of edges in the phase direction (perpendicular to the line) and mostly clear of edges in the scan line direction. Line (b) is near edges in the phase direction and contains edges in the scan direction as well. Line (c) is more or less clear of edges in the phase direction and does contain edges in the scan direction.

5.1.2 Picking the right solution

Using the reference

Even when the proposed method has a lot of ambiguous solutions, the correct solution can be separated from the ambiguous solutions. This is possible since the actual solution is available. Using the known depth, it is possible to predict the observed phase $\hat{\phi}(x, y)$, using equation 5.2. The estimated phase should thus be in the range

$$range(x, y) = \langle \hat{\phi}(x, y) - \pi, \hat{\phi}(x, y) + \pi \rangle \quad 5.1$$

The estimated phase is changed by adding or subtracting integer multiples of 2π to be within this range. This phase is then used as the estimated phase.

Without reference

When a reference is not available, which would be the case when the method is used “for real”, the reference depth map $Z(x, y)$ is not available. Instead, the distance from the camera where the object is located needs to be known approximately. Instead of the reference depth map, a scalar value \hat{Z} for the depth is now available and can again be used in equation 5.2 to predict the observed phase and thus range.

A requirement is that the distance between the ambiguous solutions is large enough so that the object fits in this range, or estimation errors will be made. The depth range is determined by ω_p as well as the distance itself. This depth range can be computed as follows:

Rewrite equation 4.18 as

$$\hat{\phi}(x, y) = \omega_o x + \omega_p \left(y - \frac{T_y}{\hat{Z}} \right) \quad 5.2$$

The depths that result when the phase differs an amount of $\pm\pi$ can be computed as well:

$$\begin{aligned} Z_{-\pi}(x, y) &= \frac{\omega_p T_y}{\hat{\phi}(x, y) - \pi - \omega_p y - \omega_o x} = \frac{\omega_p T_y}{q(x, y) - \pi} \\ Z_{+\pi}(x, y) &= \frac{\omega_p T_y}{\hat{\phi}(x, y) + \pi - \omega_p y - \omega_o x} = \frac{\omega_p T_y}{q(x, y) + \pi} \end{aligned} \quad 5.3$$

with, for clarity,

$$q(x, y) = \hat{\phi}(x, y) - \omega_p y - \omega_o x = -\frac{\omega_p T_y}{\hat{Z}} \quad 5.4$$

Now

$$\begin{aligned}
Z_{range} = Z_{-\pi}(x, y) - Z_{+\pi}(x, y) &= \frac{\omega_p T_y}{q(x, y) - \pi} - \frac{\omega_p T_y}{q(x, y) + \pi} \\
&= \frac{2\pi\omega_p T_y}{q^2(x, y) + \pi^2} \\
&= \frac{2\pi\omega_p T_y \hat{Z}^2}{\omega_p^2 T_y^2 - \pi^2 \hat{Z}^2}
\end{aligned} \tag{5.5}$$

For clarity, the function is plotted in Figure 5.2. The function can be used to find a proper frequency ω_p or location to place the object so that it fits within a solution range. The approximate distance to the camera should be used to correct the estimated phase by adding or subtracting a multiple of 2π , so that it is in the range $[\hat{\phi}(x, y) - \pi, \hat{\phi}(x, y) + \pi]$.

5.1.3 Experiment procedure

To summarize, the procedure of the experiments is listed here in chronological order:

Acquisition

1. Align camera and projector so that $T_z \approx 0$ and $T_x \approx 0$ and $\mathbf{R} \approx \mathbf{I}$.
2. Acquire a dark frame D .
3. Perform calibration procedure to determine intrinsic and extrinsic device parameters. Use the projectors intrinsic parameters and \mathbf{T} to normalize and rectify the projector pixel coordinates. Use the cameras intrinsic parameters and \mathbf{R} and \mathbf{T} to normalize and rectify the camera coordinates.
4. Build the test scene.
5. For all the combinations of ω_o and ω_p in Table 5.1:
 - Project $P_1(x, y) = \cos(\omega_o x + \omega_p y)$, using the rectified projector coordinates.
 - Capture image C_1 and subtract the dark frame.
 - Project $P_2(x, y) = \cos(\omega_o x + \omega_p y + \pi)$
 - Capture image C_2 and subtract the dark frame.
 - Compute $C_N = (C_1 - C_2) ./ (C_1 + C_2)$
6. Perform the pixel mapping procedure to generate reference material.

Processing

7. In the first 11 experiments, where $\omega_o = 0$, estimate the phase from C_N in the vertical direction, i.e. use the columns of the image to estimate a per pixel phase. This results in $\phi_p^x(y)$. Subtract $\omega_p y$ to obtain $m_p^x(y)$.
Use the image rows in experiments 12 to 31, where $\omega_o \neq 0$. This results in $\phi_o^y(x)$. Subtract $\omega_o x$ to obtain $m_o^y(x)$.
In both cases, first correct the estimated phase as described in section 5.1.2 in order to choose the right solution from the ambiguous ones.

8. In the first 11 experiments, use equation 4.21 to reconstruct the depth $Z(x, y)$. Use equation 4.22 in experiments 12 to 31. Use for x and y the rectified camera coordinates.
9. Use equation 2.20, the rectified camera coordinates and the depth map to compute for each camera pixel $X(x, y)$ and $Y(x, y)$.

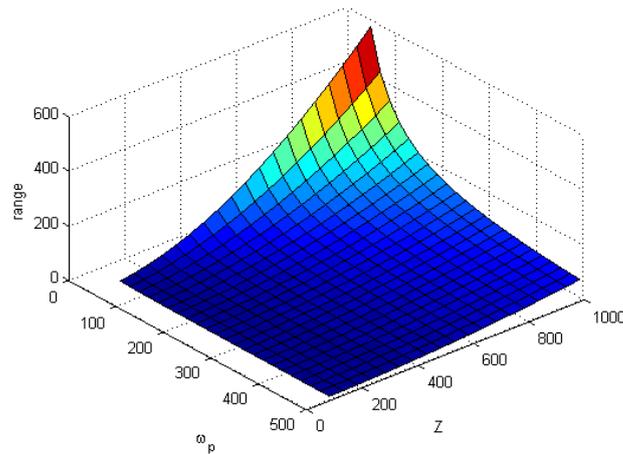


Figure 5.2 - A plot of the range - the depth between ambiguous solutions - for varying distance and ω_p .

5.2 Measurements and results

In the discussion of the results, we often refer to the experiments by their number. It is useful to consult Table 5.1 for reference.

