

Dissertation
submitted to the
Combined Faculties for the Natural Sciences and
for Mathematics
of the Ruperto-Carola University of Heidelberg,
Germany
for the degree of
Doctor of Natural Sciences

Put forward by

Diplom-Physicist: Henrik Schäfer
Born in: Stuttgart-Bad Cannstatt, Germany

Oral examination: 26.11.2014

Image Enhancement and Parameter Estimation for Time-of-Flight Cameras

Referees: PD Dr. Christoph Garbe
Prof. Dr. Karl-Heinz Brenner

Danksagung

Ich möchte mich zuallererst bei PD Dr. Christoph Garbe bedanken, der mir die Gelegenheit gegeben hat auf diesem spannenden Gebiet zu arbeiten. Er ließ mir die Freiheit allen Ideen nachzugehen, lenkte Ausschweifungen aber mit gutem Rat zurück auf Kurs.

Vielen Dank auch an Prof. Dr. Karl-Heinz Brenner, der sich bereit erklärt hat das Zweitgutachten zu übernehmen.

Weiterhin danke ich Dr. Frank Lenzen, der stets zur Stelle war um Details zu besprechen und Feedback zu geben, auch über die gemeinsame Arbeit hinaus.

Auch Rahul Nair, Dr. Stephan Meister, Moritz Becker, Jens-Malte Gottfried und Dr. Daniel Kondermann haben meine Arbeit am ToF-Projekt über das fachliche Maß hinaus bereichert.

Natürlich gilt mein Dank ebenso Gerald Mwangi mit dem ich mir das Büro teilte und dessen Füße gelegentlich unter meinen leiden mussten, sowie umgekehrt. Über dem Tisch blieb es aber stets harmonisch.

Meinen Kollegen und regelmäßigen Badminton-Partnern Masato Takami, Jose Rubio und Nikolai Ufer sei ebenfalls gedankt, nicht nur für den wöchentlichen sportlichen Ausgleich.

An dieser Stelle sollen auch alle anderen Mitarbeiter, Doktoranden, Diplomanden, Master- und Bachelor-Studenten und Praktikanten des HCI und der übrigen in den Räumlichkeiten des 2. und 3. Stocks untergebrachten Gruppen gewürdigt werden. Insbesondere die Besatzung der Sekretariate, die uns die schweren Mühlen der Bürokratie zu einer Spieluhr machen, sowie die selbstlosen Menschen die dafür sorgen, dass der Koffein-Nachschub nicht unterbrochen wird. Man konnte hier stets in freundliche Gesichter blicken oder ein paar Worte wechseln und in einer angenehmen Atmosphäre seiner Arbeit nachgehen.

Herzlichen Dank auch an das Intel Visual Computing Institute in Saarbrücken, das diese Arbeit zum größten Teil finanziert hat.

Mein besonderer Dank gilt selbstverständlich auch meinen Eltern, die es geschafft haben ihren drei Kindern eine akademische Ausbildung zu ermöglichen. Eine Chance die sie selbst nicht hatten.

Zu guter Letzt darf meine Liebe Andrea nicht unerwähnt bleiben, die mich immer unterstützt hat und mir mit ihrem unerschütterlichen Vertrauen Kraft gegeben hat, besonders in den letzten schweren Tagen.

ZUSAMMENFASSUNG: Bildverbesserung und Parameter Schätzung für Time-of-Flight Kameras

Die Bestimmung der räumlichen Tiefe einer Szene, die für eine Reihe von Anwendungen wichtige Informationen liefert, war mit bisherigen Technologien wie Stereo-Kameras oder Laser-Scannern sehr aufwendig und bisweilen problematisch. Mit Time-of-Flight Kameras steht heute eine Technologie zur Verfügung, die das Erfassen dieser Daten erheblich vereinfacht. Mit dem Microsoft Kinect 2 Sensor haben ToF Kameras kürzlich den Sprung in den Consumer-Markt geschafft. Leider weisen diese Kameras derzeit aber noch verschiedene statistische und systematische Fehlerquellen auf. Diese Arbeit zeigt wie die statistischen Fehler mit Hilfe von Methoden aus der 2D Bildverarbeitung behoben werden können und liefert zum ersten Mal Erklärungen für bisher nicht verstandene systematische Fehler durch eine genaue Analyse der Sensor-Daten. Mit neu eingeführten Ansätzen und Modellen können diese Fehler mit geringem Aufwand deutlich reduziert werden und führen zu einer signifikant verbesserten Genauigkeit der Messungen.

ABSTRACT: Image Enhancement and Parameter Estimation for Time-of-Flight Cameras

The determination of the spatial depth of a scene, which supplies important information for a number of applications, was a very complex matter with previous technologies like stereo cameras or laser scanners and sometimes problematic. With Time-of-Flight cameras there is a technology at hand today, that facilitates the acquisition of these data tremendously. With the Microsoft Kinect 2 Sensor ToF cameras have recently made the jump into the consumer market. Unfortunately these cameras still exhibit different statistical and systematic errors. This thesis shows how the statistical errors can be fixed with the help of methods from 2D image processing and gives for the first time explanations of previously not understood systematic errors by means of an exact analysis of the sensor data. With newly introduced approaches and methods these errors can be reduced notably with little effort and lead to a significant improvement in the accuracy of the measurements.

Contents

1	Introduction	11
1.1	History	11
1.2	Motivation	13
1.3	Outline	14
1.4	Camera	15
1.5	Validation	15
2	Theory	17
2.1	Time-of-Flight Principle	17
2.2	Internal Convolution	18
2.3	Poisson Noise	20
3	Bilateral Filtering	23
3.1	Standard Bilateral Filter	24
3.2	Joint Bilateral Filter	25
3.3	Adaptive Bilateral Filter	26
3.4	Discussion	27
4	Edge Detection	29
4.1	Structure Tensor	29
4.1.1	Theory	30
4.1.2	Structure Tensor and Time-of-Flight Data	31
4.1.3	Application	33
4.2	Structure Tensor and Canny Combined	33
4.2.1	Canny’s Algorithm	33
4.2.2	Adapting Canny’s Algorithm for ToF Data	34
4.3	Shadow Removal	35
4.3.1	Finding the Shadow Edges	35
4.3.2	Smartly Combining Intensity and Depth Edges	36
4.4	Texture Removal	37
4.5	Discussion	39
5	Systematic Errors	41
5.1	LED Modulation	41
5.2	Heat Model	45
5.3	Dark Signal	49
5.4	Signal Nonlinearity	50

5.5	Subframe Offset	52
5.6	Effects on the Phase Calculation	56
6	Calibration	61
6.1	Calibration Model	61
6.2	Experiments and Discussion	62
7	Internal Scattering	67
7.1	Decomposing the Signal	68
7.2	Measuring the Scattering Parameter	69
7.3	Experiments	71
7.4	Discussion	72
8	Conclusion	79
8.1	Summary	79
8.2	Outlook	81
A	Deductions	83
A.1	Intensity, Amplitude and Phase	83
A.2	Poisson Noise	86

“Sagredo: But of what kind and how great must we consider this speed of light to be? Is it instantaneous or momentary or does it like other motions require time? Can we not decide this by experiment?”

“Salviati: If not instantaneous it is extraordinarily rapid - I should call it momentary.”

from “Dialogues Concerning Two New Sciences”
by Galileo Galilei

1

Introduction

1.1 History

In a way, the first single pixel time-of-flight camera (cf. fig. 1.1) in history was built in 1849 by the French scientist Hippolyte Fizeau, when he designed the first known and working experiment to measure the speed of light on earth [10], which Galilei had tried more than two centuries before without success [14].

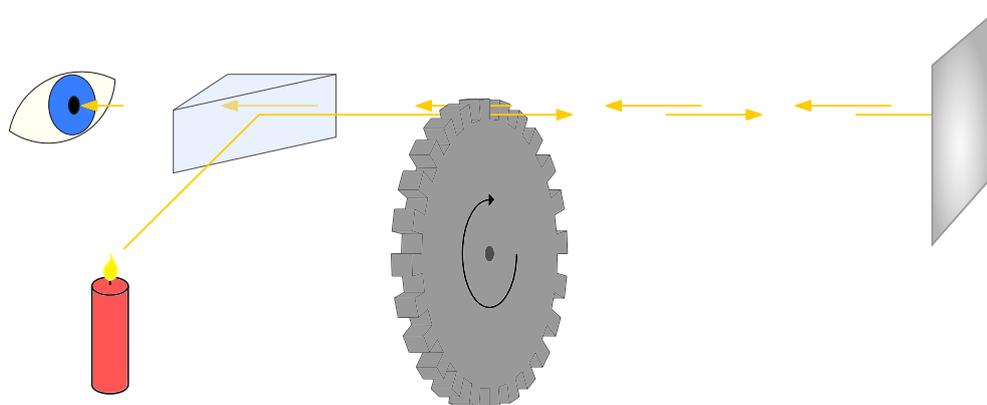


Figure 1.1: Sketch of an experiment by Fizeau in 1849 to determine the speed of light [10]. What is actually measured is the time of flight of the light signal from the cogwheel to the mirror and back. The principle is astoundingly similar to ToF cameras used today.

Fizeau put a strong light source behind the cogs of a rotating cogwheel and observed the light reflected by a distant mirror from behind the rotating cogs as well. With increasing rotational frequency the reflected light would be completely occluded by the tooth following the gap and reach the maximum intensity again at a frequency twice as high. With the number of the cogs known and by measuring

the rotational frequency of the wheel, he did essentially determine the time of flight of the light signal from the cogwheel to the mirror and back. With the distance known, the time of flight can be used to calculate the speed of light.

The speed of light is a well known scientific constant today and measured with such a precision that the meter is actually defined by the speed of light and the second. With the speed of light known, the time of flight can now be used to calculate distances, which is exactly what Time-of-Flight (ToF) cameras do and they use a principle that is astoundingly similar to the antique setup of Fizeau.

In the proposition for a ToF camera in 1995 by Schwarte et al. [51], the cogwheel is replaced with a radio frequency modulated Pockels Cell that controls the intensity of the emitted and received light. Instead of visual inspection the reflected light is recorded with a CCD chip. With the appropriate optics this allows the recording of an entire scene at once.

The technology has come a long way since its introduction in the 90s. Instead of an optical modulation, the radio frequency signal is applied directly to the light source and also the sensor, altering the sensitivity. A large diversity of models has sprouted from these advances, with ToF cameras as small as webcams, integrated in notebooks or large high resolution models with very strong illumination and long reach (see fig. 1.2).

With the introduction of the Microsoft Kinect v2 as a gesture and motion control for the new gaming and multimedia console Xbox One in 2013, millions of ToF cameras reached the consumer market and private living rooms. It supersedes the previous Kinect, which was not a Time-of-Flight camera but a structured light depth sensor and broke the record of the fastest selling electronic device, selling more than 8 million times in the first 60 days [55]. The move to Time-of-Flight can be seen as a breakthrough for the technology.



Figure 1.2: A selection of current Time-of-Flight cameras. The CamCube 3 uses large LED arrays, while the other models have only a single LED to illuminate the scene. The Creative Interactive Gesture Camera (IGC) is a typical webcam like design with an additional RGB camera built in.

1.2 Motivation

Time-of-Flight cameras have a couple of advantages compared to other distance capturing devices such as stereo cameras or laser scanners.

Stereo cameras for example use very simple hardware but require complex image processing algorithms to find correspondences in the two pictures and calculate the depth from the disparity. In areas without texture or other suitable features they have no means to determine the distance. The two cameras required to record the data have to be aligned and calibrated precisely and need to have a certain distance or baseline, circumventing a really compact system.

Laser scanners on the other hand are very complex and expensive devices. The data they acquire is not a dense sampling, but only a sparsely spread number of points or lines, which is subject to interpretation if a dense representation of the scene is supposed to be reconstructed from it.

In contrast to those devices, Time-of-Flight cameras can be manufactured at low cost and built in a compactness that allows the use even in endoscopic surgery. There are no moving parts involved and the configuration of light source and camera is only of minor importance for the distance measurements. The data is recorded by an image sensor and represents a dense sampling of the scene (cf. fig. 1.3). The distance determination also works on untextured surfaces and the computational effort to process the data is minimal.

These properties make ToF cameras a perfect device for applications such as gesture recognition for touchless human computer interaction, pose estimation for motion capturing or medical surveillance, as well as 3D scene or object reconstruction.

But despite years of research and the growing distribution and availability, the accuracy and reproducibility of ToF measurements is still inhibited by a number of errors. While the statistical errors are well understood and share the same origin as the noise in ordinary intensity images, the resulting noise in the depth images has some special properties that limit the use of standard image processing methods.



Figure 1.3: Color coded depth image.

Among the systematic errors there is a number of errors of yet unknown origin. They cause deviations in the measurements, depending on environmental, scene or camera parameters and without an idea of the source of these errors they can only be calibrated or corrected by extensive lookup tables or by fitting heuristic functions which both require large amounts of reference measurements and have to be repeated regularly.

The goal of this thesis is to provide image processing methods that successfully cope with the special characteristics of ToF depth data and to finally uncover the remaining sources of the systematic errors that have been puzzling the community for years.

1.3 Outline

Chap. 2 briefly repeats the theory of the ToF raw data processing and the propagation of the statistical errors.

The resulting noise in the depth data in general is quite large and a good smoothing or denoising is desirable. A very common method to do so is the bilateral filter, which is discussed in chap. 3.

The idea of the bilateral filter is to smooth the data without degrading discontinuities or edges in the scene. The filter is adapted to the special properties of ToF data, which greatly improves the performance. However, there are some scene configurations for which the filter does not perform satisfactorily at discontinuities. This puts the detection of depth edges into the focus of interest for chap. 4.

So far the number of publications on edge detection methods for ToF cameras is very limited. Again the special properties of the ToF data limits the usability of standard edge detection algorithms. Two very popular methods and their suitability for ToF data are investigated. They are extended and combined into a method that makes use of the full spectrum of data captured with ToF cameras, as well as the geometric arrangement of camera and light sources. The results are valuable input for dedicated adaptive denoising approaches.

While the smoothing algorithms are focused on removing the statistical depth deviation by averaging over a local neighborhood, these methods can never account for systematic depth deviations that affect a whole area.

In chap. 5 the raw data, which is used to calculate the depth values, is investigated in detail to find the properties responsible for the systematic errors. The results and interpretations are to the best of the authors knowledge the first explanations for a temperature dependent deviation of the depth values, a deviation due to the integration time and a pixel specific depth deviation.

The new knowledge is incorporated into a camera calibration in chap. 6, that can be completed in a matter of seconds without any scene or target requirements, in contrast to complex and time consuming previous approaches.

The calibration however fails to fix the intensity related distance error, another systematic error of unknown origin. The source of this error is identified in chap. 7 as in-camera light scattering. The calibration from the previous chapter enables the introduction and parameterization of a new model for in-camera light scattering

that is a lot simpler than previous propositions. It enables a significant reduction of the scattering effects and with it the intensity related distance error.

1.4 Camera

Unfortunately the raw data, which is crucial for this thesis can not be accessed with most cameras. If not otherwise specified, the datasets used in the thesis are all acquired with the PMD CamCube 3 (cf. fig. 1.2a), because the API offers access to the raw data.

The CamCube 3 has a square sensor with 200×200 pixels. Each of these pixels has two taps, of which only one is recording at a time. The active tap is alternated at 20 MHz. For each complete depth dataset or frame, the camera records four consecutive subframes, resulting in 8 raw frames.

A 12.8 mm f/1.1 lens gives it about 40° field of view in horizontal and vertical direction.

The camera illuminates the scene with two near infrared (870 nm) LED clusters, mounted left and right of the camera lens. They are modulated at the same frequency at which the tap sensitivity is switched, but with a sine signal instead of a rectangular one.

1.5 Validation

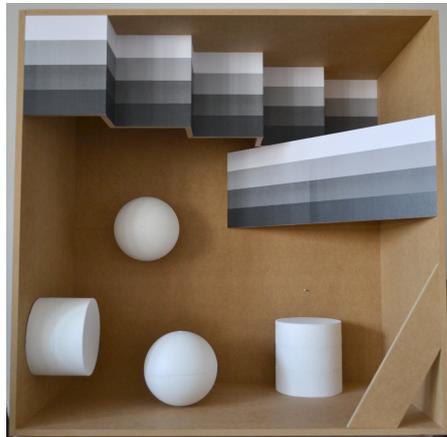


Figure 1.4: 3D target box with interchangeable 3D structures.

Validating the depth data recorded with a ToF camera is a difficult matter. It requires accurate reference data of the scene to compare to. In [37] this is investigated in detail. To enable a simpler method of validation, the relative accuracy of depth data can be judged by the consistency in a scene, if the local scene properties are known. In most publications more or less random setups are used to demonstrate success of proposed methods without an understanding of the true scene properties. These considerations in the course of this thesis led to the construction of a 3D target (cf. fig. 1.4) with simple and well known geometric

properties that is used repeatedly as a test setup for the performed measurements. The box has a width and height of 1 m and is 50 cm deep. It is built from medium-density fiberboard because of the low specularly. The ramp offers a continuous and the stairs a discrete depth progression. Both are coated with paper of varying reflectivity because of the known intensity dependency of ToF measurements. A number of simple polystyrene objects can be fixed with screws at a couple of different locations in the box as additional 3D structures.

2

Theory

The experiment by Fizeau mentioned in the introduction (chap. 1) very effectively shows the principle of pulse based Time-of-Flight cameras. For this thesis mainly a camera with a sinusoidally modulated light source (PMD CamCube 3) was used. This variation makes the distance calculations a little more complicated but also more robust, at least in the given implementation. The current chapter will give the theoretical foundation of the distance calculation with these cameras, sometimes also called continuous wave ToF cameras.

2.1 Time-of-Flight Principle

Time-of-Flight cameras obtain their name from the process they measure distance with. In principle the camera measures the time a light signal travels from the source (usually close to the camera) to a target or scene and back to the imaging sensor. Just emitting a short flash of light and measuring the time until the reflected signal reaches the sensor would be the straight forward approach but is technically not feasible. The required sampling rate to reach a reasonable distance resolution would be exceptionally high (3 GHz for 10 cm), as well as the required sensor sensitivity or illuminance respectively.

A feasible solution with current imaging sensor technology and non-hazardous illumination requires exposure times in the order of milliseconds. The actual temporal resolution for the distance measurement is achieved by modulating the light signal with frequencies in the radio spectrum (some MHz) and measuring the delay or phase shift of the reflected signal with respect to the emitted light.

To solve the problem of sampling high frequencies, the sensor sensitivity is modulated with a reference signal at the same frequency as the light source, just like the cogwheel in the experiment by Fizeau (cf. fig. 1.1). This is equivalent to an internal convolution (cf. sec. 2.2) of the reflected and the reference signal. The modulation can have various shapes. Sinusoidal modulation or rectangular (pulse

shaped) signals are possible, as well as arbitrary, non-periodic modulations [4].

By modulating the sensor with the same frequency as the light signal, an integration over many periods of the signal can still be regarded as one single measurement or sample. Assuming the target does not change until the next subframe is recorded, this can be the second sampling point.

The CamCube 3 records 4 subframes with an internal phase shift increased by 90° (cf. fig. 2.1) between each subframe. From the relative brightness changes in these pictures, the time of flight can be calculated on a single pixel level. The second tap can be used to reduce the statistical error or, with a special calibration as in [49] to calculate depth from only two subframes.

2.2 Internal Convolution

This section explains the internal convolution of a ToF camera with a sinusoidally modulated light source in detail. It can also be found for example in [49] or [48]. The deduction is performed here again for the sake of completeness.