5.2.1 Calibration result

The calibration procedure resulted in the following data:

```
Stereo calibration parameters:

Intrinsic parameters of left camera:
Focal Length:      fc_left = [ 5491.52986  5492.37905 ] ± [ 3.29472  3.19812 ]
Principal point:   cc_left = [ 1969.20734  1292.74174 ] ± [ 7.14431  6.83219 ]
Distortion:        kc_left = [ 0.04391  0.21224  -0.00186  0.00196  0.00000 ]
                  ± [ 0.00320  0.04225  0.00046  0.00053  0.00000 ]

Intrinsic parameters of right camera:
Focal Length:      fc_right = [ 2032.81500  2030.37611 ] ± [ 1.43019  1.50315 ]
Principal point:   cc_right = [ 501.08256  -111.81720 ] ± [ 2.21003  2.95681 ]
Distortion:        kc_right = [ -0.09292  0.07267  0.00073  -0.00170  0.00000 ]
                  ± [ 0.00375  0.01075  0.00058  0.00023  0.00000 ]

Extrinsic parameters (position of right camera wrt left camera):
Rotation vector:   om = [ 0.00798  0.02599  0.00430 ] ± [ 0.00101  0.00139  0.00012 ]
Translation vector: T = [ 2.55837  205.65038  -0.78995 ] ± [ 0.05561  0.12000  0.37002 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

More detail on these figures is given in section 3.2.5.

5.2.2 Reference depth map

The reference method generated the point cloud as presented in Figure 5.3. An image of the scene and the full depth map are shown in Figure 5.4.

5.2.3 Examples of observed signals

The experiments are performed according to Table 5.1. In the first eleven experiments, the projector only projects a sine wave in the phase direction. In the observed images, the phase is estimated in the vertical direction. Equation 4.21 is used to compute the depth from the estimated phase. An example image is shown in Figure 5.5A, which shows the observed signal of experiment 5.

In the experiments 12 and further, ω_o was set to be non-zero. Experiments 12 to 16 have the same ω_p as experiment 5, but with increasing ω_o . Experiments 17 to 21 have the same ω_p as experiment 8 and experiments 22 to 26 have ω_p equal to experiment 11. In these images it makes sense to estimate the phase in the horizontal direction. The observed signals of experiments 14 and 29 are shown in Figure 5.5, C and B respectively. In case of experiment 29, $\omega_p = 0$. As predicted by equations 4.21 and 4.22, no depth information can be retrieved from the observed phase. This is also clear when examining the observed signal empirically; the vertical lines (indicating constant phase) are not affected by changes in depth.

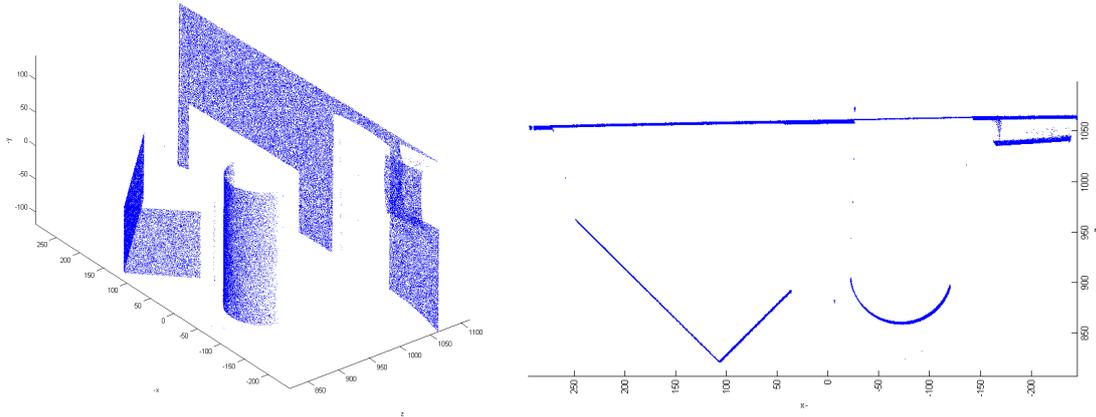


Figure 5.3 – Reference depth map. Left: 3D point cloud. Right: Point cloud viewed from top. The coordinates are w.r.t. the camera centre. The unit is millimetre.

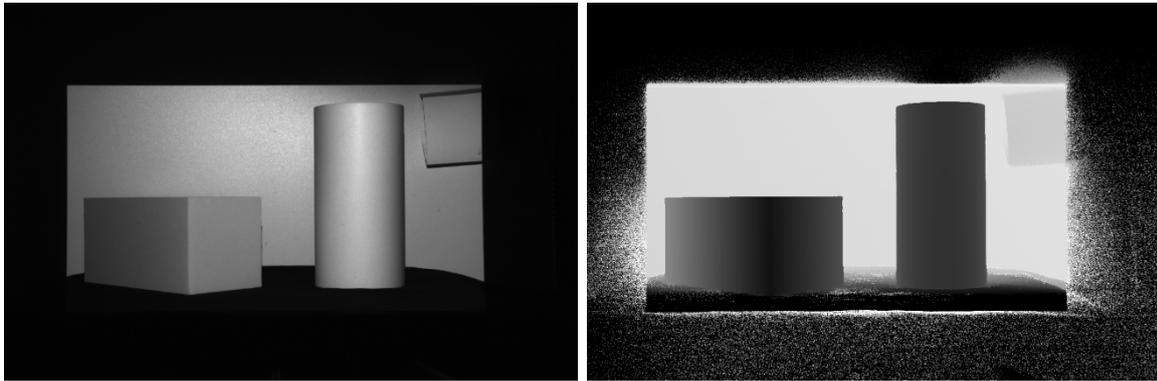


Figure 5.4 – Right: Image of the scene. Left: Depth map of the scene. Black represents <800mm while white represents >1100mm. Gray scales indicate depth in between.

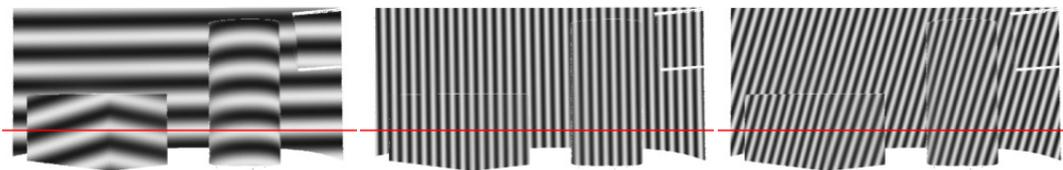


Figure 5.5 – Observed image when projecting $P(x, y) = \frac{1}{2} + \frac{4}{10} \cos(\omega_o x + \omega_p y)$

(A, left) Experiment 5: $\omega_p \approx \frac{2\pi}{100}$ while $\omega_o = 0$.

(B, middle) Experiment 29: $\omega_p = 0$ while $\omega_o \approx \frac{2\pi}{30}$.

(C, right) Experiment 14: $\omega_p \approx \frac{2\pi}{100}$ and $\omega_o \approx \frac{2\pi}{30}$.

The given frequencies are given in rad/pixel, which is comparable with the actual frequency in rad/mm on the normalized projection plane. Shadowed areas are left blank. The red lines indicate the pixel row which is used for comparing reconstruction results in the remainder of this chapter.

Not so for experiment 14 as it has the same ω_p as experiment 5. Horizontally the signal undergoes the same phase shift in experiment 14 as vertically in experiment 5, in case of a change in depth.

5.2.4 Phase direction reconstruction results

In this subsection the results are shown for the first experiments, where $\omega_o = 0$. The phase is estimated in the vertical direction. Figure 5.6 shows the results for a selection of the first eleven experiments. In each experiment the frequency decreases. The figures compare the possible solutions with the reference result. As the frequency decreases, the space between ambiguous solutions increases but the accuracy of the phase estimation decreases and the sensitivity to errors increases. This causes the accuracy of the depth estimate to decrease.

Since a reference depth map is available, the observed phase can also be predicted. The ambiguous solution that is closest to the predicted solution is chosen for comparison. Note that this choice cannot be made when measuring unknown objects with this method. Therefore, ambiguous solutions within the measurement range are unwanted. For experiments 8 to 11 this is the case and a reference measurement is no longer needed to pick the right solution as explained in subsection 5.1.2.

The 2D images in Figure 5.6 present a “best case scenario”. Above and under the analysed image row (the red line in Figure 5.5), no phase discontinuities are near. When comparing the reference and the phase based method near discontinuities in the vertical direction, the estimation errors are larger. By means of a 3D cloud, the results of experiments 1, 5, 8 and 11 are compared once more in Figure 5.7.

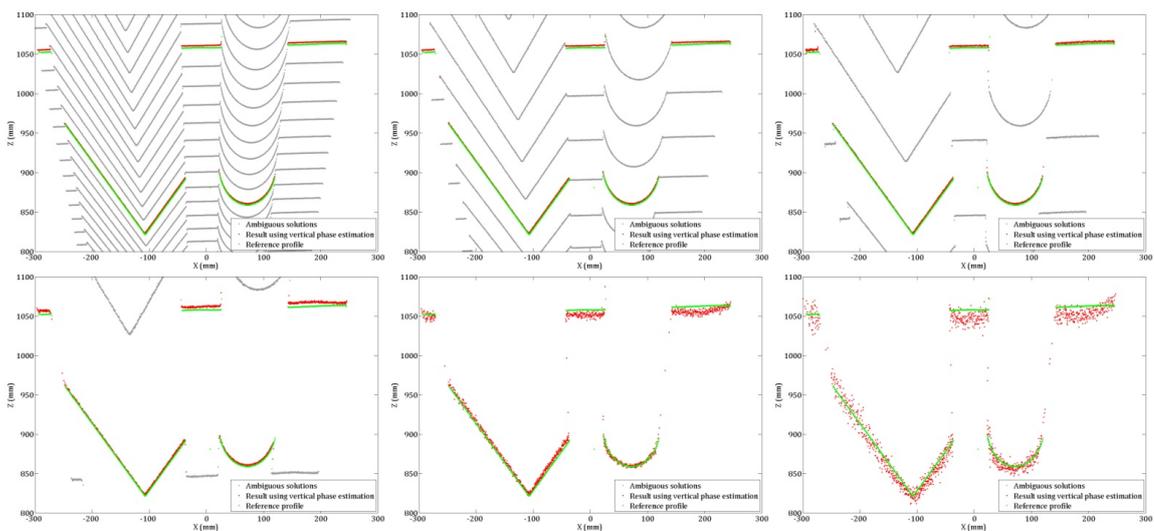


Figure 5.6 – Results for row 825 (as indicated in Figure 5.5). From left to right, top to bottom results of experiments 1, 2, 3, 5, 8 and 11 are shown. Notice how the space between ambiguous solutions (grey profiles) increases while the accuracy decreases. The green profile is the reference profile. The red profile is the nearest chosen solution.

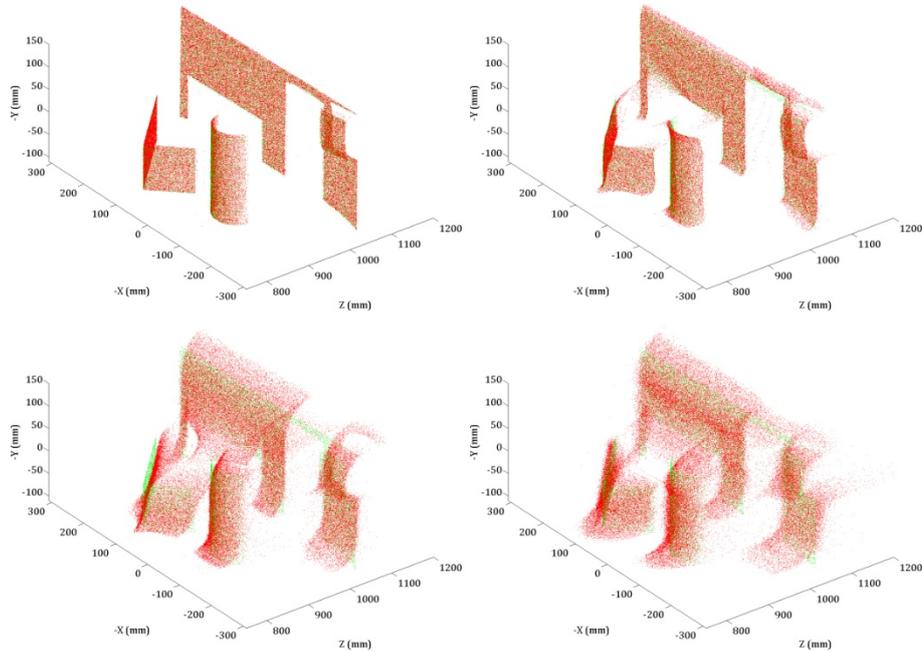


Figure 5.7 – A 3D point cloud of the results of experiments 1, 5, 8 and 11. Near discontinuities in the y-direction, phase estimation fails. The error and the area of error increases as the frequency decreases in the phase direction. Green pixels indicate the reference point cloud. Ambiguous solutions are not shown here.

5.2.5 Trouble using horizontal phase estimation

The hypothesis was made that the decrease of accuracy by choosing a low ω_p could be partly compensated by choosing a high ω_o . The sensitivity to phase estimation errors is not affected by ω_o . The phase estimation itself, however, should be more accurate. The addition of a high frequency component in the orthogonal direction makes the scan line in the orthogonal direction useful. The high frequency improves the phase estimation results as a wider band of message signal is allowed.

A few problems with the method were encountered. In Figure 5.8 the results are shown of experiments 17, 19 and 21, where the period of ω_p is approximately 175 pixels and the period of the orthogonal frequency component is 50, 30 and 10 respectively. As ω_o increases, an offset is introduced.

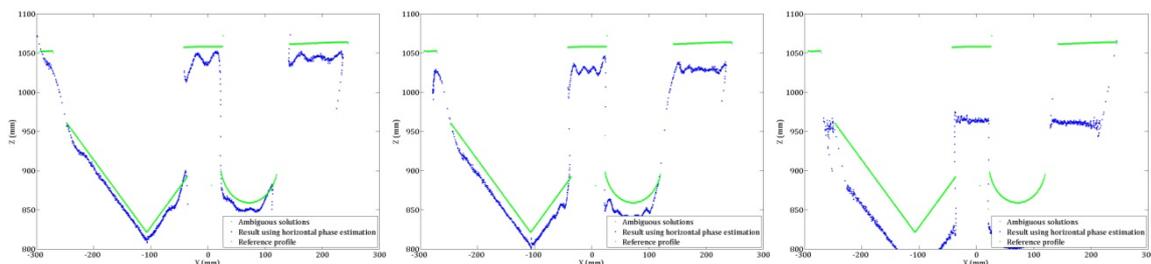


Figure 5.8 – Results of experiments 17, 19 and 21. As ω_o increases, the results get a larger offset w.r.t. the reference.

The problem is an alignment error. Despite using rectified coordinates, a corresponding coordinate pair has different x -components. Parallel alignment should ensure that corresponding coordinates have the same x -component.

Using experiment 31, this problem can be illustrated. In this experiment, the projected sine wave has no phase direction component, so $\omega_p = 0$ and $P(x, y) = \cos(\omega_o x)$. Having $T_z = 0$, equation 4.5 predicts that the camera will observe the projected phase without distortion due to the depth of the scene: $C(x, y) = \cos(\omega_o x)$.

Figure 5.9 shows this is not the case. A phase shift is observed. The phase shift can be used to correct the misalignment in the x -direction. By dividing the phase shift with the used radial frequency ω_o , the displacement can be computed. This is only possible when the misalignment causes a phase shift that is less than π radians.

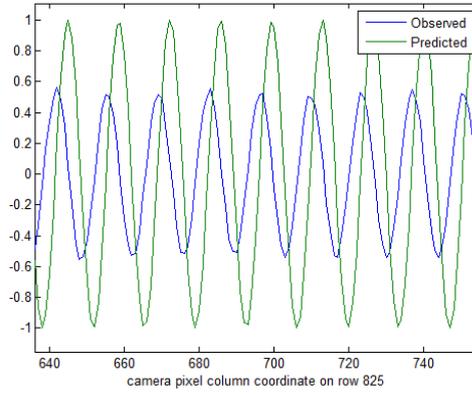


Figure 5.9 – Comparing predicted signal with observed signal.

The depth of the observed scene has no influence on the observed signal phase when projecting a sine with $\omega_p = 0$. Therefore, this correction can be seen as an extra step in calibration and is justified.

Using the observed image from experiment 31, the phase is estimated. The difference with the predicted phase $\omega_o x$ is divided by ω_o and a correction for the camera rectified coordinates is obtained for each camera pixel:

$$x_{correct} = \frac{\text{estimated phase(experiment 31)} - \omega_o x}{\omega_o} \quad 5.6$$

This correction is added to the camera's rectified coordinates and the depth map is computed once more. The new results are shown in Figure 5.10.

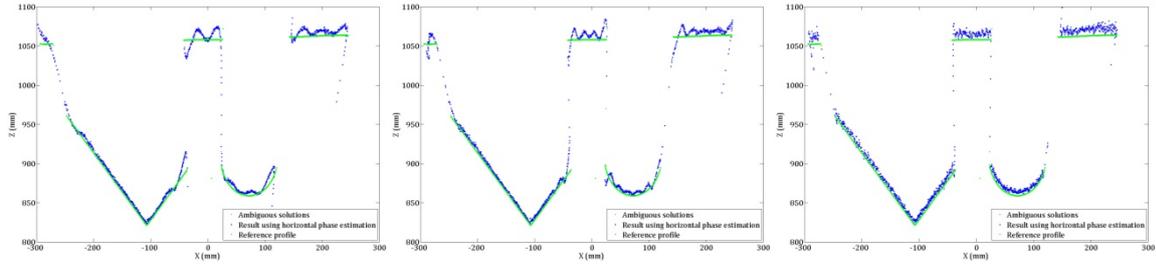


Figure 5.10 – Results of experiments 17, 19 and 21 after an extra calibration step to correct for errors in centre point and lens distortion estimation.

5.2.6 Final results

Using the extra calibration step as discussed before, the results of experiments 8 and 11 are compared with the results of experiments 21 and 26. The results of experiments 8 and 11 have been presented before in Figure 5.6 and Figure 5.7. In Figure 5.11 and Figure 5.12 they are shown once more, next to the results of experiments 21 and 26.

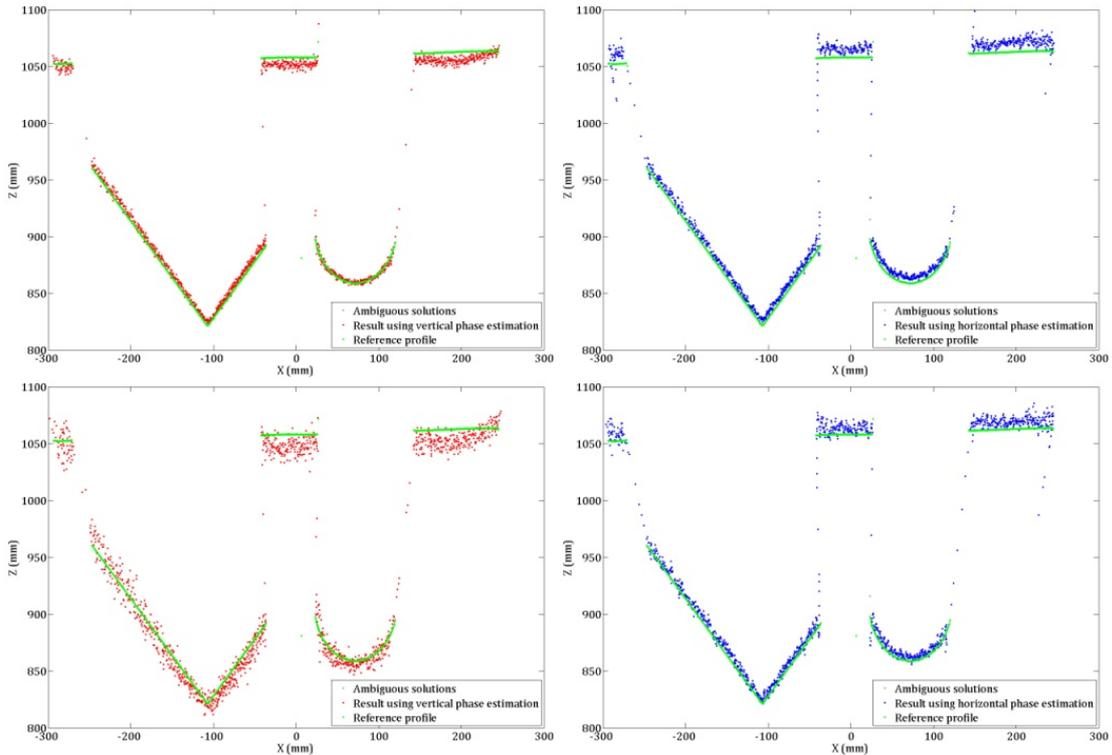


Figure 5.11 – The top two image show the results of experiments 8 and 21, both with the same ω_p . The bottom image show the results of experiments 11 and 26 which also have equal ω_p . In the left images $\omega_o = 0$, in the right images $\omega_o \approx \frac{2\pi}{10}$ rad/pixel. Especially in the bottom images, where ω_p is lowest (250 pixel period), the increase in accuracy visible when examining the spread in depth of the results.

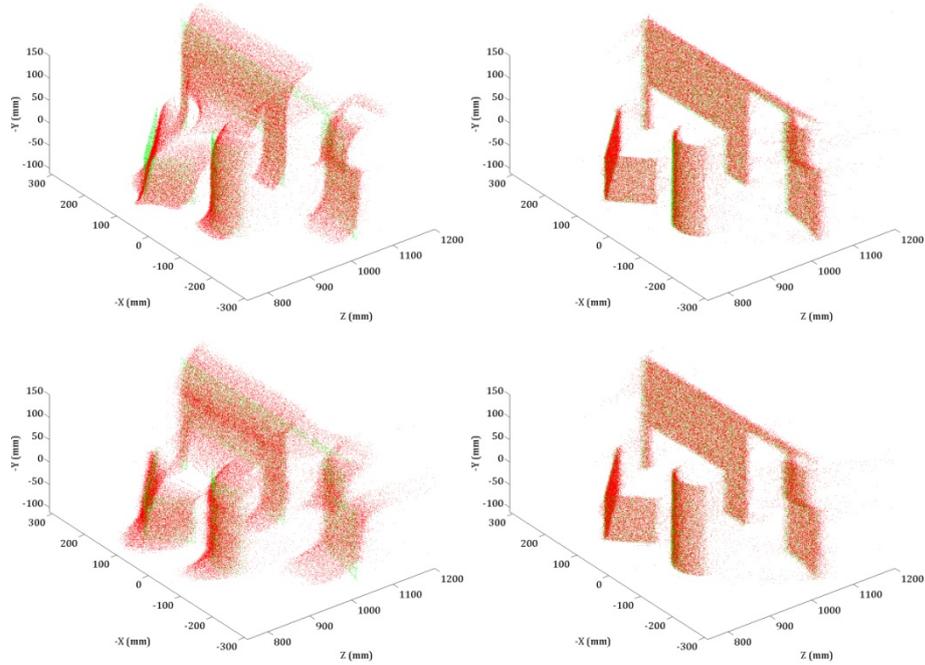


Figure 5.12 – Where in Figure 5.11 the improvements seem moderate, the effect on the total depth map is clear when presenting the entire point cloud. The new method is much less sensitive to discontinuities in depth.

5.3 Comparison and discussion

5.3.1 Comparison with reference depth map

Visual inspection

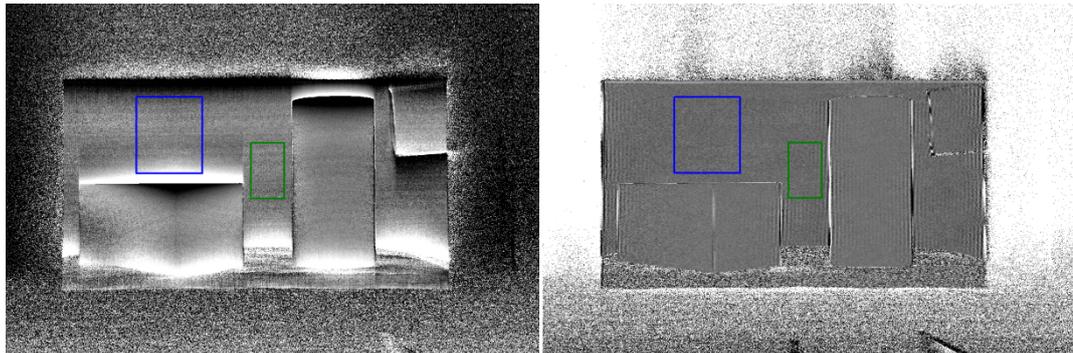


Figure 5.13 – Difference between reference depth map and the results of experiments 11 (left) and 26 (right). The shown range is -30 (black) to 30 mm (white). The blue rectangle (left in each image) indicates an area where random errors dominate errors due to edges in case of horizontal phase estimation. The green rectangle indicates a similar area for vertical phase estimation.

For a certain row, Figure 5.11 shows how the addition of a frequency component in the orthogonal direction increases the accuracy of the depth measurement. Figure 5.12 shows the nature of the error by presenting a 3D point cloud. With $\omega_o = 0$, large errors are made near depth discontinuities in the vertical direction. Estimating the phase horizontally, using a high ω_o fails

near discontinuities in the horizontal direction. However, the area of the errors remains small. This can be seen most clearly in the error maps presented in Figure 5.13, where the experimental results are subtracted from the reference depth map.

The reader can conclude from the figure that the errors due to discontinuities are much smaller when using a high ω_o . To compare the random errors, areas are selected that are not near any discontinuities. These areas are marked by the rectangles in Figure 5.13. The error values of these two particular experiments are presented in a histogram in Figure 5.14.

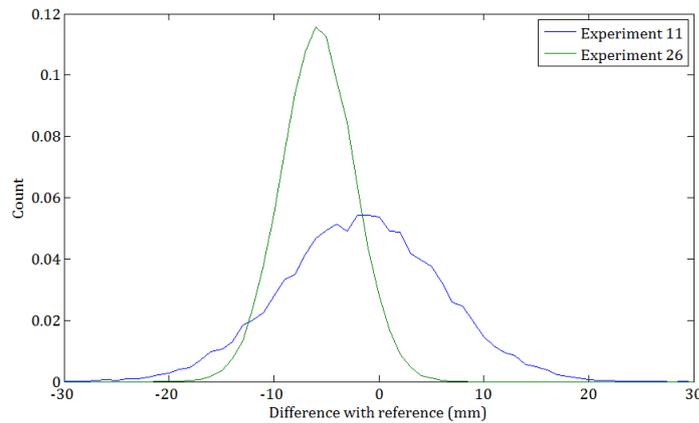


Figure 5.14 - Error histograms of the two methods for the pixels in the area indicated with the red rectangle in Figure 5.13. $[\mu, \sigma]_{11} = [-1.2, 7.5]\text{mm}$ $[\mu, \sigma]_{26} = [-5.3, 3.4]\text{mm}$.

The random error made by the new method has a much smaller standard deviation; 3.4mm compared to 7.5mm when using $\omega_o = 0$. But the difference between the means is also clearly visible. While the naive method has -1.2mm mean error, the new method has an average error of -5.3mm.

Analysis

Also interesting to see is the decrease of the accuracy as ω_p decreases, by analysing the results of experiments 1 to 11. It becomes clear from Figure 5.15 that the accuracy decreases exponentially as the period becomes larger. It is also clear that there is a permanent offset between the reference and the new method. The offset in depth is about -2.5 mm.

For the frequency in the phase direction used in experiments 5, 8 and 11, experiment are done with orthogonal frequency components as indicated in Table 5.1. When an orthogonal frequency component is used, the phase is estimated in the horizontal direction. The mean and standard deviations of these experiments are shown in Figure 5.16. In that figure they are compared with their equivalent experiment without orthogonal frequency component using vertical phase estimation.

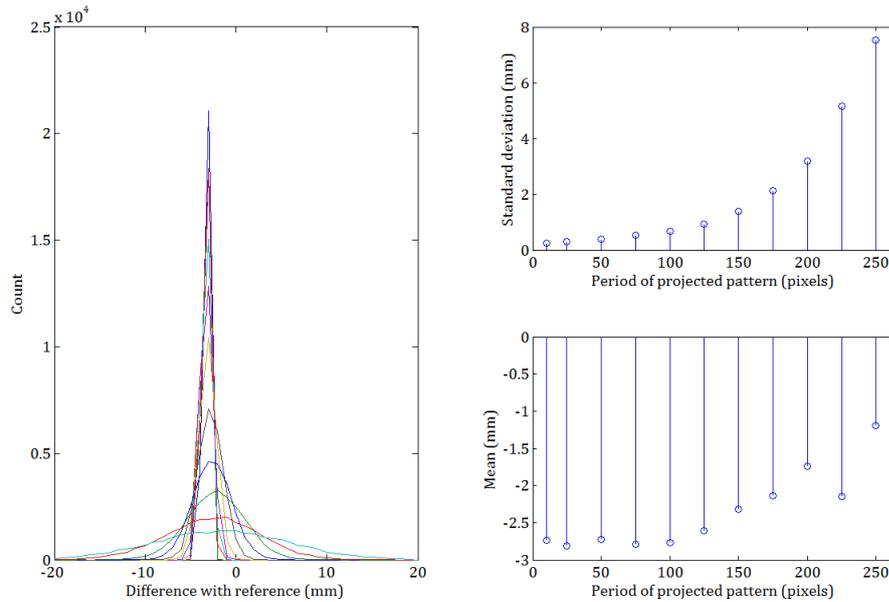