The sensor modulation that switches the taps on and off has the same influence on the recorded intensity as a Heaviside step function multiplied with the sine of the light source (cf. eq. 2.1). The reflected light signal $\sin(\omega t)$ is delayed by an angle ϕ , proportional to the traveled distance.

$$\begin{aligned} I &= \int_0^{t_{int}} (a \sin(\omega t - \phi)) \cdot H(\sin(\omega t)) dt = \int_0^{\frac{T}{2}} a \sin(\omega t - \phi) dt = \left[\frac{-a}{\omega} \cos(\omega t - \phi) \right]_0^{\frac{T}{2}} \\ &= \frac{T}{2\pi} a (\cos(-\phi) - \cos(\pi - \phi)) = \frac{T}{\pi} a \cos(\phi). \end{aligned} \quad (2.1)$$

The light signal can of course not be negative, so the light source needs a constant offset i_o . To reach a radiance that is detectable even at the minimum of the sine wave, the added offset has to be larger than the amplitude a of the modulation. This changes our measured intensity after the integration only by a constant offset:

$$I = i_o \frac{T}{2} + \frac{T}{\pi} a \cos(\phi). \quad (2.2)$$

To calculate the three parameters (ϕ, i_o, a) at least three measurements are necessary. Four are technically and mathematically advantageous and allow a fit, with a compensating effect on errors in contrast to an exact solution with only three sampling points. The four measurements are done with different internal phase shifts $(\theta_n = n\frac{\pi}{2}$ with $n = 0..3$) of the reference signal (cf. fig. 2.1). Additionally, the time of a single period at 20 MHz is too short to get a reliable measurement. Therefore, the signal is integrated over several thousand (m) periods:

$$I(\theta_n) = \frac{mT}{2} i_o + \frac{mT}{\pi} a \cos(\phi + \theta_n) = I + A \cos(\phi + \theta_n). \quad (2.3)$$

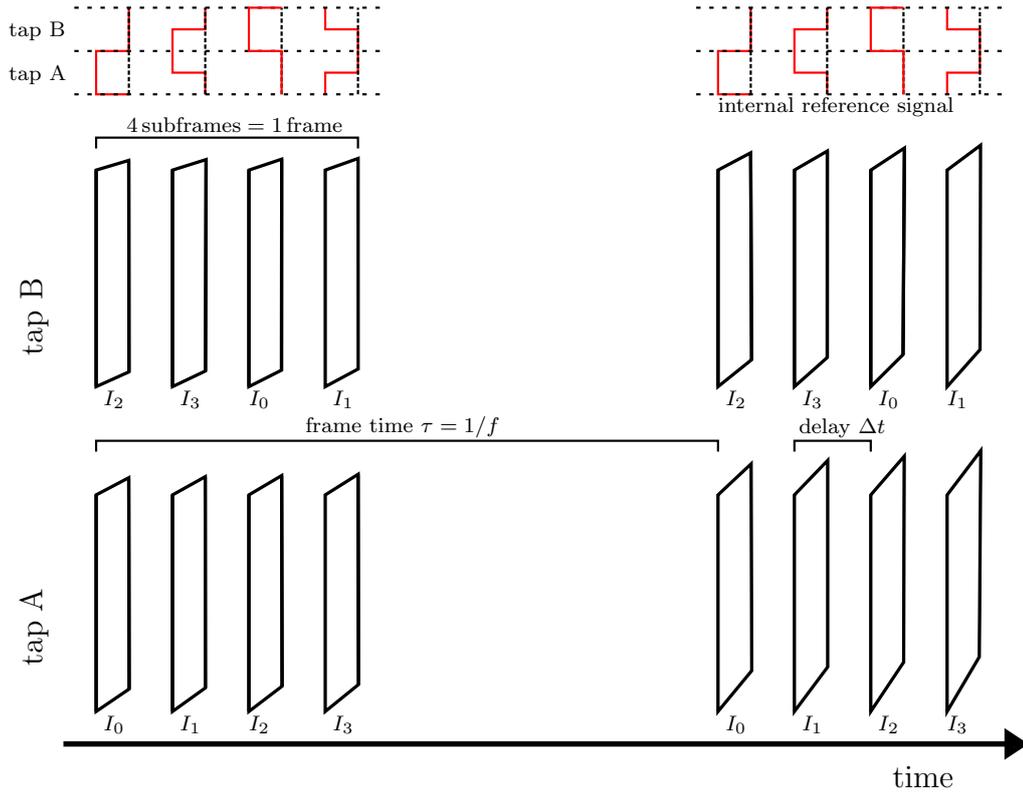


Figure 2.1: Schematic of the data recording process. In each frame the camera records four consecutive subframes at different phase shifts of the reference signal. Every pixel is divided into two taps and thus records two of those sequences simultaneously with opposing phase shifts.

The actual length of the integration for each measurement is later specified as the integration time

$$t_{int} = m \cdot T. \quad (2.4)$$

To simplify the notation $I(\theta_n)$ will be written as I_n from now on. The three parameters intensity I , amplitude A and phase ϕ can now be found by minimizing the least squares formulation of the problem:

$$\min_{I,A,\phi} \sum_{n=0}^3 ((I + A \cos(\phi + \theta_n)) - I_n)^2. \quad (2.5)$$

The results for the three parameters are found to be:

$$I = \frac{I_0 + I_1 + I_2 + I_3}{4} \quad (2.6)$$

$$A = \frac{1}{2} \sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2} \quad (2.7)$$

$$\phi = \arctan2((I_3 - I_1), (I_0 - I_2)). \quad (2.8)$$

The results or notations in previous work (cf. [43, 49, 13, 49]) is slightly varying, which is why the formulas were deduced again and can be found in sec. A.1.

The phase shift ϕ can of course be transformed linearly into the distance d as:

$$d = \frac{1}{2}t \cdot c = \frac{1}{2\omega} \cdot c \quad (2.9)$$

where t is the actual time of flight and c the speed of light. The factor $\frac{1}{2}$ is due to the light traveling from the camera to the target and back. It should also be noted here, that the distance or phase is of course not the same as the z-component in Cartesian coordinates but rather the radial distance to the camera.

Because of the periodicity of the illumination the algorithm can not distinguish between angle ϕ and $\phi + 2\pi$, which is why

$$d_a = \frac{1}{2} \frac{2\pi}{\omega} \cdot c = \frac{1}{2} \frac{c}{f} \quad (2.10)$$

is called the non-ambiguity range of a ToF camera. For the CamCube 3 with $f = 20$ Hz it is at about 7.5 m.

2.3 Poisson Noise

An ideal imaging sensor will suffer only from poisson noise, due to the finite probability of the photon transfer process. The poisson distribution has a variance σ^2 equal to the expected value (cf. [19]).

As written in [43] and [13], the variance of the vector $\vec{f}(I_n) = (I, A, \phi)^T$ can be calculated as

$$Var(\vec{f}(\vec{I})) = \begin{pmatrix} \sum_n \frac{\partial I}{\partial I_n} \sigma_{I_n}^2 \frac{\partial I}{\partial I_n} & & \dots \\ & \sum_n \frac{\partial A}{\partial I_n} \sigma_{I_n}^2 \frac{\partial A}{\partial I_n} & \\ \dots & & \sum_n \frac{\partial \phi}{\partial I_n} \sigma_{I_n}^2 \frac{\partial \phi}{\partial I_n} \end{pmatrix}. \quad (2.11)$$

The diagonal elements account for the variances of I, A and ϕ . The expected value and variance is now the respective intensity value I_n :

$$Var(\vec{f}(\vec{I})) = \begin{pmatrix} \sum_n \frac{\partial I}{\partial I_n} I_n \frac{\partial I}{\partial I_n} & & \dots \\ & \sum_n \frac{\partial A}{\partial I_n} I_n \frac{\partial A}{\partial I_n} & \\ \dots & & \sum_n \frac{\partial \phi}{\partial I_n} I_n \frac{\partial \phi}{\partial I_n} \end{pmatrix}. \quad (2.12)$$

In the previous deductions (cf. [43, 13]) the same variance σ^2 is used for all raw frames. Because of that and some typos in the deductions, it is carried out again in sec. A.2.

Using eq. 2.6-2.8, it results in:

$$\sigma_I^2 = \frac{I}{4} \quad (2.13)$$

$$\sigma_A^2 = \frac{I}{2} \quad (2.14)$$

$$\sigma_\phi^2 = \frac{I}{2A^2}. \quad (2.15)$$

It turns out that the assumption of a constant variance gives the same results as using different variances for each subframe.

The non-linearity of the variance of ϕ with respect to the amplitude has to be emphasized here. It is responsible for the very high noise levels for areas in the scene with low reflectivity.

3

Bilateral Filtering

As we have seen in sec. 2.3, even a perfectly calibrated ToF camera suffers from noise of variable magnitude, depending on the intensity and amplitude of the recorded signal. For further processing of the data, a denoising method is required.

The simplest approach would be a Gaussian filter, taking the weighted local mean value of a neighborhood. But the Gaussian does not only smooth the noise. It is essentially a low pass filter and will also round off the actual depth discontinuities (cf. fig. 3.1 center), which are very important for the geometric representation of a scene, that is the heart of a depth image. A much better filter for piecewise smooth geometric structures would be a truncated Gaussian, cut off at discontinuities. A fairly good approximation was first introduced by Aurich and Weule in 1995 [2] as non-linear Gaussian filter, that later became known as the bilateral filter (cf. [54]).

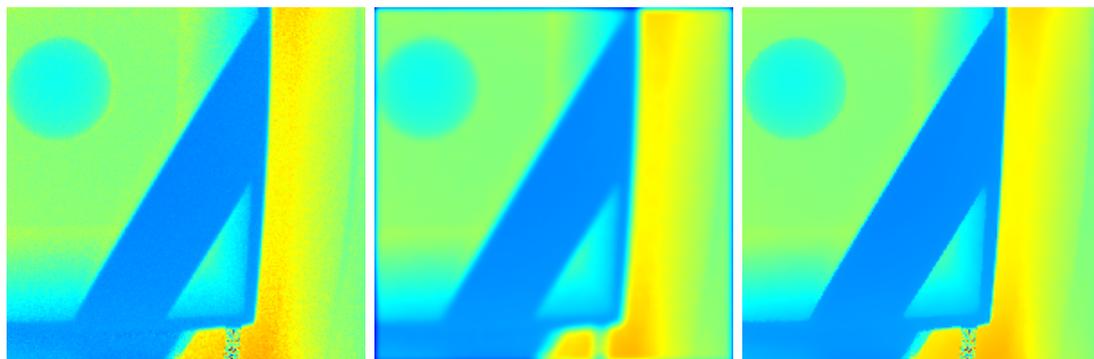


Figure 3.1: Depth data (left) filtered with standard Gaussian (center) and bilateral filter (right) with the same spatial parameter. The bilateral filter reduces noise effectively in most of the frame while keeping the edges sharp, but the small area in the bottom right with strong noise is hardly smoothed at all.

The filter is non-iterative and local, making it ideal for a fast, parallel imple-

mentation. It is widely popular in image processing today and some new ToF cameras can already apply it to the depth data directly in the camera firmware.

In the following sections the bilateral filter will be explained in detail with a couple of extensions, making it better suited for ToF data.

3.1 Standard Bilateral Filter

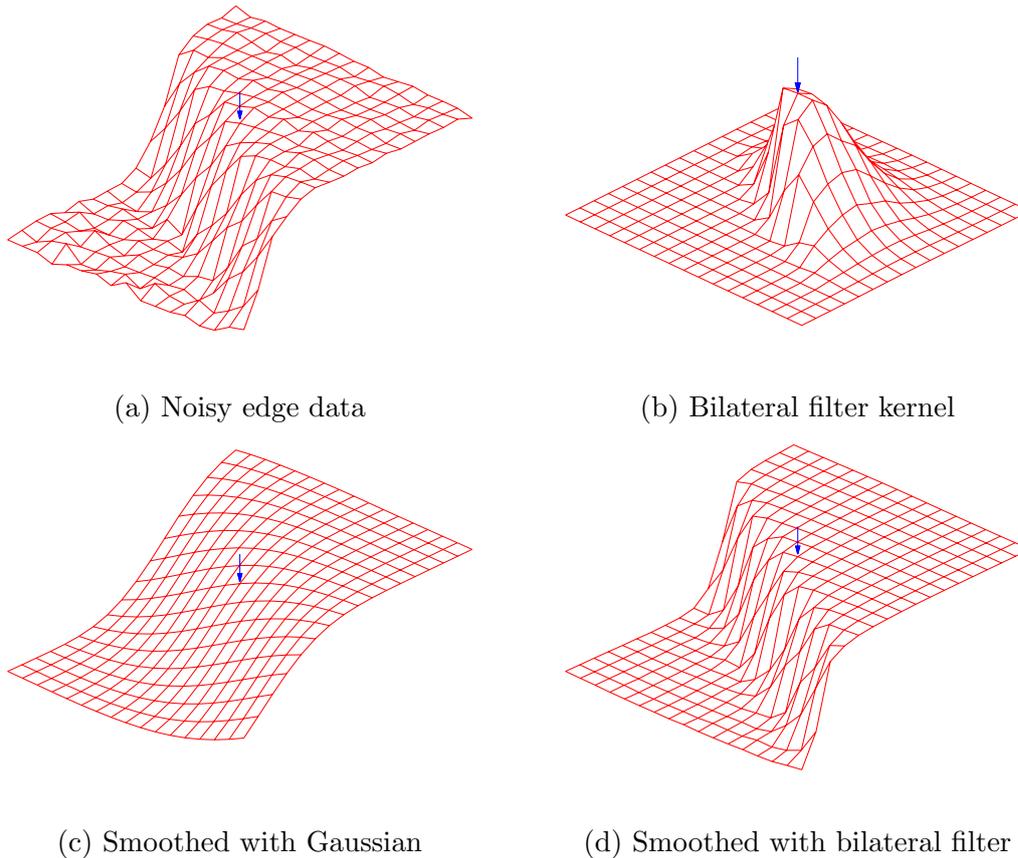


Figure 3.2: Example for the shape of a bilateral filter kernel. The spatial part of the bilateral filter used for the smoothing in this figure has the same σ as the Gaussian. The bilateral filter can be a good approximation of a truncated Gaussian.

The bilateral filter (fig. 3.2) is a combination of two Gaussian filters:

$$\tilde{d}(\vec{x}) = \sum_{\vec{y} \in \Omega(\vec{x})} G_x(\vec{x}, \vec{y}) G_z(\vec{x}, \vec{y}) d(\vec{y}). \quad (3.1)$$

One is the standard spatial Gaussian G_x , averaging over the surrounding pixels $\Omega(\vec{x})$, weighted with the distance:

$$G_x(\vec{x}, \vec{y}) = \left(\sum_{\vec{y} \in \Omega(\vec{x})} e^{-\frac{\|\vec{x}-\vec{y}\|}{2\sigma_x^2}} \right)^{-1} e^{-\frac{\|\vec{x}-\vec{y}\|}{2\sigma_x^2}} \quad (3.2)$$

and the second Gaussian, weighted with the difference in the data domain, the depth or phase data in this case:

$$G_z(\vec{x}, \vec{y}) = \left(\sum_{\vec{y} \in \Omega(\vec{x})} e^{-\frac{|d(\vec{x})-d(\vec{y})|}{2\sigma_z^2}} \right)^{-1} e^{-\frac{|d(\vec{x})-d(\vec{y})|}{2\sigma_z^2}}. \quad (3.3)$$

This increases the weight of pixels similar to the center pixel in the averaging process. An exemplary kernel and smoothing process at a step edge can be seen in fig. 3.2.

Two small measures increase the effectivity of the filter. First, as proposed in [41], the filtering process can be repeated several times - thrice for example - while σ_z is decreased in every step. This smooths the noise more effectively and reduces the smoothing of edge discontinuities.

The second is proposed in [25]. If the currently filtered pixel is a strong outlier, it will hardly be smoothed at all, because every other surrounding pixel would be weighted only weakly. Setting the weight of the central pixel ($\vec{y} = \vec{x}$) to 0 greatly increases the smoothing of such pixels without compromising the smoothing process for other cases.

While the spatial width of the filter is not very critical, the proper width of the second Gaussian σ_z is crucial for a good result. It has to be above the noise level to effectively smooth the data. At the same time it should be below the smallest step expected in the data d , to prevent the filter from smoothing actual image structures. This can be a very tough decision for data with a low signal to noise ratio or strong variations in the noise magnitude (cf. fig. 3.1). It also makes the bilateral filter unsuitable to smooth images or areas of images, where the noise has a similar magnitude as the depth discontinuities.

3.2 Joint Bilateral Filter

As written above, σ_z has to be set above the noise level for effective denoising. For ordinary intensity images the noise level is usually rather uniform across the frame. For ToF depth data this is not the case. As eq. 2.15 shows, the standard deviation of the depth measurement depends on the intensity and especially the amplitude of the signal. This makes it very difficult, if not impossible to set a good σ_z for the filtering process. In areas with low reflectivity the depth data can basically vary through the whole ambiguity range of the camera. A σ_z suitable for this area would smooth out any other depth structures as well, while a σ_z for fine depth structures leaves the strong noise in dark areas of the image almost unchanged.

One possible solution is a method proposed in [26] to upsample RGB images, where a second, higher resolution image is used as weighting for the non-spatial domain. This method can also be transferred to ToF data (cf. [56]), if a second image of the same scene with better signal to noise ratio is available from an ordinary RGB-camera for example.

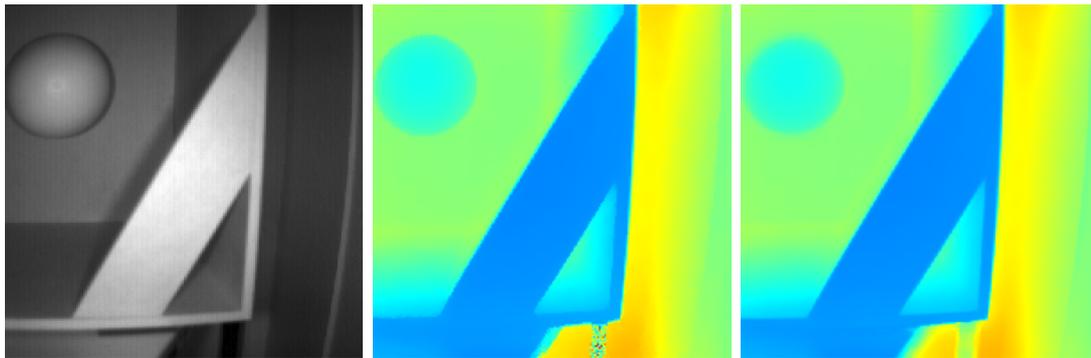


Figure 3.3: Exemplary results of a joint bilateral filter. Depth data (right) filtered with weighting from the intensity data (left). Compared to the standard bilateral filter (center), the strong noise in the bottom right is smoothed quite well, but the edges of the sphere in the top left are blurred.

In fig. 3.3 the intensity channel is used as the second weighting domain:

$$\tilde{d}(\vec{x}) = \sum_{\vec{y} \in \Omega(\vec{x})} G_x(\vec{x}, \vec{y}) G_I(\vec{x}, \vec{y}) d(\vec{y}). \quad (3.4)$$

The better signal to noise ratio helps to smooth the noisy area in the bottom right, where the standard bilateral filter failed. But at the same time the borders of the sphere in the top left are blurred because of the small intensity difference.

Another issue of the joint bilateral filter is shown in fig. 3.4, where lots of reflectivity changes appear without a change in depth. These texture edges are copied into the smoothed depth data.

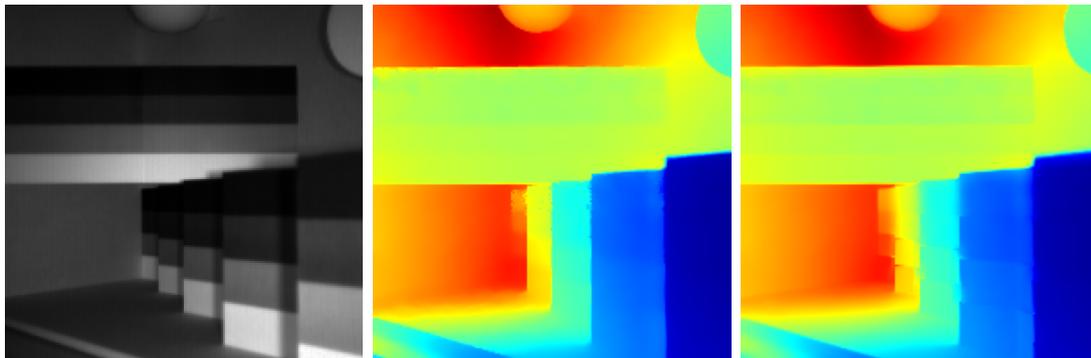


Figure 3.4: Another issue of the joint bilateral filter (right) is shown here. The texture edges in the intensity data (left) are copied into the depth data. The standard bilateral filter (center) does not suffer from these artifacts.

3.3 Adaptive Bilateral Filter

In [6] the texture copying issues are attended with a blending function $\alpha(\Delta_\Omega)$ that shifts the weighting of the bilateral filter between the additional image and the

depth data:

$$\tilde{d}(\vec{x}) = \sum_{\vec{y} \in \Omega(\vec{x})} G_x(\vec{x}, \vec{y}) (\alpha(\Delta_\Omega) G_z(\vec{x}, \vec{y}) + (1 - \alpha(\Delta_\Omega)) G_I(\vec{x}, \vec{y})) d(\vec{y}), \quad (3.5)$$

where α depends on the range of the depth values in the local neighborhood Ω after a Gaussian smoothing:

$$\alpha(\Delta_\Omega) = (1 + e^{-\epsilon(\Delta_\Omega - \tau)})^{-1}. \quad (3.6)$$

In this algorithm from [6] ϵ controls the width of the transition region and τ the center of it. The idea is to rely on the depth data for areas with homogeneous depth and switch to the intensity data in areas with depth edges. Fig. 3.5 shows, that texture copying is indeed successfully avoided. But it also shows how this model can easily fail. At depth edges that are not visible in the intensity data the smoothing kernel is not properly cut off and the edges are blurred. The performance is actually equally good or even better, if α is fixed to 0.5 for the whole image, as previously published in [32]:

$$G_{zI}(\vec{x}, \vec{y}) = \left(\sum_{\vec{y} \in \Omega(\vec{x})} e^{-\frac{|d(\vec{x}) - d(\vec{y})|}{2\sigma_z^2}} + \sum_{\vec{y} \in \Omega(\vec{x})} e^{-\frac{|I(\vec{x}) - I(\vec{y})|}{2\sigma_I^2}} \right)^{-1} \left(e^{-\frac{|d(\vec{x}) - d(\vec{y})|}{2\sigma_z^2}} + e^{-\frac{|I(\vec{x}) - I(\vec{y})|}{2\sigma_I^2}} \right). \quad (3.7)$$

This way the depth difference at the depth edges invisible in the intensity lead to a better preservation of the structure.

3.4 Discussion

The bilateral filter is a great improvement over isotropic Gaussian smoothing of ToF depth data. Because of the low resolution of ToF cameras any smearing has to be avoided. While the standard approach can be improved by using additional channels available from the ToF camera, there is no real solution that will work perfectly in any scenario. An approach sensitive only to depth values will struggle with the varying noise levels, while a filter sensitive to intensity data can cause false details in the depth data or smooth obvious depth edges. Even a combination of both channels is difficult, because the discontinuity can disappear in either of them.

The struggle of the algorithm at depth edges triggers the idea of a dedicated edge detection algorithm to find these critical regions beforehand. This is exactly the goal of the next chapter.

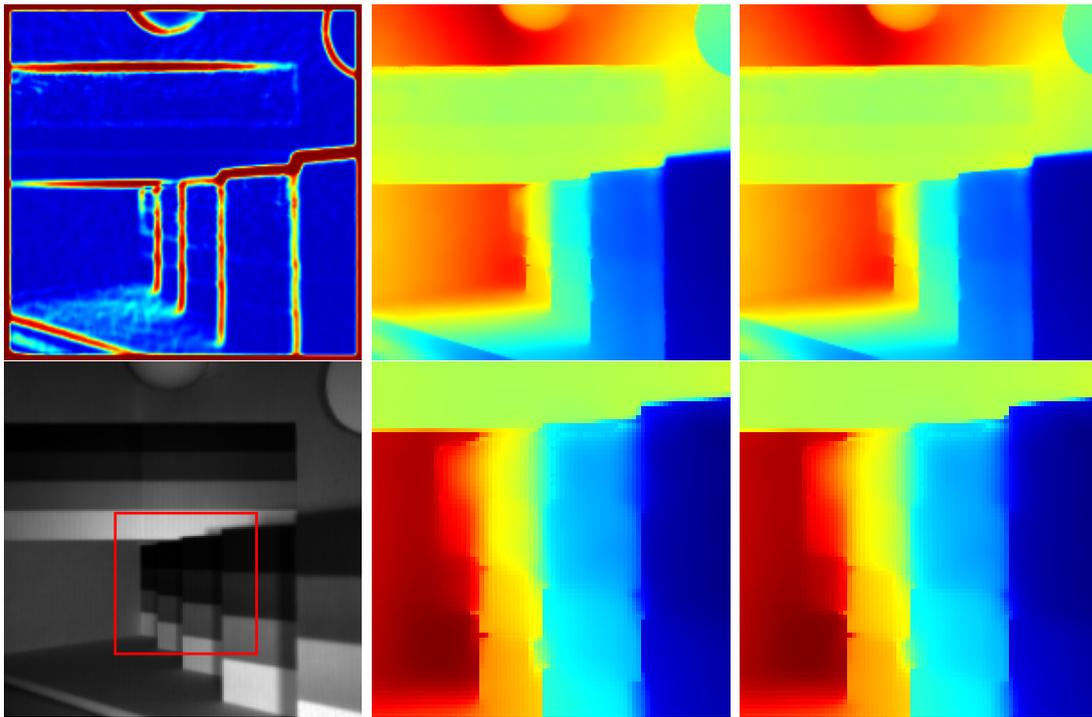


Figure 3.5: The transition function $\alpha(\Delta_\Omega)$ (top left) shows a good response to the depth edges. Still the step edges in the dark stripes are blurred (center) due to the weak intensity difference. The same algorithm with a fixed $\alpha = 0.5$ shows an equal or even better performance (right). The two depth images in the bottom row are crops from above, as marked in the intensity data (bottom left). At the vertical step edges the result with a fixed α retains slightly more sharpness.

4

Edge Detection

Edge detection has been an important field in image processing from the start. The edges already hold a large part of the information of a scene, as they mark the locations where properties like color and brightness change. Most scenes or objects can be recognized by humans just by looking at the edges.

In case of depth data, the edges are even more important, because there are no textures in depth data and an edge always represents a spatial discontinuity of an object.

Standard edge detection works fine for “good” scenes, where the depth steps at edges are large and well beyond the noise level. But this is not the case for every scene and the unique properties of the depth data, especially the varying noise (cf. sec. 2.3) that can exceed the signal in weakly reflecting or illuminated areas, pose a strong challenge for standard edge detection algorithms.

To the best of the authors knowledge, there have only been two papers dedicated to general edge detection in ToF data so far. First a paper by Lejeune et al. [31] and one paper written in the course of this thesis [47]. Ye and Hegde [57] suggested an algorithm for ToF data that was limited to straight edges and other related work is edge detection in data from range scanners (cf. [7, 20, 3, 42]), which is different from ToF data since it is not densely sampled and not as close to 2D images.

The algorithm proposed in [31] is a variation of Canny’s algorithm (cf. sec. 4.2.1), but it is limited to the depth data because it is also used on depth data from the KinectTM. The work presented here is making use of the multiple channels ToF data offers with the goal of increasing the sensitivity to edges at shallow depth steps without diluting the results by introducing false positives into the edge map.

4.1 Structure Tensor

One way to find edges in an image is the structure tensor approach, as previously described in [17, 12]. In the course of this thesis an algorithm was implemented,

where the structure tensor is used to detect edges in ToF data. This edge information can be used for adaptive denoising of the depth data (cf. [32]).

4.1.1 Theory

The theory of the structure tensor from [12] is briefly repeated here. A more complete and comprehensive explanation can be found in [19]. The structure tensor J is the Cartesian product of the gradients, locally smoothed:

$$J = G_\sigma * \begin{pmatrix} I_x \\ I_y \end{pmatrix} \begin{pmatrix} I_x \\ I_y \end{pmatrix}^T = G_\sigma * \begin{pmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{pmatrix}, \quad (4.1)$$

with G_σ representing a Gaussian filter with standard deviation σ and I_x denoting the gradients $\partial I / \partial x$ in x direction, I_y in y direction respectively. The structure tensor is symmetric and can be diagonalized into

$$J = \begin{pmatrix} u_1 & v_1 \\ u_2 & v_2 \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \begin{pmatrix} u_1 & u_2 \\ v_1 & v_2 \end{pmatrix} \quad (4.2)$$

where \vec{u} and \vec{v} are the normalized Eigenvectors to the Eigenvalues λ_1 and λ_2 of J respectively.

The structure tensor gives both, an orientation of the local values of an image, where the Eigenvector of the larger Eigenvalue points in the normal direction of the orientation and a degree of the orientation of the local neighborhood as the relation of the Eigenvalues λ_2 / λ_1 . A large value for λ_2 / λ_1 is a good indication for a high and elongated gradient, typical for an edge structure in the image data.

In [27] it is suggested, that for correct processing in the calculation of the structure tensor, the gradients have to be sampled with twice the resolution of the image. The reason is the doubling of the frequency when the point wise product of two signals is formed (cf. fig. 4.1), as in eq. 4.1 the products of the gradients. The upsampling of the gradients is done by sampling the gradient at every half-pixel position by applying a gradient filter of equal and unequal width and length. For example:

$$\nabla_1 = \begin{pmatrix} 1 & -1 \end{pmatrix} \quad (4.3)$$

$$\nabla_2 = \begin{pmatrix} 1 & 0 & -1 \end{pmatrix} \quad (4.4)$$

$$\nabla_3 = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \quad (4.5)$$

$$\nabla_4 = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix}. \quad (4.6)$$

The image itself will of course not contain any component with a frequency above the Nyquist limit and neither will the gradients. But the product of two gradients might.

Another extension proposed in [27] is an adaptive region for the smoothing of the structure tensor components. The standard procedure is a Gaussian filter. But this can cause local smearing between two adjacent edge structures, which

is why an hour-glass shaped filter is suggested (cf. fig. 4.2). The hour-glass is supposed to be aligned along the orientation of the local structure. This means the structure tensor has to be calculated twice: once, to find the orientation with a standard isotropic filter and a second time with the anisotropic filter, aligned with the orientation found before.

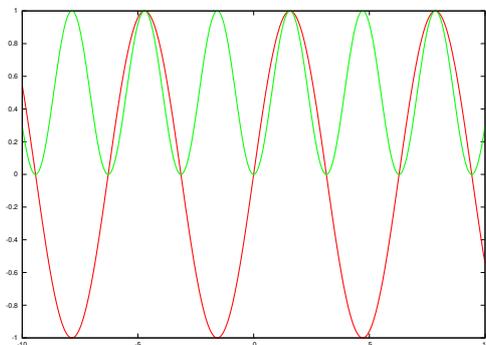


Figure 4.1: $\sin x$ and $\sin^2 x$ to illustrate the frequency doubling when a signal is multiplied pointwise with itself.

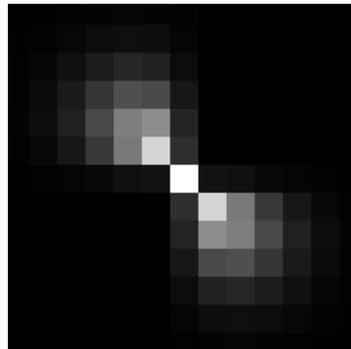


Figure 4.2: Grey-value representation of an hour-glass shaped filter kernel as suggested in [27].

4.1.2 Structure Tensor and Time-of-Flight Data

The approach is now applied to the depth and intensity data from the CamCube 3 (cf. fig. 4.3). The different properties of the depth and intensity data result in very different edge confidences for the two datasets. Due to the $1/r^2$ decrease of illumination power, the depth edges in the left part of the scene, much closer to the camera, are very well detected in the intensity image (cf. fig. 4.3 left), although the depth difference is not very large. The low reflectivity of the chair to the right and the increasing distance of the wall behind it almost cancel each other out in the intensity data. This causes very weak edges in the intensity data (cf. fig. 4.3 left). But the edges are very well visible and detectable in the depth data (cf. fig. 4.3 right).

With ToF it is possible to combine the edge information from both images without a complicated geometric calibration in a dual camera setup. The simplest way to do this is to add the resulting edge confidence maps. Despite the enhancements in sec. 4.1.1, the dark areas and the resulting strong noise still gives a strong response in the edge image and simply adding them would carry these false responses over into the final image (cf. fig. 4.4 left, bottom right corner). Adding the structure tensors one step earlier on the other hand gives very good results. It still enables control over the parameters for each dataset individually, like the size of the smoothing parameter for the structure tensor components. Structures or edges, that are not present in the second image are greatly diminished this way (cf. fig. 4.4 right).

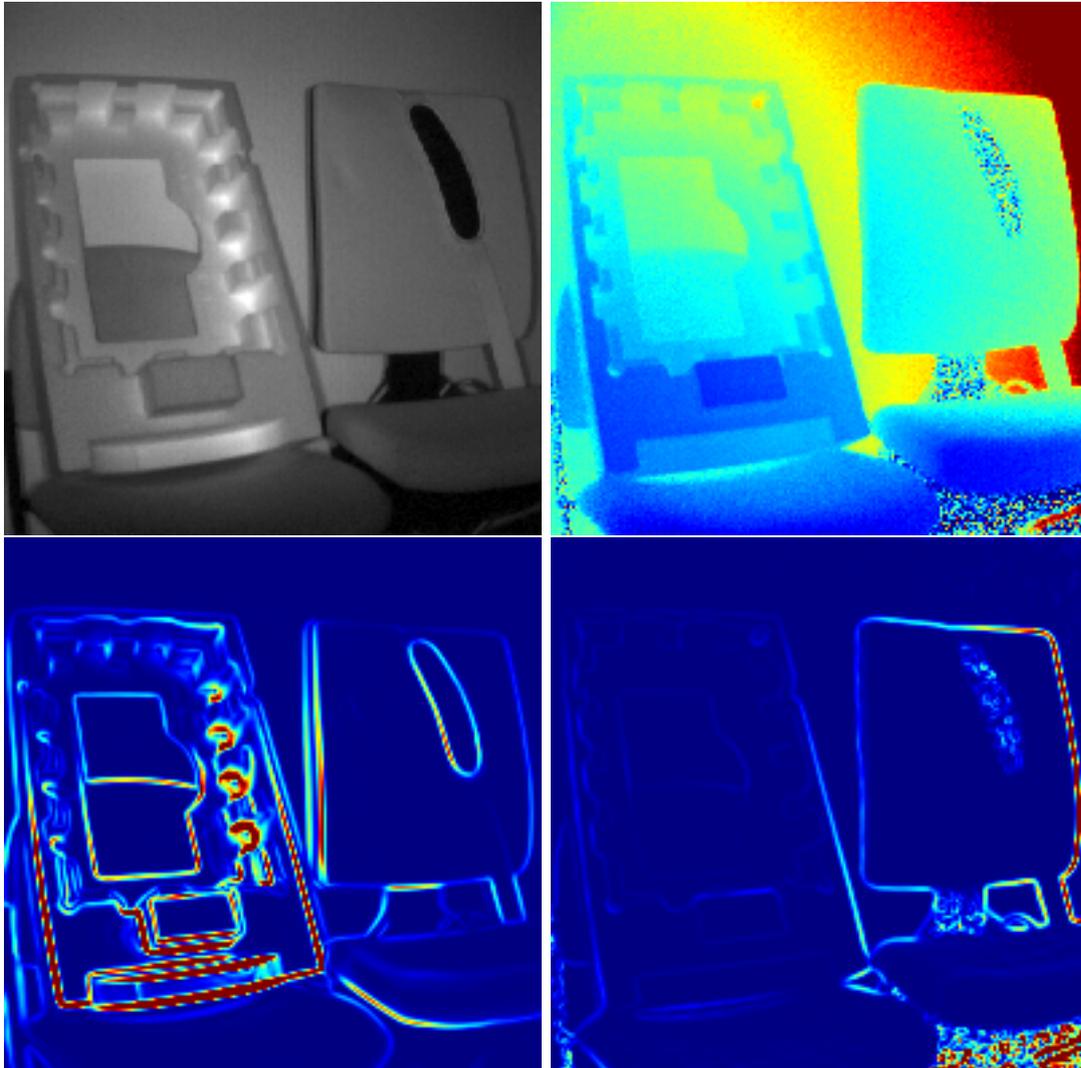


Figure 4.3: Intensity and depth image of a scene, with the according eigenvalues of the structure tensor. The approach works much better on the intensity data and struggles with the noisy areas in the depth image. The values displayed in the bottom row are the relation off the eigenvalues, which can be taken as a measure for the likelihood of an edge at a certain pixel.

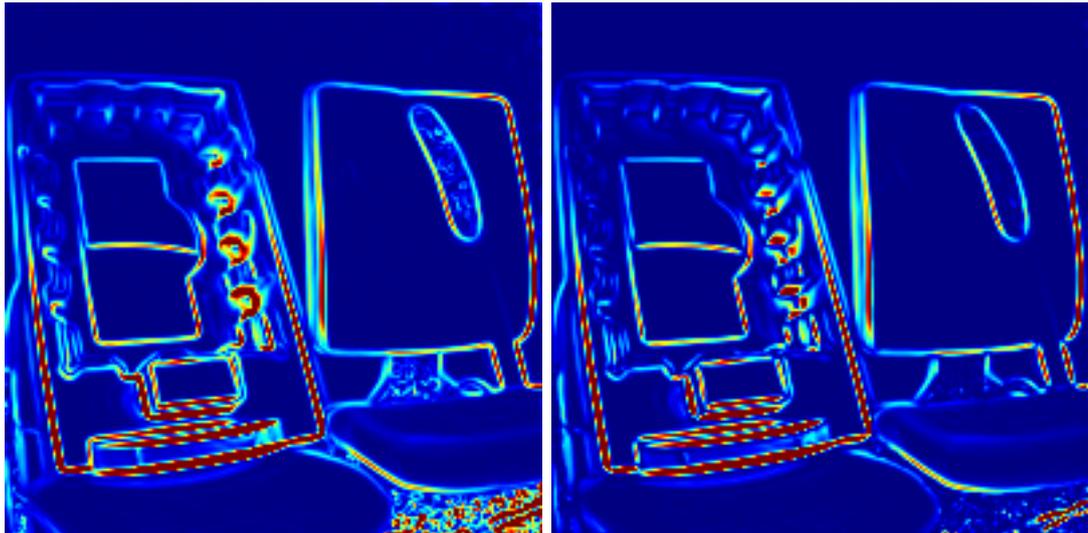


Figure 4.4: Left: sum of the two edge images from fig. 4.3. The noisy parts of the depth image are carried over without attenuation. Right: edges from the two images combined at the structure tensor level. The strong features from both datasets are carried over, while the noise of the depth data is suppressed quite well.

4.1.3 Application

The results off the structure tensor approach can be used as a guidance for adaptive denoising like in [32]. It gives a local likelihood of an edge and its orientation, used as an input for an adaptive total variation approach. The goal is to smooth the data along an edge, but not across it. However, because of the local smoothing, the likelihood of a present edge is spread over several pixels in cross-edge direction and a continuous classification of edge likelihood is prone to degrade the sharpness of weaker edges. The logical conclusion is an algorithm that comes to a final decision whether a pixel is an edge or not. Such an approach will be discussed in the following sections.

4.2 Structure Tensor and Canny Combined

In [27] a combination of the structure tensor approach and the algorithm by Canny [5] was suggested for edge detection. The idea is to use the eigenvalues of the structure tensor as an input to the Canny algorithm, instead of the gradient maximums. This makes the algorithm more robust to slight variations of the gradient and increases the sensitivity for elongated structures.