Figure 5.15 – Difference with the reference result for the first 11 experiments. The standard deviations and means of the experiments are shown on the left, as a function of the used wave period. On the right, the histograms of the errors for each experiment are shown.

Experiments 12 to 16 indicate that the accuracy is slightly worse than when using $\omega_o = 0$. In that case the period of the wave in the phase direction is 100 pixels.

Experiments 17 to 21 show an improvement, judging on the standard deviation, of about 0.5mm when using an orthogonal component having a period larger than 30 pixels. The component in the phase component has a period of 175 pixels in this case.

For very low frequencies, with a period of 250 pixels in the phase direction, the addition of an orthogonal component improves the accuracy by over 5mm. This is shown by experiments 22 to 26.

Earlier in Figure 5.15 a systematic error of about -2.5 mm was shown. This systematic error seems to increase when ω_o is set to be non zero and increases even more when ω_o increases. Using equation 3.6, the systematic error can be put into perspective. The rectangles in Figure 5.13 indicate patches on the backplane of the rig. This surface is at a distance of about 1050mm from the camera. The camera-projector base line is about 205mm. This results in an error sensitivity of

$$\frac{\delta Z}{\delta e} = \frac{Z^2}{T_y} \approx \frac{1050^2}{205} \approx 5.3 \cdot 10^3 \quad 5.7$$

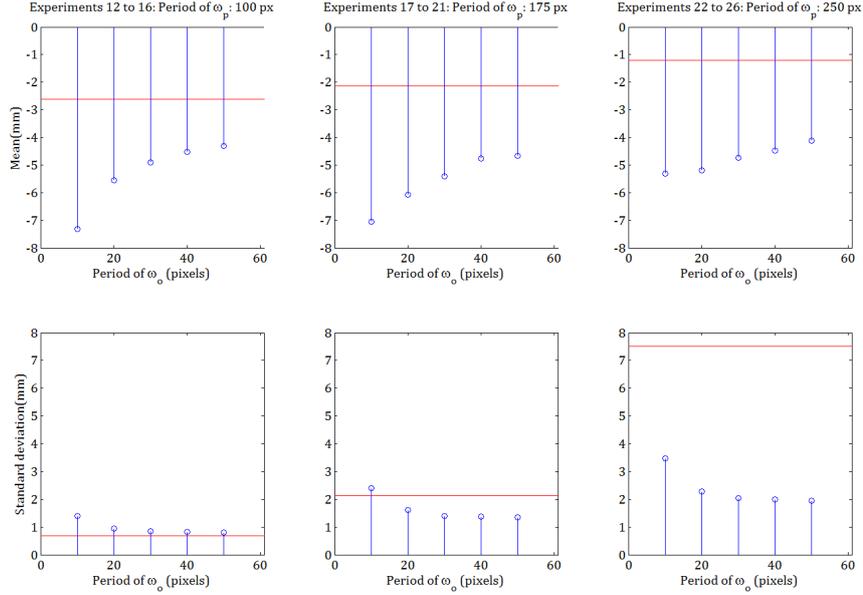


Figure 5.16 – The means and standard deviations of experiments 12 to 26. The results are divided in three groups. Experiments which have the same ω_p are grouped. In the top three plots, the mean is shown. In the bottom three plots the standard deviation is shown. The red lines indicate the performance when $\omega_o = 0$ for the same ω_p , i.e. the results of experiments 5, 8 and 11, from left to right.

In normalized coordinates, the error caused by a one-pixel misalignment in the y -direction is, when the projector focal distance is about 2030,

$$\Delta Z \approx \frac{5.3 \cdot 10^3}{2030} \approx 2.6 \text{ mm} \quad 5.8$$

The ~ 2.5 mm offset in the first 11 experiments seems like a pixel offset. Expansion of parts in the projector due to an increase in temperature could cause this effect or possibly a mistake in the implementation of the pixel mapping procedure that generated the reference depth map.

The systematic error shown by the first 11 experiments could only be caused by an error in the y -direction. Using $\omega_o \neq 0$, errors in the x -direction are increasingly important as ω_o increases. A unit misalignment in the x -direction gives a phase estimation error of ω_o . Using the sensitivity to phase errors given in equation 4.19, this phase error causes a depth estimation error of

$$\frac{\delta Z}{\delta e} = \omega_o \cdot \frac{\delta Z}{\delta \phi} = \omega_o \cdot \frac{Z^2}{T_y \omega_p} = \frac{\omega_o}{\omega_p} \cdot \frac{Z^2}{T_y} \quad 5.9$$

When comparing this expression with equation 5.7, the method is more sensitive to misalignment errors in the orthogonal direction than in the phase direction with a factor ω_o/ω_p . This is usually a large number. In experiment 26 this ratio is largest; $\omega_o/\omega_p = 25$. A pixel misalignment in the x -direction thus causes a $25 \cdot 2.6 \text{ mm} = 65 \text{ mm}$ offset in depth.

The systematic errors in the experiments with $\omega_o \neq 0$ are in the order of mm more. In experiment 26, the mean error is -5.3mm . Compared with the 65mm error in case of a pixel misalignment, the misalignment is under a tenth of a pixel width. This alignment error is kept small due to the extra calibration step presented in section 5.2.5.

Final discussion

The ω_o/ω_p ratio explains why the systematic error grows, as ω_o increases. If a pixel error in the pixel mapping implementation would cause the systematic difference of about 2.5mm in the first 11 experiments, the same type of error would be expected in the experiments with $\omega_o \neq 0$ as well. Since this would cause a systematic difference of 65mm in experiment 26, which is not the case, a pixel error in the reference method is probably not the case. This can be concluded when you consider that the pixel mapping procedure first maps columns and maps rows subsequently in the same manner. An error in the implementation should thus show up in both directions.

More likely is a slight parameter estimation error during calibration or the drift of parameters due to changing temperatures in the rig in the time between the calibration and the measurements. The curtain to darken the inside of the rig needs to be open during calibration in order to present the checkerboards. As the curtain closes during the experiments, the temperature rises significantly within the rig.

The encountered misalignment in the x -direction was corrected for during the measurement, by an extra calibration step. A pattern with $\omega_p = 0$ was projected and the phase was estimated in the observed images. This phase was used to correct the x -component of the rectified camera coordinates, which should be equal to the x -component of the corresponding projector coordinates.

5.4 Conclusion

The experiments presented in this chapter were successful in validating the hypothesis. The addition of an orthogonal frequency component when projecting a sine wave decreases the error significantly. Especially transient effects near discontinuities in depth were diminished. Also a decrease in the standard deviation of the random error was demonstrated. However, the method turns out to be very sensitive to alignment errors of the devices. An increase in the systematic error was observed as ω_o increased.

The method allows a very low frequency component in the phase direction. This makes a large unambiguous measurement space possible. Without the need for phase unwrapping, an object can be reconstructed as long as it fits within such a space. In section 5.1.2 an expression is presented that computes the distance between ambiguous solutions. It also describes the method to pick the right solution after a measurement is done.

6 Conclusion

6.1 Achievements

During this project a series of Matlab scripts is written to form the Structured Light Matlab Toolbox. The existing stereo calibration toolbox of Bouguet is adapted so that a camera-projector set-up can be calibrated both intrinsically and extrinsically, including lens distortions. A method is developed to examine the projector and cameras intensity response function. The used devices proved to respond linearly and therefore a method to compensate non-linear behaviour was not developed. An existing structured light method was adapted to generate a reference depth at the camera resolution with sub millimetre precision, depending on the distance of the object to measure.