4.2.1 Canny's Algorithm

Canny's algorithm [5] is still a very popular approach for edge detection. It is divided into several steps. The first step is a calculation of the gradients in the image. In a second step local maximums of the gradients are detected as possible

edge pixel candidates and in a third step called the hysteresis, connected lines of these maximums are identified as edges. The critical part is the connection of the local maximums. To avoid detecting every small maximum in the gradients as an edge, two thresholds are set (cf. fig. 4.5). One is the lower threshold t_l , that is supposed to include most maximums that could be an edge. The second threshold t_h is higher, to include only maximums that definitely are edges. A connected segment of maximums higher than t_l is only interpreted as an edge, if at least one of the maximums is also higher than the second threshold t_h . This makes the selection of the thresholds the most crucial part of the algorithm. The result is a one pixel wide line, denoting edges in the image.

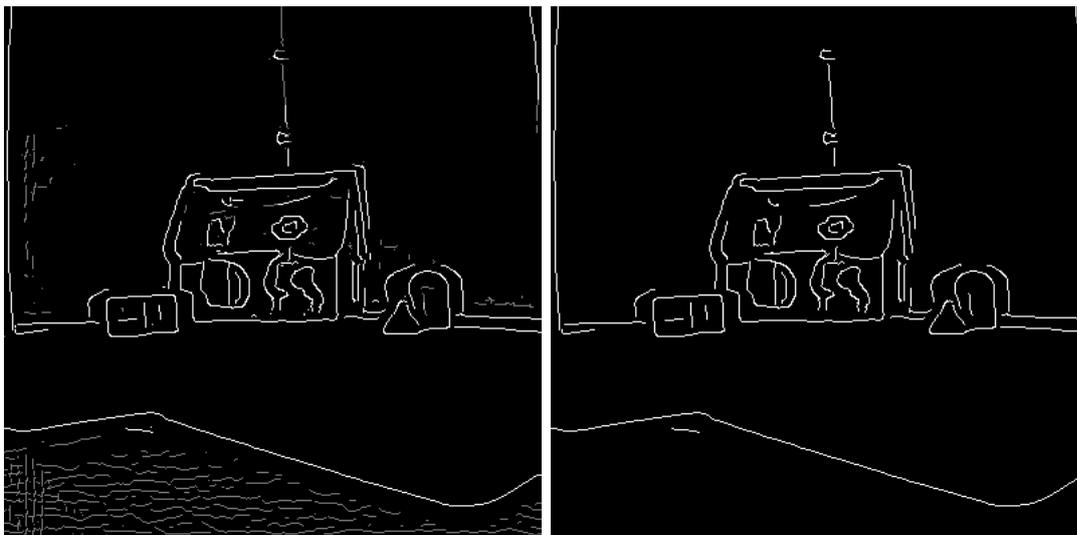


Figure 4.5: Left: Pixels marked as edges. Grey pixels are above the lower threshold t_l , white pixels above the higher threshold t_h . Edges without at least one white pixel are dismissed. Right: Edges after the hysteresis step.

4.2.2 Adapting Canny's Algorithm for ToF Data

The strong noise in Time-of-Flight depth data makes it very difficult to set appropriate thresholds t_l and t_h . Using the structure tensor edge indication instead of the gradients directly is already an advantage. Further steps to make the algorithm more robust to noise are introduced here in the hysteresis step. Sudden changes in the edge orientation are prohibited. A measure is the scalar product of the eigenvectors of the two consecutive pixels. A threshold $t_s \leq 1$ is assigned to break edges in case of a sudden change of orientation. A positive side effect is the break up of edges that run around corners into two separately detected edges. This will be very useful later on. Another extension is to exclude edges with a length below a certain threshold n_m . Because of the strong noise in ToF depth data high gradients can also occur in areas of uniform depth, but usually only very short ones. This feature comes at a cost of course, because actual short edges in the scene are either removed or limit the parameter to a very low value. Still, depending on the scene it is a very useful additional selection parameter.

4.3 Shadow Removal

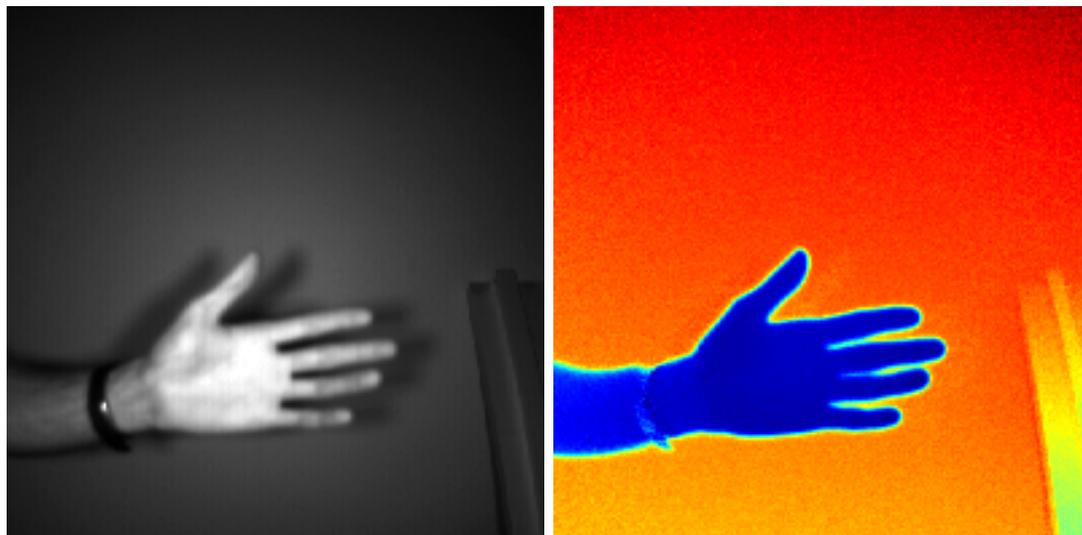


Figure 4.6: The light sources mounted to the left and right of the lens cast strong shadows on the background next to an object.

One advantage of ToF cameras are the multiple channels they provide. For instance, the intensity and depth data provided are both perfectly aligned because they are recorded from the same raw data. This offers the possibility to fuse the edges detected in the different channels without problems of alignment. The intensity data suffers a lot less from noise than the depth data and is therefore much more suited for edge detection. A problem of the intensity data however is, due to the light sources of the camera mounted left and right of the lens, vertical depth edges occlude one of the two light sources for areas next to them (cf. fig. 4.6). The result is a strong change of intensity at the edge of this shadow, although the reflectivity in the scene is possibly constant. Detecting these shadow edges offers two advantages:

1. the shadow edges can be removed from the intensity edges and
2. the edges causing the shadows have to be real depth edges.

4.3.1 Finding the Shadow Edges

To identify shadow edges as such in the intensity data, the shadow casting edge in the depth data must be found. Since the positions of the light sources with respect to the lens are known and the ToF camera can measure the distance, this connection can be formed by calculating the angle α' between the edge pixel and the light source (cf. fig. 4.7). This angle has to be identical for both, the shadow edge and the depth edge:

$$\tan \alpha'_1 = \frac{d_1}{a + x_1} = \frac{d_1}{a + x_2} = \tan \alpha'_2. \quad (4.7)$$

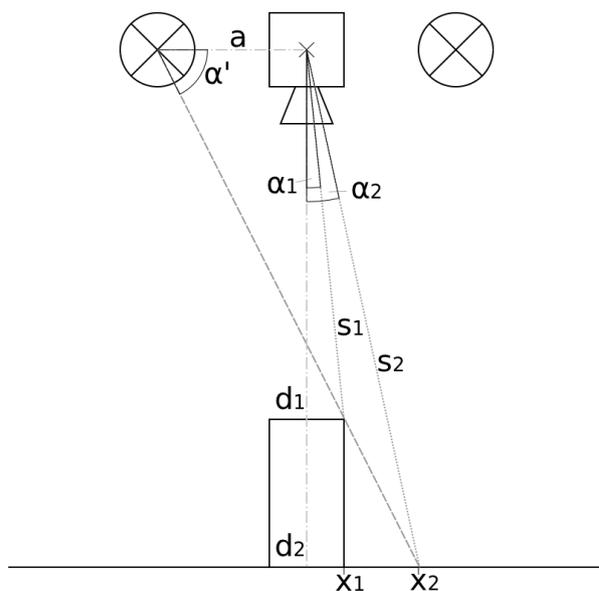


Figure 4.7: Schematic of the shadows cast by a depth edge due to the light sources mounted left and right of the camera. Shadow- and depth-edge share the same angle α' towards the light source.

The two angles α_1 and α_2 are the respective angles of the depth edge and the intensity edge here. To find the correspondences, this angle is assigned to all edge pixels with respect to both light sources. The pairs are then found by conducting a search along each row of the two edge maps. The criteria for a match are:

1. The edges have a vertical component above a certain threshold.
2. Matching angles according to eq. 4.7 with a certain tolerance for the width of the light source and the inaccuracy of the depth measurement.
3. For angles calculated towards the left light source the shadow edge has to be to the right of the depth edge and vice versa.
4. Once two corresponding pixels are found, all pixels that are part of both edges are checked for correspondences. Only if a certain percentage fulfill the former correspondence criteria, they are considered as a depth and shadow edge pair.

Fig. 4.8 shows an example for detected depth and shadow edges.

4.3.2 Smartly Combining Intensity and Depth Edges

As suggested before (cf. sec. 4.2.2), the depth edges are a difficult matter, because of the high noise level. To find all relevant depth edges the thresholds even for the adapted algorithm usually have to be set so low, that also many false edges are detected. The shadow removal part can help here though.

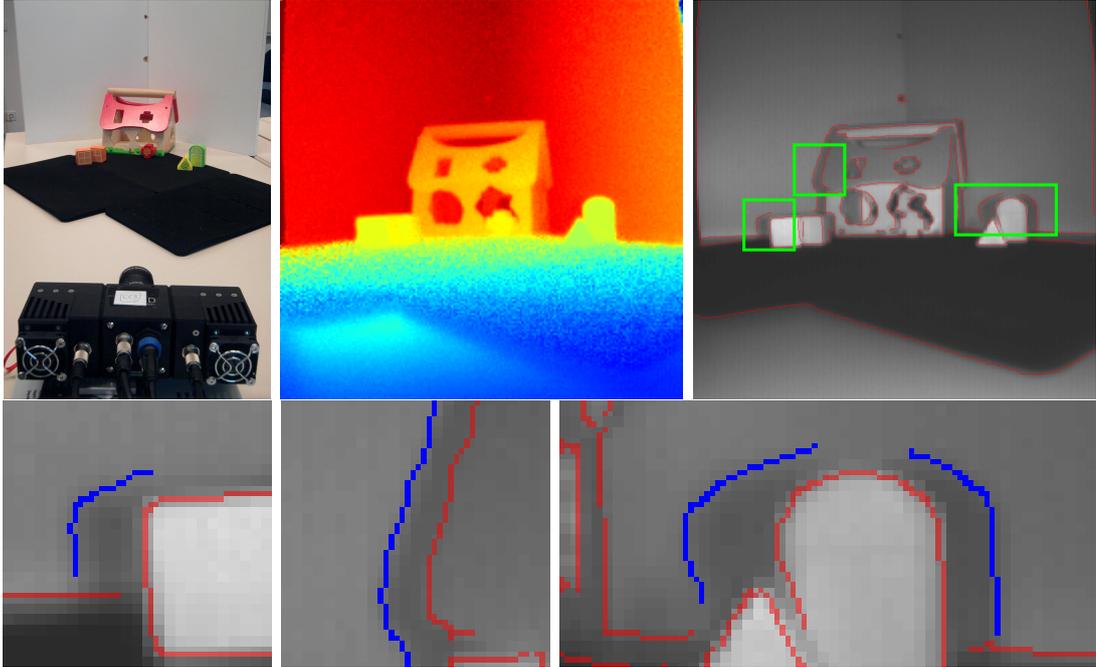


Figure 4.8: Example scene for detected shadow edges. Previously published in [47].

To find all possible shadow casting depth edges, an edge map is created from the depth data with very low thresholds and a low confidence for the detected edges (cf. fig. 4.9e). Because of the high number of falsely detected edges, they are not fused directly into the final result, but only the depth edges with a corresponding shadow in the intensity edges (cf. fig. 4.9f) are selected. The result of this operation is an edge map that consists of the intensity edges, but the shadow edges are exchanged with the corresponding depth edges (cf. fig. 4.9h).

A second set of depth edges detected with higher thresholds for a very low false positive rate (cf. fig. 4.9d) is combined with these edges for the final result (cf. fig. 4.9i).

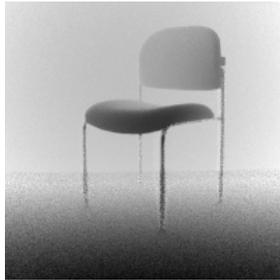
4.4 Texture Removal

Another issue with edges from intensity data are possible textures that result in intensity edges without a change in depth (cf. fig. 4.10), just like shadow edges. Because of the fusion of depth and intensity edge maps, they will also end up in the final result of the algorithm proposed above. But since the depth data is known, a straight forward examination of the depth values on either side of the edge can help to identify these texture edges.

To reduce the effects of noise and mixed pixels at edges, the area of the comparison can be extended to a small neighborhood on both sides of the edge E to



(a) RGB image of the scene



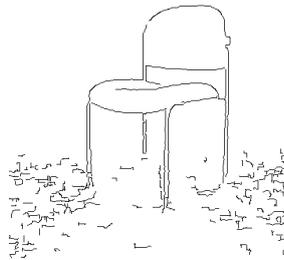
(b) Depth data



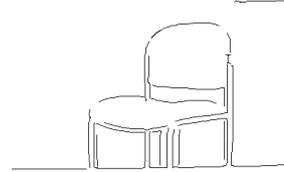
(c) Intensity data



(d) High confidence depth edges



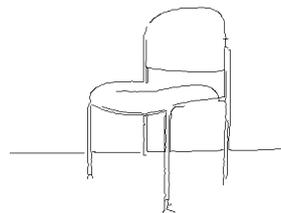
(e) Low confidence depth edges



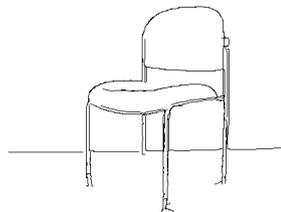
(f) Intensity edges



(g) As fig. 4.9d



(h) Fig. 4.9e and 4.9f smartly combined



(i) Fig. 4.9g and 4.9h combined

Figure 4.9: Overview of the steps in the edge detection procedure. The photo (fig. 4.9a) is included for a better comprehension of the scene.

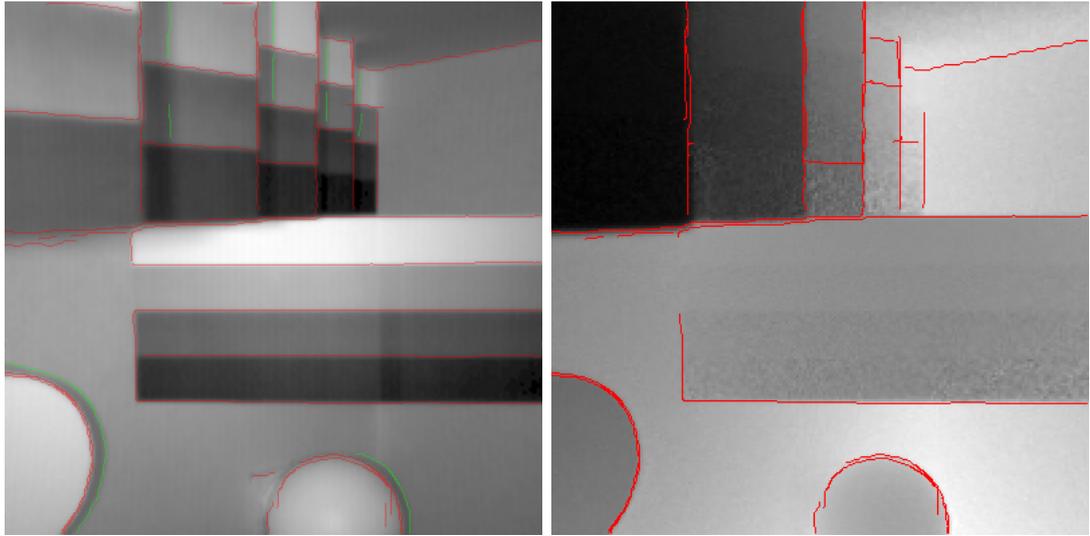


Figure 4.10: Intensity data (left) with marked intensity edges (red) and detected shadows (green). The second image (right) shows the depth data with the edges overlaid after applying the texture removal algorithm. Most of the texture edges can be removed successfully.

$N = 2$ pixels for example. The depth difference can then be calculated as

$$d_{diff} = \frac{1}{|E|N} \sum_{\vec{x} \in E} \sum_{j=1}^N |d(\vec{x} + j\vec{n}) - d(\vec{x} - j\vec{n})|, \quad (4.8)$$

where \vec{n} is the normal vector of the edge. The formula in Eq. 4.8 calculates the absolute value of the difference at each pixel and then adds the values along the edge. On first glance just taking the average depth on both sides seems more accurate, but the sign of the depth difference can change, like in the case of two ramps running in opposite directions next to each other. The local absolute value is a more robust measure, as long as the result is normalized with the length $|E|$ of the edge and the width N of the neighborhood. A threshold t_d has to be assigned to separate real depth edges and texture edges, where local noise or systematic errors can still result in a small depth difference (cf. chap. 6).

Averaging the depth difference along the whole detected edge is a big advantage compared to a local edge detection algorithm, that decides about a possible depth edge on a pixel scale. The algorithm can fail, if a texture edge turns into a depth edge along the way, without interruption.

4.5 Discussion

The combination of depth and intensity data, as well as geometric information of the scene and the camera setup has led to a combined approach for a depth edge detection in ToF data. This smart combination of provided information makes it possible to exceed the limits of standard edge detection algorithms as they are used in usual image processing.

The goal of the investigations was to provide reliable edge locations for adaptive denoising algorithms of the depth data, as in [32] or [47].

A problem of the approach is also connected to the nature of the ToF data. Although the depth and intensity data is measured with the same sensor, the location of detected edges of the same object can vary slightly and cause a double edge, when combining both into one edge map. These mismatched edges have two reasons.

First, the maximum gradient in the intensity data, which is usually considered to be at the border of an object, can appear slightly inside the object boundaries, if the object does not have a flat surface towards the camera, but the surface is convex and reaches an angle parallel to the optical axis. The camera mounted light source, as well as the reflectivity properties of the surface can cause a strong gradient on the object itself and make the actual transition to the background less pronounced, like the intensity data of the polystyrene sphere in fig. 4.10.

And second, the pixels that are neither completely part of the foreground, nor of the background and therefore mix the intensities or depth values, behave differently for depth and intensity data. While the intensities mix linearly according to the ratio of fore- and background, the depth value of such a mixed pixel is a combination of two complex numbers with phase and amplitude, that combine in this matter. The result of these different mixing behaviors are object boundaries that appear to vary slightly between the depth and the intensity data and can cause the location of detected edges to differ slightly between the intensity and depth data.

The decomposition of these mixed pixels for a more accurate edge location, as well as the removal of systematic errors in the depth data, which is not possible with smoothing algorithms that simply form the average over a local neighborhood, requires a thorough analysis of the raw sensor data, which is exactly what the next chapter will focus on.

5

Systematic Errors

To reach a better accuracy, reproducibility and consistency in the recorded depth maps than possible with local denoising algorithms like the bilateral filter (cf. chap. 3) or an adaptive method utilizing the depth edges from chap. 4, systematic errors in the depth data have to be eliminated.

These systematic errors include deviations of the depth data dependent on the real distances, the integration time, the intensity and the camera temperature. A number of publications have investigated these phenomenons and suggested calibration methods in recent years. [11] gives a good overview of the errors and publications. But with the exception of the distance related distance error, also called wiggling error, there exists no explanation for the origin of these errors. The calibration is usually a time consuming task and has to be repeated before every measurement, because of the temperature dependency.

This chapter takes another look at the systematic depth errors. Most of the previous publications are limited to an examination of the processed phase, amplitude and intensity data. This chapter reveals for the first time the common origin of the previously unexplained errors in the raw data of the sensor.

Because of the required raw data access, the investigations are limited to the CamCube 3. But the appearance of the same systematic errors in other ToF camera systems is a strong indicator for similar sources in other models.

Parts of the research in this chapter were already published in [46].

But

5.1 LED Modulation

The distance calculated from the camera shows a periodic deviation from the actual physical distance. This error is usually referenced as the “wiggling error” (cf. [49]) and considered to cause the largest deviation from the real distance for continuous

wave ToF cameras. It was already investigated in [29, 50, 43, 45, 13] for example.

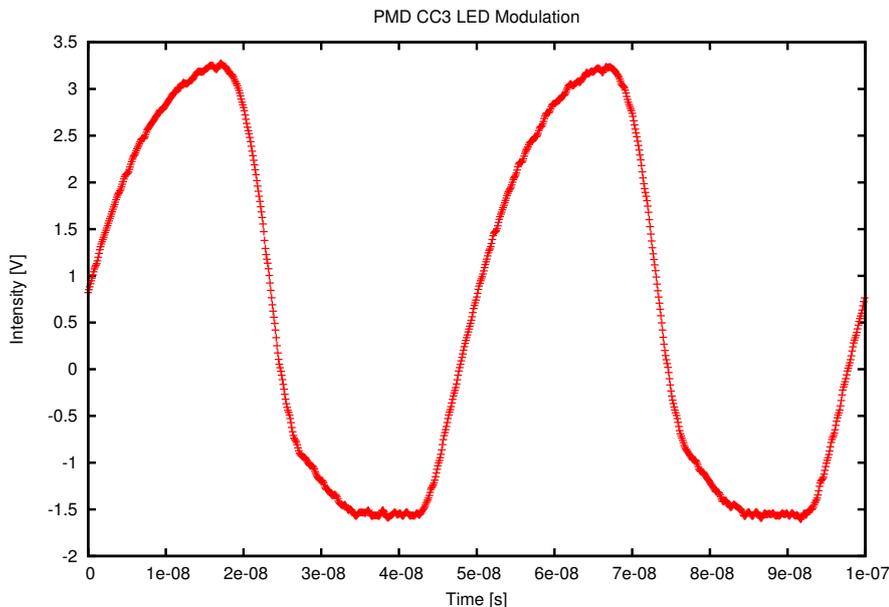


Figure 5.1: Actual intensity modulation of the PMD CC3 LED arrays, measured with a photo diode.

The primary reason for this error is a deviation of the illumination modulation from a perfect sine-wave (cf. fig. 5.1). The algorithms for the depth calculation usually assume a perfectly sinusoidal light signal but in the real modulation higher order frequencies up to at least 5 times the base frequency occur with a considerable amplitude (cf. fig. 5.2).

The work of Lange in [29] suggests that the higher harmonics are due to the fact that the LEDs are actually modulated with the same digital signal as the sensor, but the internal capacities cut off most of the frequencies above 20 MHz, resulting in an almost sine-shaped function with minor residuals of the higher order frequencies.

Because of the sensor modulation, which is equivalent to an integration over half of the main frequency period, the even multiples (40 MHz and 80 MHz) do not contribute to the measured intensities. The uneven multiples (60 MHz and

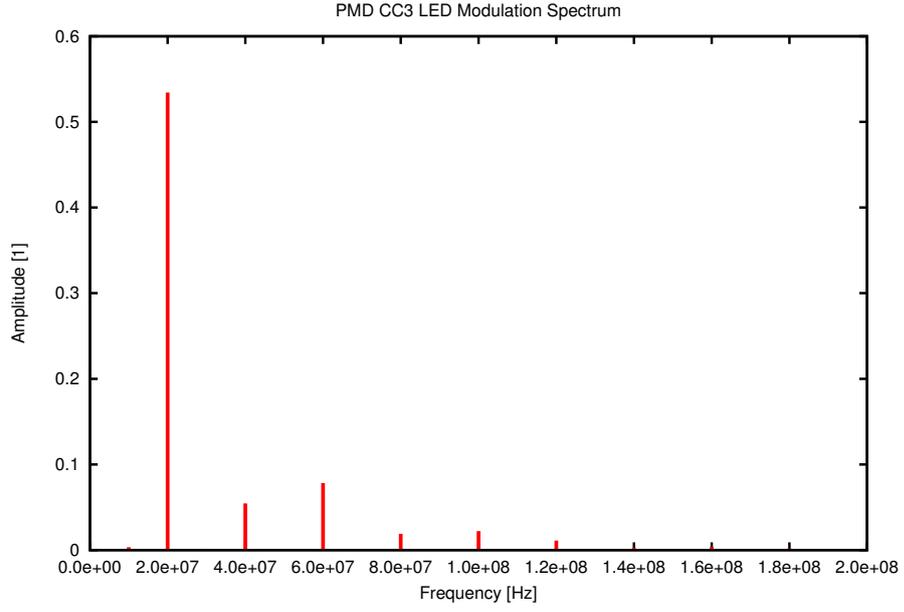


Figure 5.2: Frequency spectrum of the CC3 LED modulation, showing the higher harmonics in the signal.

100 MHz) on the other hand clearly change the measurement:

$$\begin{aligned}
 I(\theta_i) &= \frac{mT}{2} i_o + \sum_n \int_0^{mT} (i_n \sin(n(\omega t - \phi) + \phi_n)) \cdot H(\sin(\omega t + \theta_i)) dt \\
 &= \frac{mT}{2} i_o + \sum_n \int_0^{mT} (i_n \sin(n(\omega t - \phi - \theta_i) + \phi_n)) \cdot H(\sin(\omega t)) dt \\
 &= \frac{mT}{2} i_o + \sum_n m \int_0^{\frac{T}{2}} i_n \sin(n(\omega t - \phi - \theta_i) + \phi_n) dt \\
 &= \frac{mT}{2} i_o + m \sum_n \left[\frac{-i_n}{n\omega} \cos(n(\omega t - \phi - \theta_i) + \phi_n) \right]_0^{\frac{T}{2}} \\
 &= \frac{mT}{2} i_o + \frac{mT}{2\pi} \sum_n \frac{i_n}{n} (\cos(n(-\phi - \theta_i) + \phi_n) - \cos(n(\pi - \phi - \theta_i) + \phi_n)) \\
 &= \frac{mT}{2} i_o + \frac{mT}{\pi} \sum_{n=1,3,5,\dots} \frac{i_n}{n} (\cos(n(\phi + \theta_i) - \phi_n)),
 \end{aligned} \tag{5.1}$$

where ϕ_n denote the phase shifts of the different higher harmonics compared to the base signal. This greatly attenuates the influence of the higher frequencies and a least squares fit with the 3rd and 5th order frequencies is a quite accurate fit (cf. fig. 5.3). The resulting depth deviation has a periodicity of about 1/4 of the non-ambiguity range (cf. fig. 5.4) and 0.1 rad amplitude.

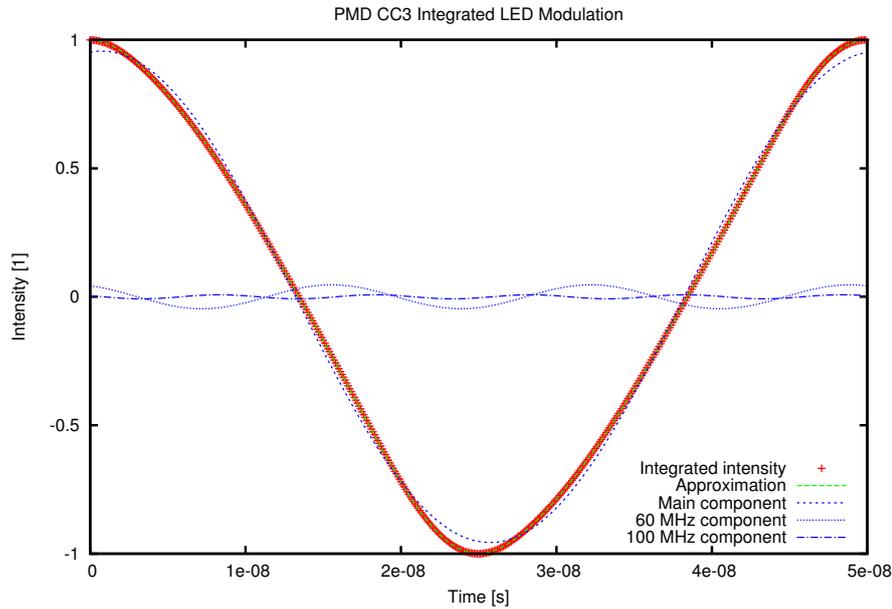


Figure 5.3: LED-signal integrated over 1/2 period. A least squares fit with harmonic functions of 20 MHz, 60 MHz and 100 MHz gives a very good approximation.

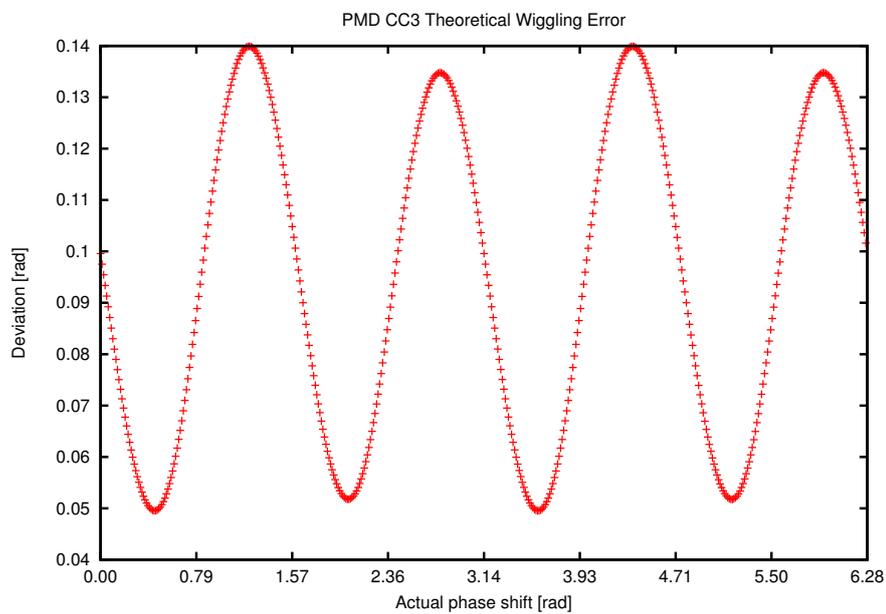


Figure 5.4: Theoretical wiggling error simulated with the measured LED modulation in fig. 5.1. The error has a periodicity of about 1/4 the non ambiguity range and an amplitude of about 0.1 rad, which corresponds to 1.88 m and 11.9 cm.

Theoretically the knowledge of the exact illumination modulation function, or an accurate enough approximation, should allow a permanent calibration of the wiggling error. Yet, no proposed calibration method has taken this approach and for good reason. As will be shown in the following sections, there are other error sources that result in a deviation of the same periodicity. So while the effects of the illumination can be compensated with exact modeling, a part of the wiggling error will remain.

5.2 Heat Model

A variety of publications has shown the sensitivity of ToF cameras to the camera temperature. In [22] a differentiation is made for the internal temperature, which reaches a steady state if the camera runs more or less at the same frame rate and the external temperature, that will cause an additional offset in the depth measurement. A plot (cf. fig. 5.5), previously published in [30] for example shows how the calculated intensity, amplitude and phase shift or depth of the CamCube 3 changes during the warm-up period of the device. Other papers discussing temperature effects are [21, 38]. But an explanation for the temperature dependency of the values has not been given yet.

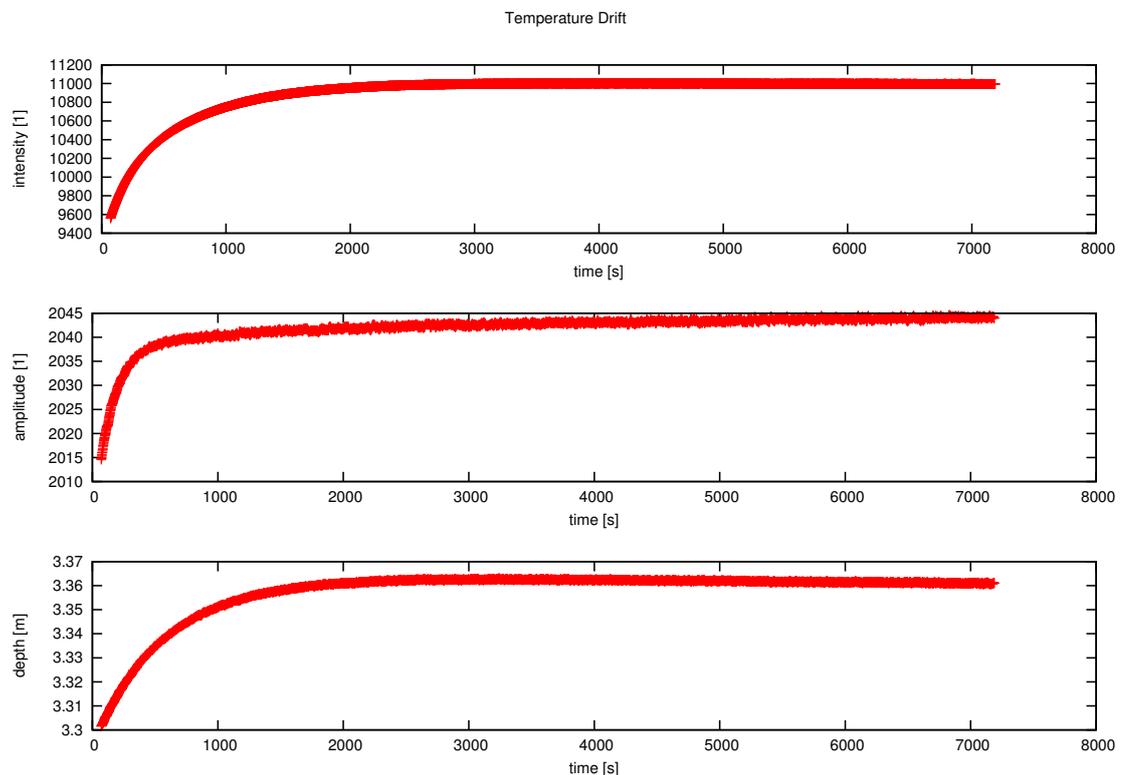


Figure 5.5: Change of intensity, amplitude and depth during warm-up of the camera. Previously published in [30].

To make reliable measurements of the influence of the integration time on the depth measurements, mentioned in [21] for example, it is necessary to record at

different integration times, but with the same camera temperature.

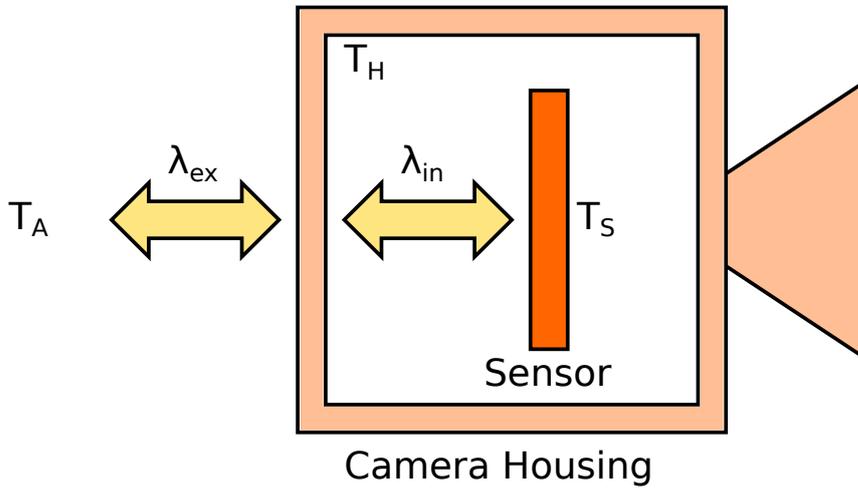


Figure 5.6: Schematic of the camera for the heat model. For the steady case and constant heat conductivities λ_{in} and λ_{ex} , the temperature of the sensor T_S is linearly dependent on the ambient temperature T_A and the housing temperature T_H .

The heat generated by ToF cameras during recording is quite substantial. Some of them even require active cooling. To understand and control the amount of thermal power with different recording parameters a heat model is formulated that assumes three major heat sources:

- the thermal power in standby P_0 ,
- the thermal energy released in one readout operation E_r and
- the thermal power when the sensor is active P_a .

While the contribution of the standby power is constant, the contribution of the readout operations depends on the frame rate at which the camera is operated and the contribution of the active sensor and camera depends on both, the frame rate and the current integration time. The total thermal power P_h can be summarized as:

$$P_h = P_0 + fE_r + 4 \cdot t_{int}fP_a \quad (5.2)$$

with the frame rate f and the integration time t_{int} .

If frequency and frame rate are kept constant, the camera and sensor will reach a steady state after the warm-up phase, assuming the heat conductivities λ_{ex}

between the camera and the surrounding and λ_{in} between the sensor and the camera housing (cf. fig. 5.6) are constant. In this steady state the heating power P_h and the cooling power P_c will be equal:

$$\begin{aligned}
 P_h &= P_c \\
 \Leftrightarrow P_h &= \lambda_{in}(T_S - T_H) \\
 \Leftrightarrow P_h &= \lambda_{ex}(T_H - T_A) \\
 \Leftrightarrow P_h &= \lambda_{ex}\Delta T.
 \end{aligned} \tag{5.3}$$

The actual sensor temperature can not be measured, but the measurement of the temperature of the camera housing and the ambient temperature is enough to determine relative sizes of the different thermal power sources. The length of the heat conducting paths are both unknown and irrelevant, which is why the conductivities λ_{in} and λ_{ex} can be considered as integrated values with the units [W/K].

Measurements of the ambient temperature and the camera housing temperature are carried out for different frame rates and integration times in the steady state (cf. tab. 5.1). The temperature of the camera housing was measured with an attached Greisinger GMH 3710 digital thermometer, the ambient temperature with a conventional alcohol thermometer.

A combination of eq. 5.3 and eq. 5.2:

$$\Delta T = \frac{P_0}{\lambda_{ex}} + f \frac{E_r}{\lambda_{ex}} + 4 \cdot t_{int} f \frac{P_a}{\lambda_{ex}} = P'_0 + f E'_r + 4 \cdot t_{int} f P'_a \tag{5.4}$$

can be used for a least squares fit of the relative thermal power parameters to the data in tab. 5.1. A plot of the data and the fit is shown in fig. 5.7.

f [Hz]	t_{int} [μ s]	T_A [$^{\circ}$ C]	T_H [$^{\circ}$ C]	ΔT [K]
0	0	24.1	29.9	5.8
0	0	24.5	29.7	5.2
10	1000	24.1	31.4	7.3
12.5	2000	24.5	31.8	7.3
12.5	4000	24.6	33.7	9.1
20	2500	24.6	33.8	9.5
20	4000	22.3	33.4	11.1
25	1000	24.0	32.3	8.3
25	1000	23.7	32.1	8.4
25	2000	24.7	34.2	9.5
25	4000	24.3	37.4	13.1

Table 5.1: Recorded data to fit the different contributions to the thermal power of the CC3.