In the second part of the project, research was done on phase based profilometry. The hypothesis that depth estimation can be improved when projecting a sinusoidal pattern by adding an orthogonal frequency component was tested and validated with success. For low frequencies in the phase direction, where the space between ambiguous solutions is large, significant improvements were shown. Not only the deviation of the random error decreased with several millimetres, the transient effects caused by discontinuities in depth were practically overcome. Those effects actually make unambiguous phase based depth estimation impossible when not using a high orthogonal frequency component.

6.2 Conclusions

The new profilometry method is thoroughly analysed mathematically allowing for the simulation of an experiment with a given depth map as well as full image 3D reconstruction given a measurement. The range of interest where the objects to reconstruct are located is needed in order to pick the right solution from the ambiguous ones.

The method knows two parameters; ω_p and ω_o . These are the frequency components of a projected frequency in respectively the phase direction and the orthogonal direction. ω_p influences the space between ambiguous solutions as well as the sensitivity to phase estimation errors. The orthogonal frequency component ω_o solely acts as an aid in phase estimation and enables an increase in accuracy when projecting with a low ω_p .

The sensitivity to misalignment errors in the orthogonal direction is a factor ω_o/ω_p larger than misalignment errors in the phase direction. This is usually a large factor as ω_p is required low for less ambiguity and ω_o is required high for better phase estimation. This effect makes the method very sensitive to calibration parameters.

In this project no efforts were made to improve the actual phase estimation in a single shot estimation. A FFT based phase estimator was used and two shots were taken to remove the offset due to ambient light.

For useful values of ω_p , the addition of an orthogonal frequency component with a period of 30 pixels decreases the standard deviation of the random error by 0.5 mm for ω_p with a period of 175 pixels and over 5mm for ω_p with a period of 250 pixels. In both cases the overall error decreased drastically since the transient effects due to depth (and thus phase-) discontinuities decreases when projecting high frequencies.

6.3 Recommendations

With the hardware and rig available from the predecessors, the software created during this project is the second step towards a professional development toolbox. Even though very useful, there is always room for improvements. In the first subsection recommendations for the development platform are presented.

The research done with respect to phase based profilometry also resulted in promising results. The second subsection sums a number of recommendations for future work on the use of an orthogonal frequency component in phase based profilometry.

6.3.1 Development platform

Geometric calibration

- Currently the projected checkerboard corners were used to find a homography from camera to projector. Using this homography, the printed checkerboard corners were transformed to projector coordinates. This eases the incorporation of an existing calibration toolbox. However, due to lens distortions, the usage of the found homography is not totally justified. The corners of the projected checkerboard can also be directly used in the minimization scheme to find the calibration parameters.
- The proposed automatic corner finder is limited as it cannot handle checkerboards that are rotated about 45° . This is no problem when the user is familiar with this limitation and presents the checkerboard patterns properly, but the method is not robust and ill documented. More robust and faster algorithms must exist and it is recommended that such an algorithm is used instead.

Radiometric calibration

- The toolbox comes with a procedure to measure the input-output relation of the projector and the camera. However, if the devices would show non-linear behaviour, it would be the task of the toolbox to correct the projector and camera intensity values. This option is not yet available and should be

implemented. A polynomial model, gamma curve or a look-up table are possible options to correct the values, but this requires some more research.

- For the best result, a strong recommendation is to use a better LUX meter. A better LUX meter should not have to stabilize towards a proper value by continuous polling for measurement values. Also, it should have a consistent offset and slope for different measurement ranges.

Reference depth map

- The propagation of the error in calibration parameters into errors in the depth map is not yet elaborated. This relation would be a useful feature for the toolbox; it allows for a more substantiated comparison with other results.
- Use an accurate 3D model of the test object and use the pixel mapping to position the models and the backplane. Use the properly positioned objects to generate a reference depth map. This should result in a highly accurate reference depth map.

Test objects

- Besides the block and the cylinder, a cone and a pyramid would be a welcome addition. These kinds of objects also present slopes in the vertical direction. The current objects only provide slopes in the horizontal direction.

Software

- The software now consists of a set of Matlab scripts and functions. The user interface is command line based. A user friendly GUI can be designed that guides the user through the procedures and scripts.
- It proves useful to view the live image from the camera, while aligning the devices. Currently, Matlab has to release the camera handle so that the EOS Utility can be used for live view. Even though not a problem, when building a GUI, adding live view in the Matlab environment can be a useful feature.

Rig

- The influences of temperature changes were noticed and even used to explain the systematic errors encountered in this research. The effects of temperature could be examined more by shooting images of a constant projection on the backplane over time, while opening or closing the curtain. The automatic corner finder could be used while projecting a checkerboard, to automate this research. Also a way of temperature stabilisation should be implemented, like a cooling fan, so that no changes are caused by opening or closing the curtain.