The resulting values are:

$$P'_0 = (5.61 \pm 0.21)\text{K}, \quad E'_r = (0.051 \pm 0.015)\text{K s}, \quad P'_a = (14.8 \pm 1.2)\text{K}, \tag{5.5}$$

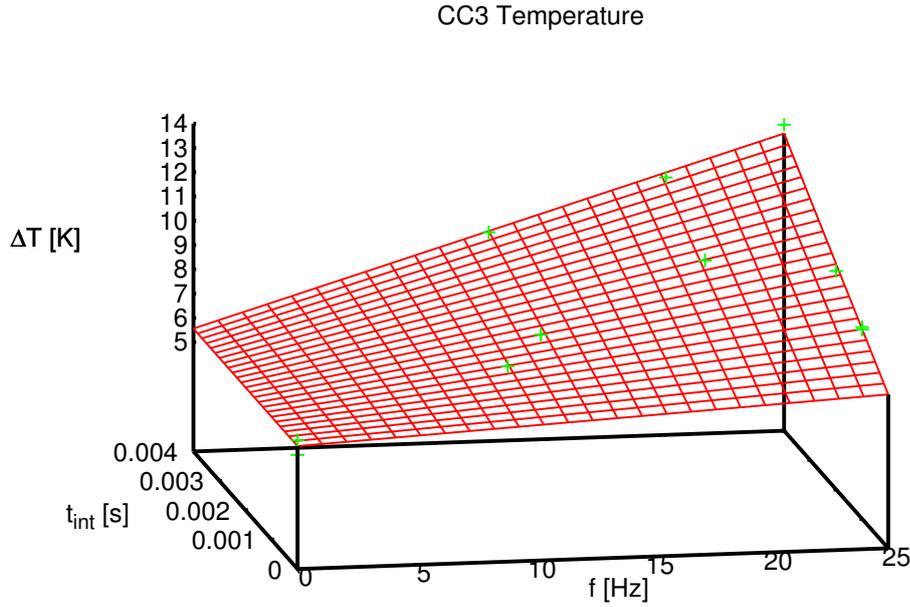


Figure 5.7: 3D Plot of the temperature data from tab. 5.1 with fit of the heat model from eq. 5.4.

which are the relative values, scaled with the heat conductivity λ_{ex} (cf. eq. 5.4). They suffice to compensate a change of the thermal power due to a different integration time with a changed frame rate. It is also interesting to note, that the fixed amount of thermal energy generated by the readout process for every frame is equal to an active sensor with an integration time of:

$$t_{int,eq} = \frac{E'_r}{4P'_a} \approx \frac{3400}{4} \mu\text{s} = 850 \mu\text{s}. \quad (5.6)$$

The exact adjustments for the compensation can be calculated with the help of eq. 5.4:

$$\begin{aligned} \Delta T_1 &= \Delta T_2 \\ P'_0 + f_1 E'_r + 4 \cdot t_{int,1} f_1 P'_a &= P'_0 + f_2 E'_r + 4 \cdot t_{int,2} f_2 P'_a \\ f_1 E'_r + 4 \cdot t_{int,1} f_1 P'_a &= f_2 (E'_r + 4 \cdot t_{int,2} P'_a) \\ f_2 &= \frac{f_1 (E'_r + 4 \cdot t_{int,1} P'_a)}{E'_r + 4 \cdot t_{int,2} P'_a}. \end{aligned} \quad (5.7)$$

The compensation is limited by the possible integration times (11-4000 μs) and frame rates (0-40 fps). High frame rates are also limited to shorter than maximum integration times.

The accuracy of the thermal parameters measured here is very low (cf. eq. 5.5). This is in part due to the long warm-up periods required to reach a steady camera

temperature, making the acquisition of data points for the fit very time consuming and in part also due to the limited accuracy of the measurements with the thermometer attached to the outside of the housing. But the acquired data is good enough for a minor thermal stabilization for a limited period of time and to separate the temperature dependency of the measurements from other parameters. A stabilized camera temperature over a longer period of time can not be granted anyway without active control of the ambient temperature.

5.3 Dark Signal

According to the EMVA Standard 1288 [9], the only camera parameter with a strong temperature dependency is the dark current. This indicates, that a temperature dependency of the ToF measurements must be strongly related to the dark current of the sensor. A recording of the intensity with a covered lens (cf. fig. 5.8) reveals a very similar profile and difference compared to the open lens intensity in fig. 5.5.

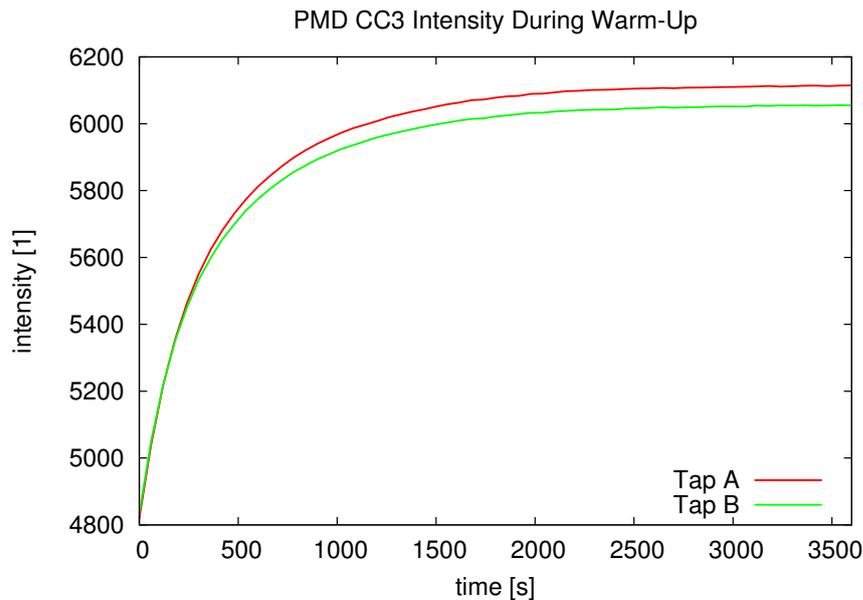


Figure 5.8: Dark signal of the different taps during camera warm-up. The plots show a large temperature dependency of the dark signal that changes for the two taps differently. The change of intensity is very similar to the measurement in fig. 5.5 without covering the lens.

A look at an image representation of a single subframe also reveals some very strong fixed pattern structures (cf. fig. 5.9). That is not a surprise and probably the result of variations in the pixel properties due to variances during the manufacturing process, the readout circuits or the heat propagation within the camera (cf. [1, 39]).

Unexpected is certainly the high intensity of the dark signal. In a scene with badly reflecting areas the mean value around 6000 of the dark signal can easily

account for the better part of the total intensity recorded there.

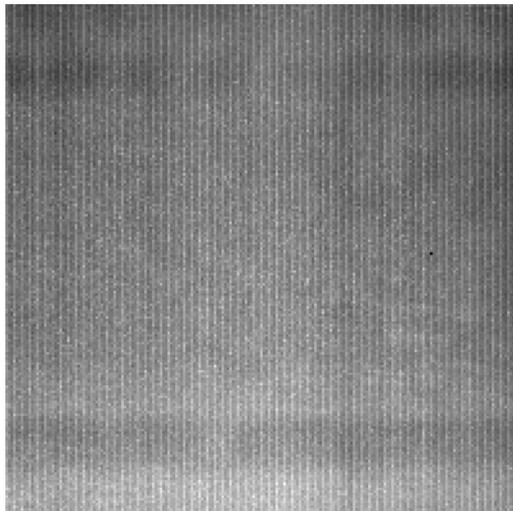


Figure 5.9: Dark signal of the CC3, tap A, first subframe, averaged over 100 frames. The mean value is at about 6000 with a standard deviation in the order of magnitude of 1%

the

An evaluation of the dark signal at different integration times (cf. fig. 5.10) shows, that the largest part of it is actually integration time independent and can therefore not be caused by a dark current. This behavior suggests to split the dark signal into a constant offset I_{do} and an integration time dependent dark current part I_{dc} :

$$I_{ds} = I_{do} + I_{dc}(t_{int}). \quad (5.8)$$

Because every I_i in eq. 2.8 is made up of measurements from both taps A and B, the different intensities of the signals (cf. fig. 5.8) do not contribute to any depth error. And while also the fixed pattern in the dark signal (cf. fig. 5.9) can not explain the temperature dependency of the depth measurements or any other systematic error, the recordings in fig. 5.10 also show an undeniable non-linearity of the integration time dependent part. The influence of this property will be discussed in the next section.

5.4 Signal Nonlinearity

The new heat model (cf. sec. 5.2) allows it to make measurements at different integration times but at the same sensor temperature. The expectations for such a measurement would be a linear dependency on the integration time. However, a measurement of the dark signal (cf. fig. 5.10) reveals an exponential behavior.

The measurements were made by warming up the camera for one hour with $t_{int} = 100 \mu\text{s}$ and repeated at two different frame rates $f_1 = 20 \text{ Hz}$ and $f_2 = 30 \text{ Hz}$. The frame rates for the measurements at higher integration times are adjusted according to eq. 5.2 to compensate the change in thermal power.

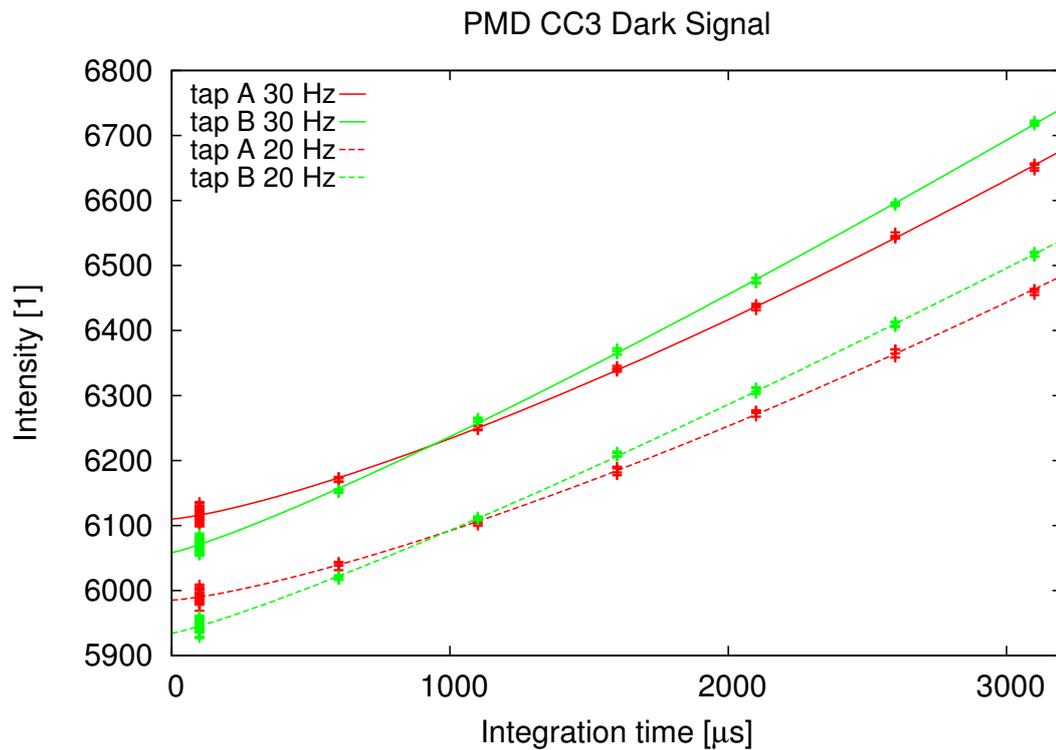


Figure 5.10: Dark current of a single pixel, averaged over 100 frames at two different camera temperatures and fit of $I(t_{int}) = (at_{int})^b + c$. The different heating power at different integration times was compensated by decreasing the frame rate according to eq. 5.7. The measurement at 100 μs was repeated successively to see how well the temperature compensation worked.

An exponential function of the form

$$I(t_{int}) = (a \cdot t_{int})^b + c = I_l(t_{int})^b + I_{do} \quad (5.9)$$

is fitted to the data and the resulting parameters can be found in tab. 5.2. It is interesting to note that the dark current a and the offset c change significantly between the two measurements, while the exponent b seems to stay constant. It can also be interpreted as a linear intensity I_l , subject to the exponent b and a constant offset I_{do} .

Tap	f [Hz]	a [1/ μ s]	b [1]	c [1]
A	20	0.034 ± 0.003	1.32 ± 0.03	5985 ± 2
B	20	0.081 ± 0.007	1.15 ± 0.02	5934 ± 2
A	30	0.040 ± 0.004	1.31 ± 0.03	6110 ± 2
B	30	0.089 ± 0.008	1.16 ± 0.02	6058 ± 2

Table 5.2: Fit results of the data in fig. 5.10. While the offset of the signal (c) and the dark current (a) increases significantly, the exponent (b) stays constant (a in 1/ μ s).

A look at the results for the whole image domain gives an average exponent of $\bar{b}_A = 1.32 \pm 0.08$ for Tap A and $\bar{b}_B = 1.19 \pm 0.05$ for Tap B.

To verify the constant sensor temperature, the measurement at 100 μ s is repeated consecutively between the other data points. There is a slight decrease of the dark signal there, indicating an overcompensation of the increased integration time (cf. fig. 5.11). This might cause the exponents to be slightly underestimated.

The influence of a non-linear photo response on the measured depth was already investigated in [49]. The simulations revealed a periodical offset of the depth with a periodicity of 1/4 the non-ambiguity range, just like the higher harmonics of the illumination. With exponents between 1 and 2 the influence on the depth is only in the range of millimeters.

So while the non-linearity of the sensor gives a minor contribution to the wiggling error, it offers no explanation for any of the unexplained errors like the intensity related distance error or the temperature dependency of the depth measurements.

5.5 Subframe Offset

A close look at the equation responsible for the phase shift calculation (eq. 2.8) should be enough to see, that only a systematic offset of the different intensity data I_i can cause an error in the depth data. A mere offset of the two taps will cancel out, as all I_i contain data from both taps. A non-linearity has very limited effects, because all of the values are affected by it. So there has to be something else that is different between the recorded subframes.

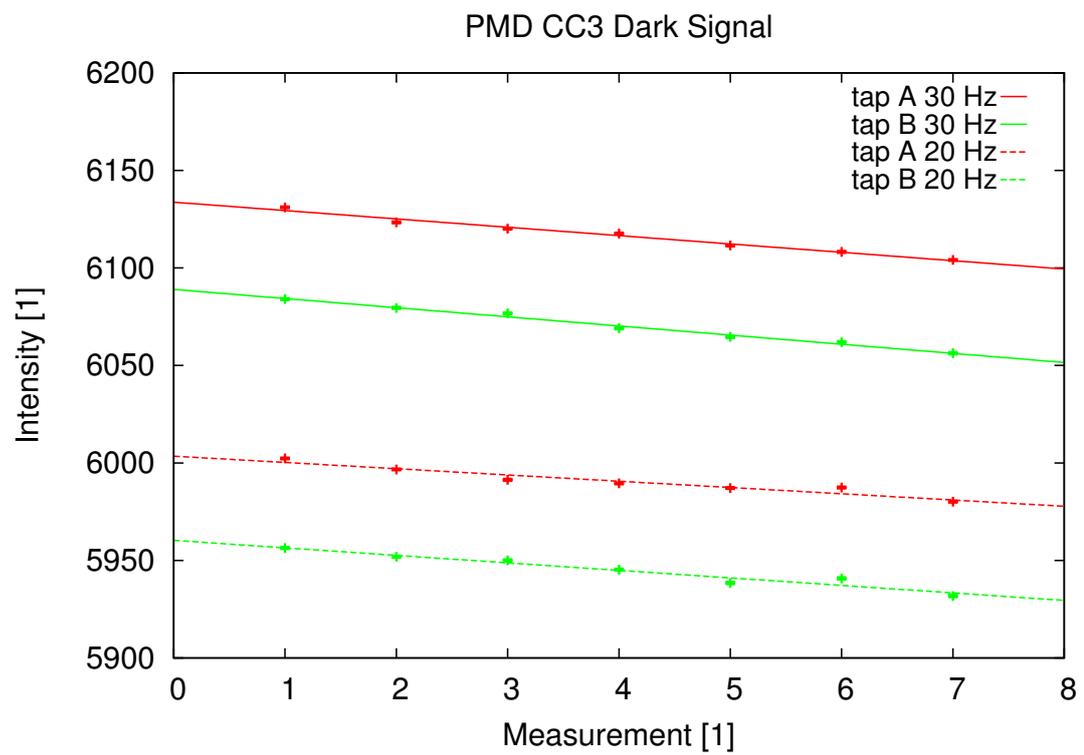


Figure 5.11: Decrease of the dark current at $t_{int} = 100 \mu\text{s}$ for the measurements in fig. 5.8. The slowly decreasing dark current indicates a slight overcompensation of the increased integration time. The lines are a linear fit to the data points.

A deviation in the recorded data is very hard to identify, because the recorded intensities are at very different levels due to the modulated illumination and sensor. But another look at the dark signal reveals a very interesting behavior.

Once the camera is in a steady state with respect to the temperature, the dark signal is only affected by Poisson noise. The signal to noise ratio can be greatly improved by averaging over several recorded frames. A look at the different taps and subframes of a pixel is shown in fig. 5.12. The data, recorded in a thermal steady state shows a small but clear offset between the different subframes. The two taps show a very different behavior.

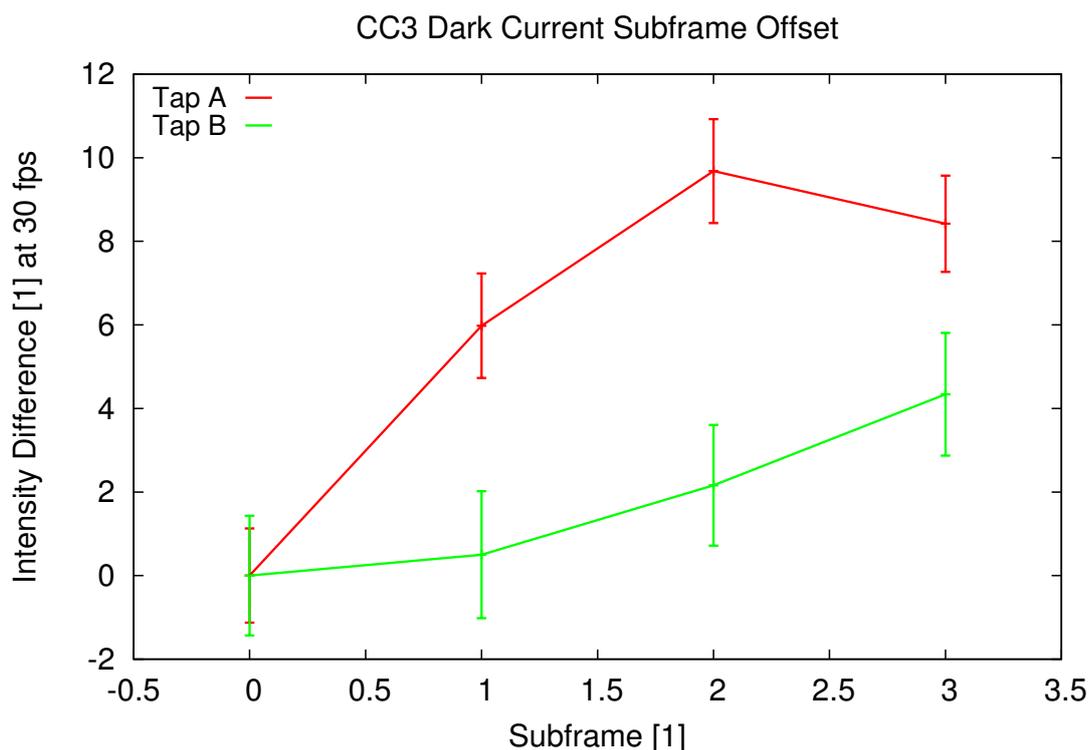


Figure 5.12: Dark signal difference of the subframes compared to the first subframe. The data is averaged over 100 frames at a single pixel, to reduce the Poisson noise. The error bars show the statistical error of the mean value for each subframe. The data is recorded at a frame rate of $f = 30$ Hz and $t_{int} = 200$ μ s integration time.

To better understand what is going on a precise time scale for the recorded subframes is very important. To determine the delay of the different subframes at different integration times, a textured cardboard attached to a motor rotating at different speeds is recorded (cf. fig. 5.13). The analysis reveals a constant time Δt of about 3.75 ms in between the consecutive subframes (cf. fig. 2.1).