6.3.2 New phase based profilometry approach

- The method requires a per-pixel phase estimate and is able to take ω_0 into account when reconstructing depth. The actual means by which the phase

estimate is obtained does not really matter. Currently a 1D phase estimation scheme is used, a 2D phase estimation might result in an even more accurate result.

- Any projector distortions caused by the projector's limited resolution and PSF can be overcome by replacing it with a more natural way to project sine waves: a laser and an optical slit. A high power laser is required, since the camera needs to acquire a considerable amount of light for a proper picture, but the projector sine wave is pure and continuous.
- The basic phase estimation scheme used in this research should be replaced by a more complex one that allows jumps and changes in albedo. [1] has great potential when combined with the results of this research. The space between ambiguous solutions is large, so the particles' probability can be set zero outside a certain non-ambiguous range. Using the approach of [1], the surface slope can be tracked and the observed pixel intensity can be predicted accordingly, using equations 4.1 and 4.2, using any desirable periodic function $f(t)$ instead of just a sine wave.
- Using two images instead of a single shot, enables the projection of the sine wave and the same wave with a π phase shift. By means of subtraction, the offset due to ambient light is removed, which enhances the phase estimation results significantly. The two shots could also be combined using colour. This makes it a signal shot method, given that the objects are colour neutral.
- During the experiments, the method was found to be sensitive to alignment estimation errors in the x -direction. Therefore, a reference shot with $\omega_p = 0$ was required for extra accuracy. A phase based profilometry method is presented in [22], and uses reference planes to calibrate the system. This idea could be combined with the current method, making geometric calibration unnecessary.
- When using a reference plane for calibration purposes, the proposed method starts to look like a Fourier Transform Profilometry approach. It may be interesting to investigate the combination of the addition of an orthogonal frequency component and FTP.

A Appendix A - Automatic corner finder

In order to locate the corners automatically, the following algorithm is applied to an image. The found corners positions are not accurate, but accuracy is not required as the result of this algorithm is refined by an existing corner finder implemented by Bouguet in [14].

To locate $N \times M$ checkerboard corners:

1. Of the input image I find the cross derivative I_{xy} .
To do so, a kernel is built of the cross derivative of a gauss and used to filter the image. This filter matches a checkerboard corner and will generate a high positive output for the 'white-black-black-white' type of corners and very negative output for the 'black-white-white-black' type of corners.
2. Find the local maxima and minima in the image. Keep only the top O maxima and bottom O minima. $O = 2$ times the number of corners to be found.
3. Pick a random maximum and find the four closest minima.
4. Use the four minima locations to compute a homography.
5. Iterate until no more grid coordinates are found:
 - a. predict more grid locations with the found homography
 - b. find the closest maxima or minima locations near the predicted locations and store as "corresponding points"
 - c. delete "corresponding points" that differ over 3 pixels with their predicted counterpart
 - d. use the remaining coordinates to re-compute a homography
6. If the amount of connected points is smaller than $M \times N$, start over from step 3. After 25 tries, the algorithm fails.
7. If the amount of connected points is larger than $M \times N$, the external corners of the checkerboard are found as well. Select only the part that is as large as the expected grid and return the found grid locations.

Note that the grid may not be rotated too much, or the algorithm will fail. When rotated 45° , the cross derivative kernel does no longer respond to the corners.

The function is implemented in `get_grid.m`.

B Appendix B – Structured Light Toolbox for Matlab Manual

The Structured Light Toolbox for Matlab (SLT) is a combination of scripts and functions for Matlab to ease the development of a structured light system. The theory behind the functions and procedures is already described in chapter 3. This appendix presents a manual on how to operate the toolbox.

B.1 Toolbox setup

Installation

Installation on a new system is done three steps:

- *Add the folder containing the toolbox files to the Matlab path.*
In Matlab, select: File → Set path... → Click “Add with subfolders...”. Now provide the path of the toolbox folder. Finally, save the new search pad by clicking “Save” in the bottom left corner of the window.
- *Recompile mex-files.*
For the image acquisition and the projection of images, three mex-files are created. On a different system or a different version of Matlab, these files may need to be recompiled:
 - Go to the subfolder “\Required functions\Canon EOS Interface” in the toolbox folder. Now run `recompile_canon.m`.
 - Go to the subfolder “\Required functions\Projector Interface” in the toolbox folder. Now run `recompile_beamer.m`.
 - Go to the subfolder “\Required functions” in the toolbox folder. Now run `recompile_dmatch.m`.
- *Set the calibration results folder.*
A calibration procedure requires the acquisition of a lot of images and thus a lot of storage space. In order to not save these files in the current directory, the toolbox must have a storage path for calibration results.
 - Start the toolbox by issuing the command `start_toolbox`.
 - Provide the calibrations path to the function:
`set_calibrations_path(new_path)`
 - If no path is provided, “C:\Temp\Calibrations\” will be used instead.

Installation is now complete. The toolbox can be started by the command `start_toolbox`. This ensures the existence of some critical variables in the workspace. **Note that starting the toolbox will clear your workspace!**

Storing and loading calibration results

When the toolbox is started, a variable called `calibrations` is available in the workspace. This is a struct with 5 sub variables:

```
calibrations =  
    geometric: ''  
    dark_frame: ''  
    linearity: ''  
    scenic: ''  
    reconstruction: ''
```

When a calibration procedure is performed, the calibration result is assigned a name by the user. Set the struct field of the `calibrations` variable with that name. Not all fields need to be filled in. Then run: `reload_calibrations`. The calibration results are now loaded into the workspace.

`reload_calibrations` looks for the calibration result in the folder provided during installation of the toolbox. Use `set_calibrations_path` to alter this path.

B.2 Geometric calibration and coordinate rectification

Calibration

The geometric calibration procedure is performed and guided by the script `geometric_calibration`. Run this script to start the procedure. Several inputs are expected from the user, provide when requested:

- A name for the calibration, to identify and reload the results.
- The dimensions of the printed checkerboard.
- The size of a checkerboard square. The user may decide the unit in which to provide the size, this unit will be incorporated in the focal distances of the devices and eventually the reconstruction results.

When all necessary information is provided, the procedure will begin. The following steps are repeated 18 times:

- A blue checkerboard is projected. “Catch” the projected checkerboard with the printed checkerboard (e.g. like in Figure 3.4). You have approximately three seconds to position the printed checkerboard; the count-down is audible by a ticking sound. Only for the first capture you have 8 seconds.
- The image is shot and will be processed immediately. This takes some time, but some time is also needed for the user reposition the calibration grid. After three “ticks” a new image will be shot.
- After successful processing – i.e. when the grid corners are automatically found – a bell sound is audible. When no sound is heard, the grid detection failed, probably due to a bad image. This image will automatically be re-taken.

The “bell-sound” will be heard 18 times, if all is well this required 18 images, more if the grid detection failed in some cases.

After the images are taken, the script will continue directly with the calibrations. In about 10 to 20 seconds, the calibration results are available and stored.

Rectification

If necessary, reload the geometric calibration results. Now simply type `normalise_and_rectify` in the command window. The following mat-files are now created in the current path:

- `normalised_projector_coordinates.mat`
This file contains the 768 by 1024 matrices `pxn` and `pyn`, which contain the normalized projector coordinates for each projector pixel.
- `normalised_camera_coordinates.mat`
The file contains the 1301 by 1954 matrices `cxn` and `cyn`, which contain the normalised camera coordinates of the lower left green pixels of the Bayer tiles of the camera.
- `rectified_projector_coordinates.mat`
The file contains the 768 by 1024 matrices `pxr` and `pyr`, which contain the rectified projector coordinates.
- `rectified_camera_coordinates.mat`
The file contains the 1301 by 1954 matrices `cxr` and `cyr`, which contain the rectified camera coordinates of the lower left green pixels of the Bayer tiles of the camera.

The last two files also both contain:

- `T_rect`: the relative translation of the two devices in the rectified setup.
- `R_rect`: the rotation matrix of the rectified camera relative to its actual orientation. To compare 3D coordinates computed with the rectified setup, these have to be rotated by `R_rect` to match the results of the reference depth map.

B.3 Pixel mapping procedure and 3D reconstruction

Pixel mapping procedure

The pixel mapping procedure is started by the command `pixel_mapping_procedure`. A name is requested to store the results. No further input is required.

The script will start acquiring images. After image acquisition, the images are processed. In some cases, an Out Of Memory error can occur during the fitting of the quadratic function. Using the `clear` or `pack` commands, Matlab can free memory, but sometimes restarting Matlab is required. In any case, the command `pixel_mapping_procedure` must be issued once more. When providing the same

name as before, the script recognizes which (if not all) images are already acquired and will continue processing immediately.

3D reconstruction

`do_reconstruction` is the script to run to generate the reference depth map or point cloud. Make sure to first load the geometric calibration result as well as the pixel mapping result, by setting `calibrations.geometric` and `calibrations.reconstruction` with the correct names and calling `reload_calibrations`.

The result is stored with the pixel mapping results and can be reloaded via `calibrations.reconstruction`. The results are:

- `Xt`: a 1301x1954x3 matrix containing a 3D coordinate for each camera pixel. `Xt(:, :, 1)` contains the x-component, `Xt(:, :, 2)` the y-component and `Xt(:, :, 3)` the z-component. `Xt(:, :, 3)` is thus also the depth map.
- `pxr_c`: the rectified corresponding projector x-coordinate for each camera pixel.
- `pyr_c`: the rectified corresponding projector y-coordinate for each camera pixel.

B.4 Radiometric calibration

Linearity measurements

This part of the toolbox focuses on the acquisition of the input-output relations of the projector and camera. This part is very specific for the LUX-meter that is currently used and may be completely rewritten when another meter is used.

Perform the following steps:

- Start the toolbox if necessary by typing `start_toolbox`.
- Make sure the toolbox projection window is active by typing `sbeam(1)`. The projector should project a fully white screen.
- Place the LUX meter on the backplane.
- Make sure the variable `mouse_click` is empty or does not exist by typing `clear mouse_click`.
- Move the mouse cursor onto the projector screen and click on the location of LUX meter in the window. This creates the variable `mouse_click` in the workspace.
- Start the measurement procedure by the command `lux_procedure` and wait for it to finish.
 - **NOTE!** When aborting the procedure by using Ctrl-C, or in case of an error, issue the command `stop(measurement_timer)`. The procedure uses a timer structure in order to keep polling the LUX meter properly. This is a necessary evil as the LUX meter will need to restart when the polling stops temporarily.

Dark frame acquisition

The acquisition of a dark frame is simply done by the command `dark_frame_analysis`. Besides resulting in the mean dark frame, the variation of the pixel values is also computed. Make sure the scene is absolutely dark or the lens is capped before starting. Like other calibrations, the acquired dark frame is assigned a name which has to be provided by the user when starting the script.

B.5 Additional functions

Image acquisition

To take an image, make sure the camera is initialized. When starting the toolbox, the camera has already initialised. When this failed, because the camera was not connected or turned on at that time, or you do not want to use the toolbox, use: `canon('init')`.

For debug purposes, all actions of the camera are shown in the command window. In case of an error, the error code is shown for reference in the Canon Software Development Kit. To turn off these messages: `canon('debug', false)`.

To take an image, type `canon('shoot', 'filename')`. `filename` must be less than 10 characters long, may not contain a path and may not end with an extension. The camera shoots and downloads `filename.cr2` and `filename.jpg` from the camera into the current directory.

Note that Matlab is able to continue while the image is still being taken or being downloaded. Opening `filename.jpg` directly after shooting (e.g. in a script) will result in a "File Not Found" error. To make sure your script only continues when the file is downloaded, use `cshoot('filename')` instead. Or add the line:

```
while(~exist('filename','file')), pause(0.1); end;
```

The `pause` statement is required! The camera tool must be able to interrupt Matlab for the download to start and complete.

`cshoot` has more modes of operation, type `help cshoot` for more information.

When necessary, release the camera handle with the command `canon('exit')`.

Raw image conversion

The raw CR2 files returned by the camera cannot be read directly by Matlab. Use `rawread` to convert the files to readable TIFF files. `rawread` is a wrapper function for the external executable `dcraw.exe` [12]. Type `help rawread` for the possibilities.

Image projection

An add-on is written for Matlab to display an image on a secondary (or tertiary, etc.) display. The projector is connected and mapped by Windows as a secondary display in 'Extended Desktop Mode'.

`beamer(I)` finds the most right display that is connected, creates a borderless window there and maximizes. In that window the function displays the image `I`, cropping if larger than the screen resolution. If the image is smaller than the screen size, only the part of the image is refreshed. `size(I,3)` must be 3 (colour planes) and the class must be `uint8`.

Since the requirements for `beamer` are rather strict, use the wrapper function `sbeam` instead. `sbeam` can handle grayscale images, `double` images and is able to scale intensities, like the Matlab command `imshow`. By default, the shown intensity range is `[0 1]`. Use `sbeam(I,[])` to use `[min(I(:)) max(I(:))]` as the display range. Type `help sbeam` for more options.

Direct and global illumination component separation

The procedure described in [11] is implemented in the Matlab script `light_component_separation`. It separates the direct and global component of the observed light. The direct component is the part of the light that directly originates from the projector. The global component is the amount of light due to inter reflections and ambient light.

It is required that the dark frame `D` is available and thus a dark frame analysis should be run first, or (re)loaded via `calibrations.dark_frame` and `reload_calibrations`.

The results are stored and can be loaded by setting `calibrations.scenic` with the proper name and then running `reload_calibrations`.

Bibliography

- [1] F.F. Berendsen, "One-Shot Structured-Light Range-Imaging Using Particle Filters," University of Twente, Enschede, Masterthesis SAS004-09, 2009.
- [2] A.C. Nijmeijer, "One-shot 3D reconstruction using instantaneous frequencies," University of Twente, Enschede, Masterthesis SAS012-08, 2008.
- [3] J. Salvi, J. Pagès, and J. Batlle, "Pattern codification strategies in structures light systems," *Pattern Recognition*, vol. 37, pp. 827-849, 2004.
- [4] J. Gühring, "Dense 3D surface acquisition by structured light using off-the-shelf components," in *Proceedings of Videometrics and Optical Methods for 3D Shape Measurement*, Stuttgart, Germany, 2001, pp. 220-231.
- [5] P. Fechteler and P. Eisert, "Adaptive Colour Classification for Structured Light Systems," *IET Computer Vision*, vol. 3, no. 2, pp. 49-59, June 2009.
- [6] X. Su and W. Chen, "Fourier transform profilometry: A review," *Optics and Lasers in Engineering*, vol. 35, pp. 263-284, May 2001.
- [7] S.S. Grothi and K.R. Lolla, "Novel Single Shot Structured Light Technique for Accurate, Reliable and Dense 3D SHape Measurement," in *Proceedings of International Conference on Optics & Optoelectronics*, Dehradun, 2005.
- [8] J. Heikkilä and O. Silvén, "A Four-step Camera Calibration Procedure with Implicit Image Correction," in *Proceedings of Computer Vision and Pattern Recognition*, San Juan, 1997, pp. 1106-1112.
- [9] D.A Forsyth and J Ponce, *Computer Vision*. New Jersey, U.S.A.: Prentice-Hall, 2003, ch. 4, pp. 63-67.
- [10] D. B. Goldman and J. Chen, "Vignette and Exposure Calibration and Compensation," in *Proceedings of 10th IEEE International Conference on Computer Vision ICCV 2005*, Beijing, 2005, pp. 899-906.
- [11] S.K. Nayar, G. Krishnan, M.D. Grossberg, and R. Raskar, "Fast Separation of Direct and Global Component of a Scene using High Frequency Illumination," *ACM Transactions on Graphics*, vol. 25, pp. 935-944, 2006.

- [12] Dave Coffin. (2009, December) Pukkita's Digital Darkroom Corner. [Online]. <http://www.insflug.org/raw/>
- [13] Z Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations," in *ICCV*, 1999, pp. 666-673.
- [14] J. Bouguet. (2008, June) Camera Calibration Toolbox for Matlab. [Online]. http://www.vision.caltech.edu/bouguetj/calib_doc/
- [15] R. Hartley and A Zisserman, *Multiple View Geometry*. Cambridge, United Kingdom: Cambridge University Press, 2000, ch. 7, pp. 179-180.
- [16] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330-1334, 2000.
- [17] R. Van Koten and A. Keemink, "Calibration of a structured light system," University of Twente, Enschede, 2009.
- [18] C Mauer, "Measurement of the spectral response of digital cameras with a set of interference filters.," University of Applied Sciences Cologne, Cologne, Thesis 2009.
- [19] Y. Xu and D.G. Aliaga, "Robust Pixel Classification for 3D Modeling with Structured Light," in *Proceedings of Graphics Interface 2007*, New York, 2007, pp. 233-240.
- [20] Y. Xu and D.G. Aliaga, "An adaptive Correspondence Algorithm for Modeling Scenes with Strong Interreflections," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 3, pp. 465-480, June 2009.
- [21] C. Guan, L.G. Hassebrook, and D.L. Lau, "Composite structured light pattern for three-dimensional video," *Optics Express*, vol. 11, no. 5, pp. 406-417, 2003.
- [22] J.F. Cuevas, M. Servin, and R. Rodriguez-Vera, "Depth object recovery using radial basis functions," *Optics Communications*, vol. 163, pp. 270-277, May 1999.
- [23] K.S. Shanmugan and A.M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*, Wiley, Ed.: Wiley, 1988, ch. 5, pp. 323-325.