The different subframes can now be put into an exact time scale. Further measurements of the dark current difference at different integration times and frame rates (cf. fig. 5.14) show, that the offsets change with these parameters. There is a tendency for larger differences at slower frame rates and shorter integration times. Every pixel and tap has a slightly different profile, but in the thermal steady state the offsets remain constant.

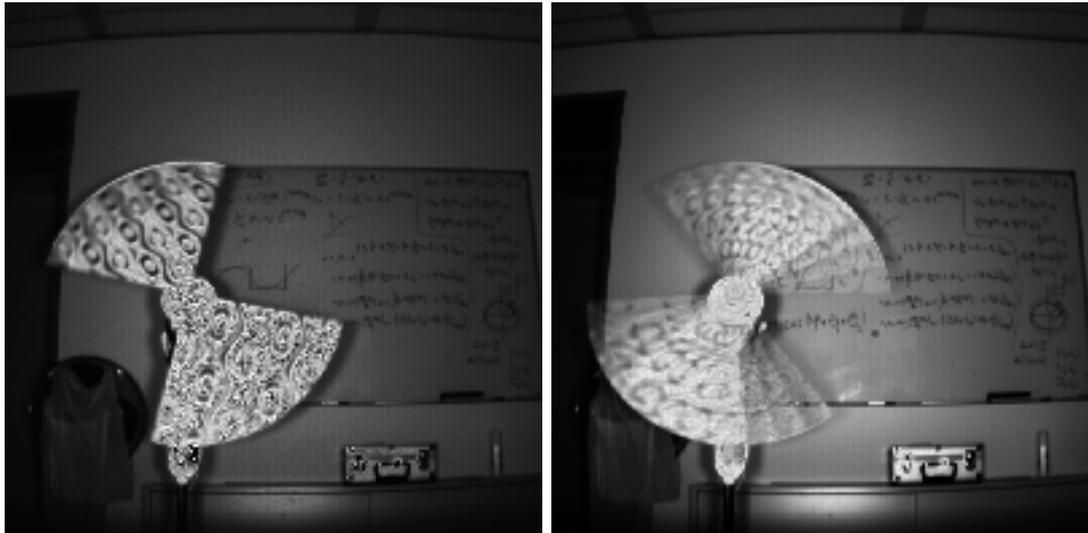


Figure 5.13: Single subframe (left) and intensity image (right) of a textured cardboard mounted to a motor to determine the time between subframes. An analysis reveals a constant time of about 3.75 ms between the end of one and beginning of the next subframe, which is about the length of the maximum possible integration time of the CC3 (4000 μs).

PMD CC3 Subframe Deviation

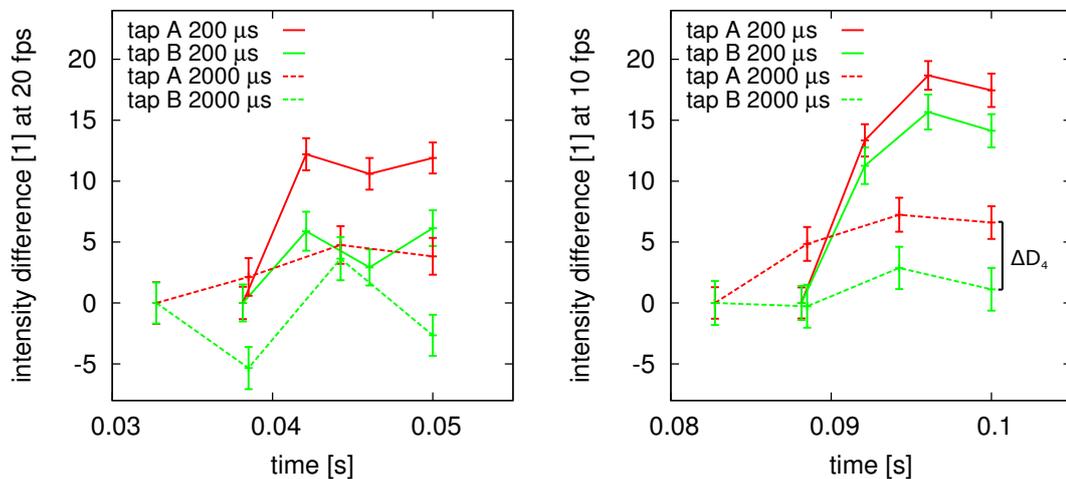


Figure 5.14: Dark signal difference of the subframes compared to the first subframe. This shows a clear increase in intensity, probably due to a short-term thermal effect. The effect is more pronounced for short integration times (solid lines: $t = 200 \mu\text{s}$) and less for long integration times (dashed lines: $t = 2000 \mu\text{s}$). The x-axis shows the time passed since the end of the last frame, making the sets recorded at different frame rates distinguishable and giving an idea of the temporal distribution of the subframes.

A possible explanation for the offsets is a short-term temperature effect. The thermal power of the active sensor found in sec. 5.2 could generate an oscillating temperature in the pixels, causing a periodic variation of the dark current. A possible heat source are the currents required for the sensor modulation at 20 MHz. But the largest part of the heat for short integration times is generated by the readout process, which is about as much as the sensor being active for 3400 μs (cf. eq. 5.6). The heat generated by the active sensor can be assumed to be generated while the frames are recorded. But when exactly the heat of the readout process is generated or how it is spread over the period of the recording of a whole frame can not be easily determined.

While the general camera temperature at low frame rates or short integration times is lower, a local heat source only active for a limited time would cause a stronger temperature gradient than at a higher frame rate or integration time, where the passive periods of the sensor are shorter and the opportunity to cool down is reduced. Thus, the dark current shows a stronger variation for lower frame rates and shorter integration time. For long integration time a large part of the heat is generated over an elongated time period.

In the following section the influence of these subframe offsets on the depth calculation will be discussed.

5.6 Effects on the Phase Calculation

The variations in the dark current uncovered in the previous section look rather small on first glance, with an intensity change of about 5 to 15 counts. And the combination of the different subframes (cf. tab. 5.3) for the intensities I_i of eq. 2.8 reduces it even further.

	tap A	tap B
I_0	subframe 0	subframe 2
I_1	subframe 1	subframe 3
I_2	subframe 2	subframe 0
I_3	subframe 3	subframe 1

Table 5.3: Overview of how the subframes combined into one frame I_i for the calculation of ϕ as in eq. 2.8 (cf. fig. 2.1).

A reformulation of eq. 2.8 including the different subframes can help to understand the influence of the offsets on the calculated depth:

$$\begin{aligned}
\phi &= \arctan2((I_3 - I_1), (I_0 - I_2)) \\
&= \arctan2(((I_{A3} + I_{B1}) - (I_{A1} + I_{B3})), ((I_{A0} + I_{B2}) - (I_{A2} + I_{B0}))) \\
&= \arctan2((I_{A3} + I_{B1} - I_{A1} - I_{B3}), (I_{A0} + I_{B2} - I_{A2} - I_{B0})) \\
&= \arctan2(((I_{A3} - I_{B3}) + (I_{B1} - I_{A1})), ((I_{A0} - I_{B0}) + (I_{B2} - I_{A2}))) \\
&= \arctan2((\Delta I_3 - \Delta I_1), (\Delta I_0 - \Delta I_2)), \tag{5.10}
\end{aligned}$$

where ΔI_i denotes the difference of the two images recorded in the two taps A and B in the same subframe i . This reformulation shows that only the differences of the two taps of each subframe contribute to the arguments. Likewise only the differences of the subframe offsets $\Delta D_i = D_{Ai} - D_{Bi}$ (cf. fig. 5.14) will matter:

$$\begin{aligned} \phi &= \arctan2(((\Delta I_3 + \Delta D_3) - (\Delta I_1 + \Delta D_1)), ((\Delta I_0 + \Delta D_0) - (\Delta I_2 + \Delta D_2))) \\ &= \arctan2(((\Delta I_3 - \Delta I_1) + (\Delta D_3 - \Delta D_1)), ((\Delta I_0 - \Delta I_2) + (\Delta D_0 - \Delta D_2))) \\ &= \arctan2(((\Delta I_3 - \Delta I_1) + (\Delta D_3 - \Delta D_1)), ((\Delta I_0 - \Delta I_2) + (-\Delta D_2))). \end{aligned} \quad (5.11)$$

ΔD_0 is 0 because the first subframe is used as the reference point.

Modeled intensities of different subframes (cf. fig. 5.15) with exemplary subframe offsets from fig. 5.14 exhibit only a very small change due to these offsets. In the well exposed range of a couple of thousand counts over the offset it is not noticeable at all. Only a zoom in on the very low intensity area of the simulation shows how the different offsets mix up the order of the recorded intensities.

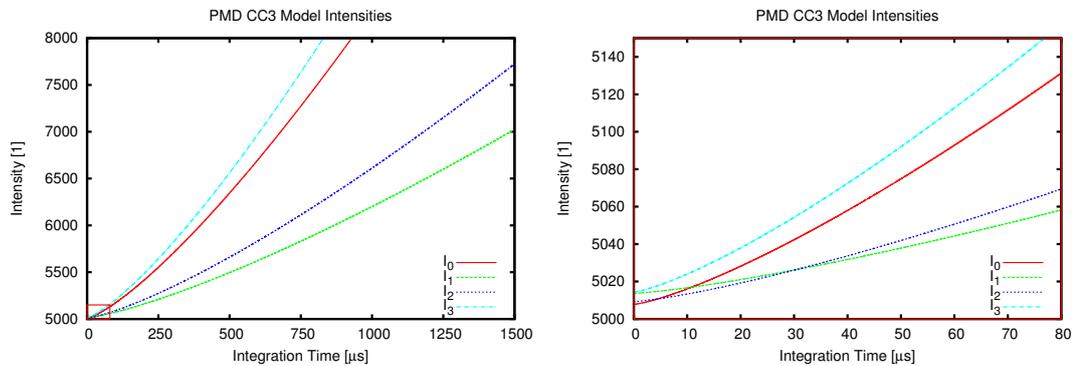


Figure 5.15: Subframe intensities, modeled as $I_i(t_{int}) = (a_i \cdot t_{int})^b + c_i$. The a_i are calculated from a cosine function with an offset of 0.4, and amplitude of 0.2. The exponent b is set to 1.3, the phase shift to 1 rad. The subframe offset values approximately correspond to those in fig. 5.14 at $f = 10$ Hz and $t_{int} = 200$ μ s are added to these intensities. The deviation only shows in the very low integration time or intensity range (right), where the intensities actually change their order.

Despite this very small effect on the recorded intensities, the influence on the calculated phase shift is noticeable even for higher intensities (cf. fig. 5.17). For an increasing integration time or intensity, the influence of the subframe offsets disappears and the phase shift converges towards the result only incorporating the non-linearity. But towards low intensities the calculated phase deviates more and more.

Essentially the calculated signal is a superposition of the optical signal recorded by the camera and a second signal of the subframe offsets (cf. fig. 5.16). Both of these have a phase and an amplitude. Usually, the signal of the subframe offsets is very weak and hardly has an influence in the superposition. Only for low amplitudes of the optical signal it will cause an increasing error. In the thermal steady state it is constant within the limits of the noise and for a vanishing intensity of the optical signal it will be the phase of the subframe offsets that is calculated.

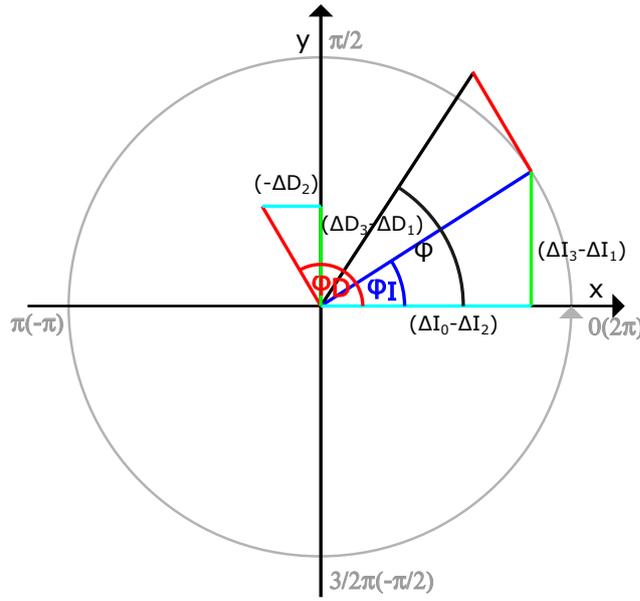


Figure 5.16: Phase diagram of combined optical signal (I) and signal off subframe offsets (D) according to eq. 5.11.

Very important to note here is, that the subframe offsets vary from pixel to pixel and change with the integration time, frame rate and temperature. The error induced by it also oscillates with the distance, because the offset phase ϕ_D is fixed. The induced error is the largest for a phase difference of $\pi/2$ or $3\pi/2$. A plot of the simulated error with respect to the phase shift is shown in fig. 5.18. The periodicity is 1/2 the non-ambiguity range and thus different from that of the higher harmonics (cf. sec. 5.1) or the error induced by the nonlinearity (cf. sec. 5.4).

The two simulations in fig. 5.17 also show, that although the absolute values of subframe offsets for the 10 Hz measurement are higher, the influence on the calculated phase is larger with the values from the 20 Hz measurement. Real measurements will of course differ greatly from the simulation, because the intensity and amplitude will hardly be the same for the whole non-ambiguity range. And also the non-sinusoidal modulation was not incorporated into the model.

The exemplary simulations given here show, that a description of the errors the subframe offsets will cause is very complicated because of all the parameters that have to be taken into account. The good thing about the subframe offsets is, that they can be measured with a relatively small effort. This offers the foundation of a new and very quick calibration method that will be proposed in the next chapter.

(eq. 2.8)

the

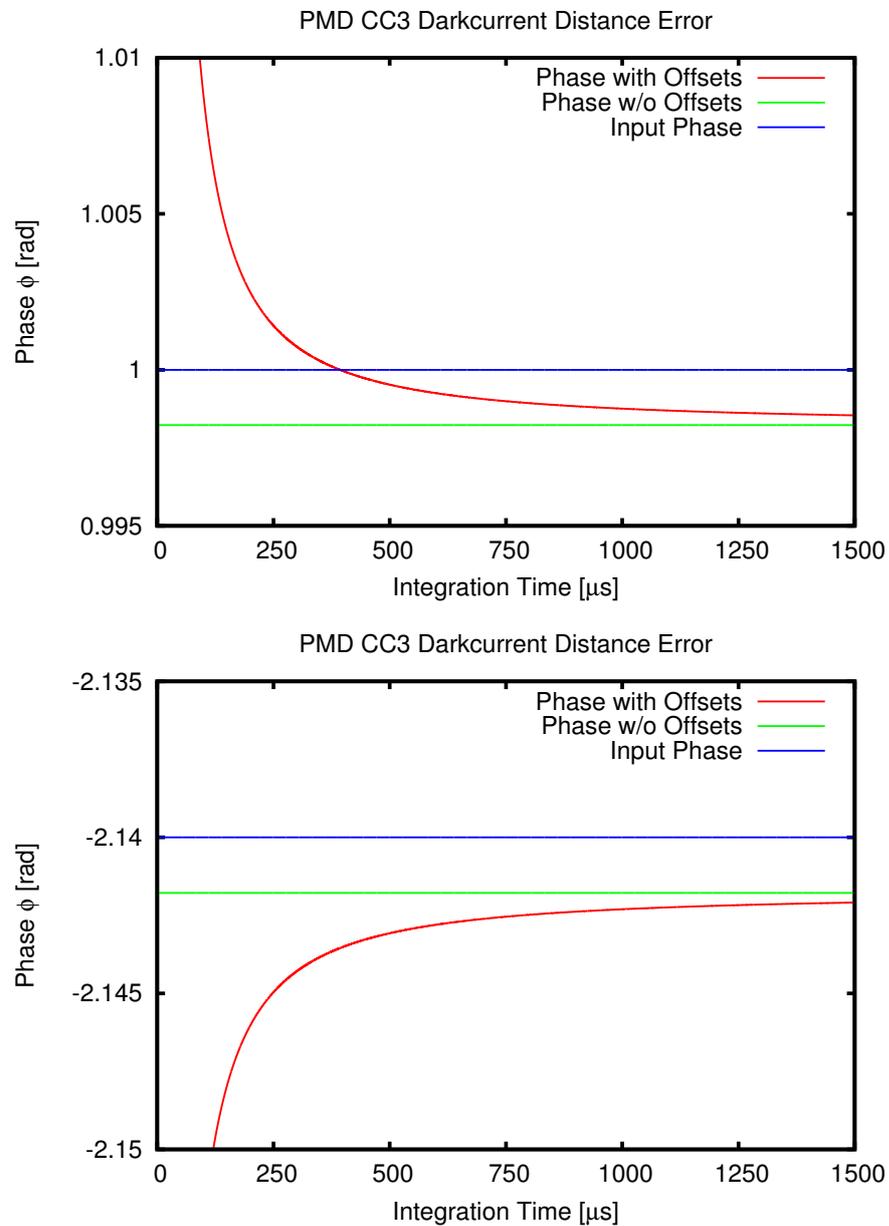


Figure 5.17: Phase $\phi = \arctan2((I_0 - I_2)/(I_3 - I_1))$ with (red) and without (green) the offset variations. The blue line is the input phase shift. The plots show, that the phase deviation due to the subframe offsets is also dependent on the input phase shift. It can either reduce or increase the deviation due to the non-linearity (cf. sec. 5.4).

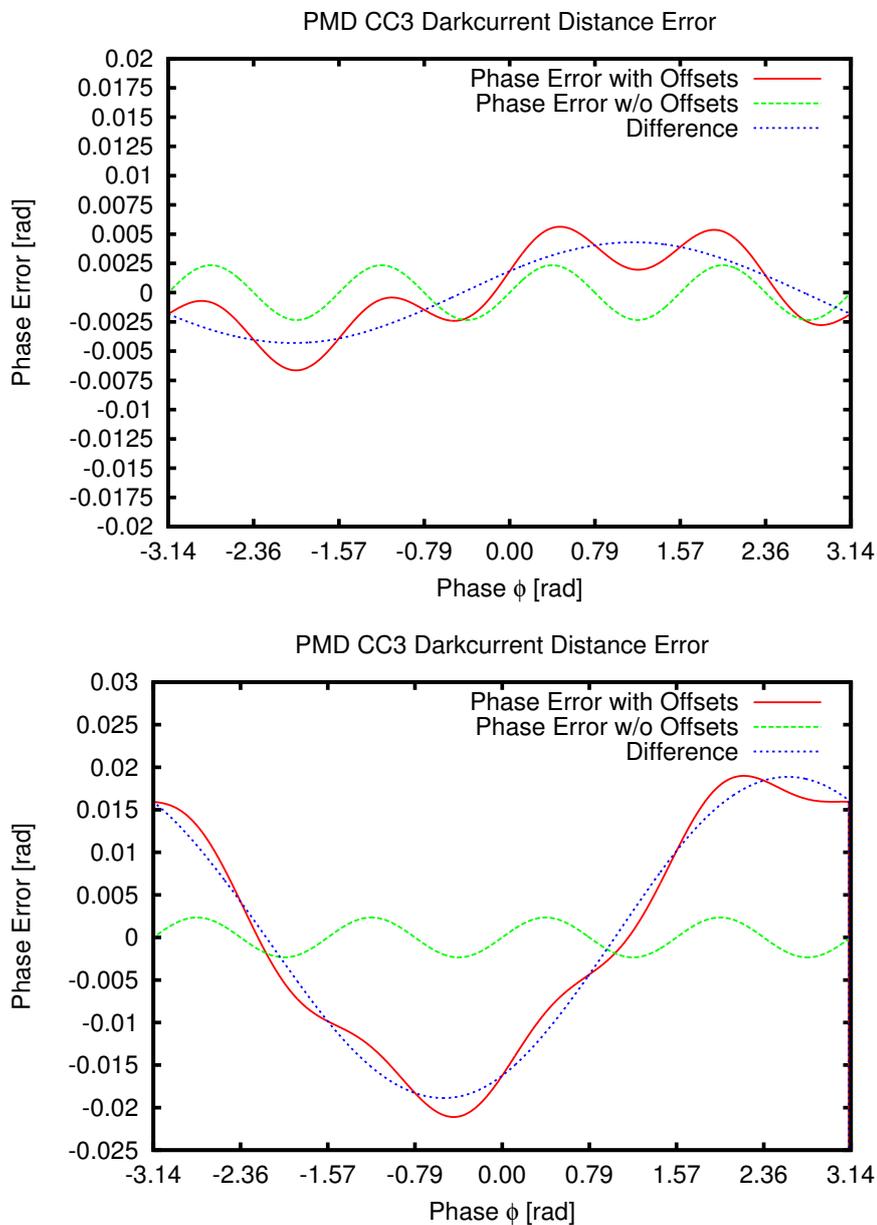


Figure 5.18: Phase error due to the non-linearity (green) and additionally the subframe offset. The data is modeled with the same parameters as in fig. 5.17 at a fixed integration time of $200 \mu\text{s}$ (top). The subframe offsets add another periodic deviation with a periodicity of $1/2$ the non-ambiguity range (blue). The bottom plot shows the error with the same parameters except for the subframe offsets, which are from the 20 Hz recording in fig. 5.14.

6

Calibration

In the previous chapter a couple of signal or sensor properties were investigated. The new insight gained there allows a new fast and flexible calibration of the camera raw data. Previous suggestions for ToF camera calibration (cf. [34, 35, 52, 21, 33, 18, 39]) were limited to the a posteriori data and done in the form of polynomial or spline fits, or with lookup tables to compensate the deviations for different integration times and intensities.

The raw data based approach presented here is not an absolute calibration but removes or reduces the effect of temperature dependency, integration time dependency and fixed pattern depth noise. It is still necessary to repeat parts of the calibration before every measurement, but the data required is recorded with a covered lens. There are no calibration targets required and the process is a matter of seconds.

A brief description of the method was already published in [46].

6.1 Calibration Model

The model used for the calibration is an intensity following eq. 5.9 but extended by a differentiation of the light current signal I_{lc} and the dark current signal I_{dc} . To better differentiate the parts of the signal and the dependencies on the parameters, they are added to the intensity components in brackets. t_{int} as integration time is obvious, θ denotes the internal phase shift or the respective subframe and m differentiates between the taps A and B:

$$I(t_{int}, \theta, m) = I_{off}(m) + (I_{dc}(t_{int}, \theta, m) + I_{ls}(t_{int}, \theta, m))^{b(m)}. \quad (6.1)$$

To calibrate the model there are 3 measurements that have to be made:

1. a determination of the exponent b of each tap

2. a measurement of the offset $I_{off}(m)$ of each tap
3. a measurement of the dark current intensity $I_{dc}(t_{int}, \theta, m)$ of each tap and subframe.

Since the exponent $b(m)$ is a temperature independent parameter it can be acquired beforehand and saved in a parameter file for each pixel and tap.

The offset $I_{off}(m)$ of each pixel and tap can be approximated by a dark signal measurement at the shortest integration time the camera allows (11 μ s in case of the CC3).

Now the dark current intensity $I_{dc}(t_{int}, \theta, m)$ is the most important part because it incorporates the subframe differences of the short-term temperature drift. It can also be recorded by a dark signal measurement of which the offset $I_{off}(m)$ has to be subtracted.

It is important to separate offset and dark current since the dark current is subject to the exponential function while the offset is not. Because of their temperature dependency both the offset and the dark current should be measured just before the actual measurement, after the camera had enough time to reach a thermal steady state.

Both values are subject to noise and to reduce the effect of it an average value should be recorded for several frames. Depending on the frame rate this requires a different amount of time. For the measurements presented in the following section 200 frames were recorded for each value, requiring 10 seconds for each measurement at 20 Hz. Combined with the effort of putting a cap on the lens and back off, the calibration process can be finished within 30 seconds. The measurement of the offset should not be extended for too long because it is recorded at a different integration time and would change the camera temperature in the long run.

With the three parameters known, eq. 6.1 can be transformed to give the linear incident light dependent signal that is actually relevant for the depth calculation:

$$I_{lc}(t_{int}, \theta, m) = (I(t_{int}, \theta, m) - I_{off}(m))^{1/b(m)} - I_{dc}(t_{int}, \theta, m) \quad (6.2)$$

6.2 Experiments and Discussion

The following figures show the effects of the calibration on the raw and processed data. The data was recorded at an integration time $t_{int} = 130 \mu$ s and is averaged over 200 frames to reduce the Poisson noise. In fig. 6.1 the raw and phase data is shown uncalibrated (left column) and calibrated (right column). As raw data two subframes of tap A (top row) and tap B (center row) with an identical internal phase shift of $\theta = \pi$ are picked.

The calibration greatly increases the apparent noise in the intensity subframes, since the pixels are not calibrated photometrically, but only the consecutive subframes of each tap are calibrated against each other and linearized. Because the depth is calculated on a pixel base, calibrating pixels against each other is not important for the depth measurement. The fixed pattern vertical stripes, very noticeable in the uncalibrated tap A image (top left) are completely gone after the calibration.

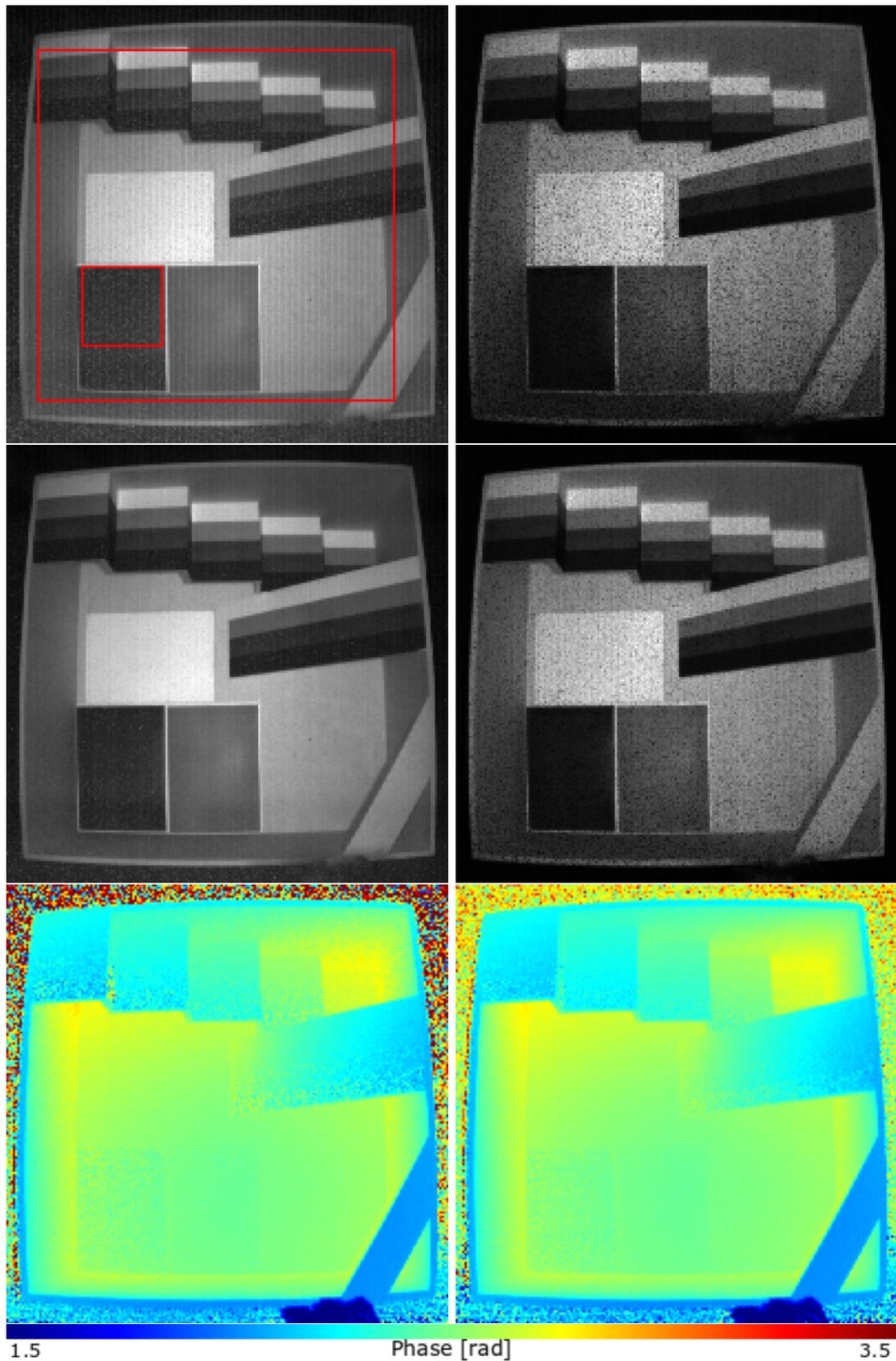


Figure 6.1: Raw data and phase before (left) and after (right) calibration. The raw intensity images (gray) are normalized separately. The phase images (bottom row) share the same color scale. Crops marked in the top left can be found in fig. 6.2 and 6.3.

A look at the phase data (bottom row) calculated from the raw values with eq. 2.8 shows that the noise is reduced here, especially in the areas of lower intensity. Tab. 6.1 gives a bit of statistics about the phase data. The strong decrease of the variance is a result of the decreased noise in the background. The mean phase value stays almost the same, as can be seen in the phase image. The mean depth of the different box parts is also the same before and after the calibration.

value	uncalibrated	calibrated	difference
mean	2.47	2.44	0.035
minimum	0.80	0.30	-2.86
maximum	5.31	5.35	4.23
variance	0.11	0.05	0.053

Table 6.1: Statistics of the phase data in fig. 6.1 in [rad]. Interesting to note is the decrease in the variance of the phase by 50%, while the total range of the data actually increases. Also the mean phase changes only slightly.

These effects become even more obvious with a look at a crop of the box interior in fig. 6.2 and tab. 6.2. Because of the better exposure here the noise is a lot weaker even before the calibration, compared to the dark area around the box. The major part of the variance is due to the different distances of the box interior and the reduced noise does not show up in a change of the variance. The missing shift of the mean phase shows, that the calibration does not affect the average depth.

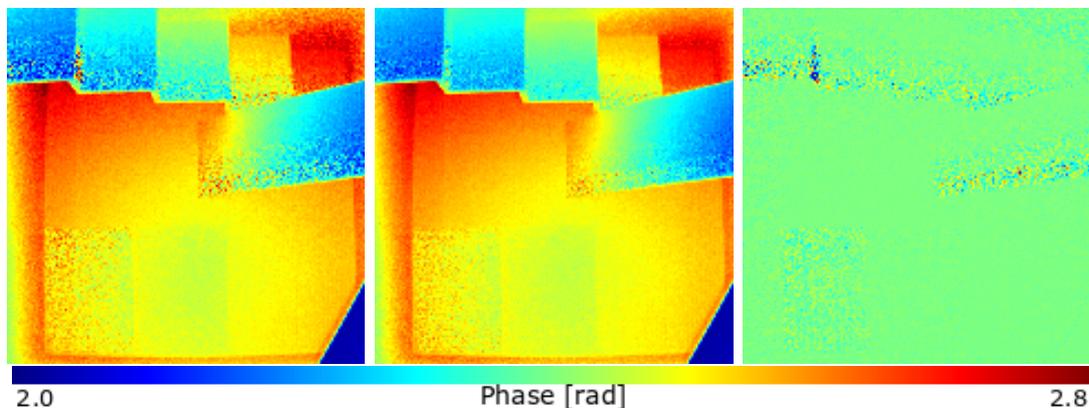


Figure 6.2: Crop of the phase images from fig. 6.1. Both images use the same color scale again. The lower noise in the calibrated data (center) is more obvious now. A difference image (right) shows the most changes happen in the areas with low intensity (cf. fig. 6.1). The color scale for the difference image uses the same color spread as the phase images, but is centered at 0.

It can also be seen in fig. 6.2, that the calibration has the strongest effect in the dark and areas. Because of the weaker intensity and amplitude, the Poisson noise has a much larger magnitude here (cf. eq. 2.15). But also the influence of the subframe offsets is increased as predicted in sec. 5.6. The result is a strong reduction of the noise in the depth data by the calibration.

value	uncalibrated	calibrated	difference
mean	2.482	2.478	0.004
minimum	2.00	2.02	-0.31
maximum	3.15	2.79	0.77
variance	0.018	0.017	0.001

Table 6.2: Statistics of the phase data in fig. 6.2. While the changes are noticeable in the images to the naked eye, there is hardly a change to the statistics here because the highly noisy areas around the box are cut off.

In a crop of the flat dark patch in the lower left corner the calibration reduces the variance of the phase from 0.0016 to 0.0010 (cf. fig. 6.3). Since the variance of the phase values is governed by the noise and not by actual 3D-structures the calibration affects the variance noticeably here.

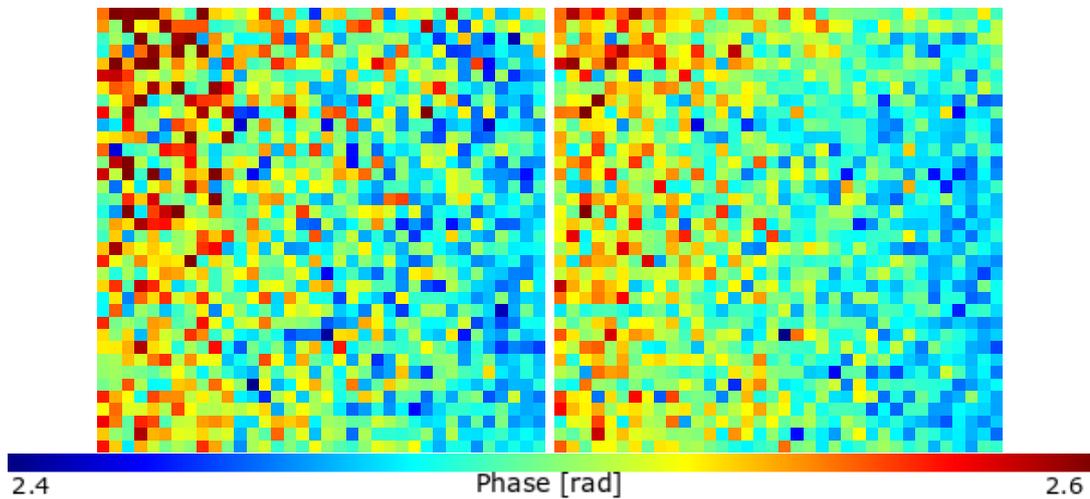


Figure 6.3: Small crop of the flat dark patch in the lower left corner of the phase images from fig. 6.1. Both images use the same color scale again. The variance of the phase in this flat area is reduced considerably from 0.0016 to 0.0010 (cf. tab. 6.3) by the calibration.

The fact, that the dark patches in the bottom left corner still exhibit clear edges in the depth data (cf. fig. 6.2) reveals that, while the error caused by the subframe offset is intensity dependent, it does not contribute to a systematic shift of areas with different intensities. The explanation is simple: the specific error by which the subframe offset changes the measured depth depends among other things on the exact relation of the subframe offsets. These are not uniform from pixel to pixel and slight variations can make a huge difference. So there must be yet another source for the intensity related distance error. One of these error sources is investigated in the next chapter.

value	uncalibrated	calibrated	difference
mean	2.501	2.498	0.003
minimum	2.40	2.40	-0.12
maximum	2.69	2.63	0.17
variance	0.0016	0.0010	0.0011

Table 6.3: Statistics of the phase data in fig. 6.3.

7

Internal Scattering

It is found (cf. [30]), that a bright object in the foreground will cause the whole background in a ToF image to appear closer to the camera than without the bright foreground object, even if it is not directly visible to the camera. The source of this behavior is usually suspected to be in the lens, where the incident light can deviate from the theoretical path due to multiple reasons: scattering in optical glasses, reflections on optical or other surfaces, diffraction on aperture blades or reflections at the sensor or film (cf. [53, 44, 28, 36, 30]). All these possible effects, also known as lens flare are very hard to measure or to model, which is why a simplified homogeneous diffuse scattering is assumed for the scattering model presented here.

For standard imaging and image processing diffuse scattering is not of very high importance. In the case of Time-of-Flight imaging on the other hand, the effects can cause serious errors in the recorded data.

Previous work on in-camera scattering (cf. [40, 24, 23]) investigated the effects of scattering on processed data (amplitude and phase) while here the raw data is analyzed directly. Furthermore the previous papers assumed the scattering to be mainly due to internal reflections between the sensor and the rear surface of the lens. The resulting calibration and correction algorithms are highly complicated and claim to decrease the scattering effect by 70% in the most recent work, but there are areas in the image with no apparent improvement at all.

The approach described here and previously published in [46] improves the estimated depth in every area of the scene. It must be noted though, that different cameras were used for the former publications, which inhibits a direct comparison. While the occurrence of multiple reflections can be ruled out as a dominant effect and diffuse scattering is considered more accurate for the CamCube 3, the findings can not be transferred to other systems without reservation.

In this chapter all I_{xx} denote intensity values of single pixels, taps and sub-frames. The \bar{I}_{xx} are the mean values of the whole image domain, but still only of

a single tap and subframe.

7.1 Decomposing the Signal

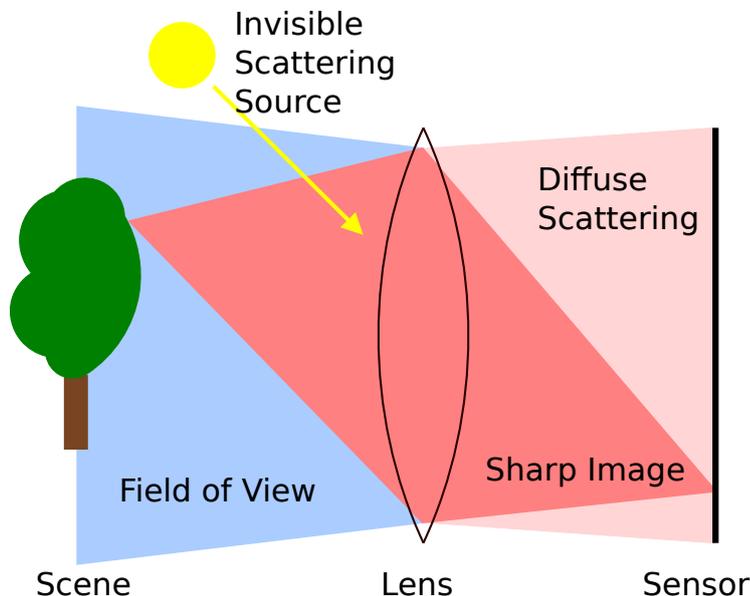


Figure 7.1: Diffuse scattering in a lens. In the model presented here it is assumed that a fraction of the incident light is spread homogeneously over the sensor. Invisible scattering sources have to be avoided, because the model can not correct their contribution to the signal.

If it is assumed (cf. fig. 7.1), that a fraction of the incident light from the scene is scattered diffusely inside the lens of the camera and spread out over the sensor, this effect will change the raw data of each pixel and tap. The effect on the calculated phase will be more severe for dark areas because the additional intensity is larger in comparison. It will of course also depend on the difference in distance of the pixel under consideration and the mean distance of the whole scene, weighted with the intensity of each point.

The scattering contributes to the light signal I_{lc} (cf. Eq. 6.2), which can now be separated into a sharp optical image of the scene I_{lci} and the scattered light I_{lcs} :

$$I_{lc} = I_{lci} + I_{lcs}. \quad (7.1)$$

In the approach presented here a uniform spread of the scattered light over the whole sensor is assumed, making the scattered part of the intensity I_{lcs} constant over the whole image domain. If there are no strong scattering sources outside the field of view of the camera that scatter light back into the lens, the scattering can be modeled as

$$I_{lcs} = s \cdot \bar{I}_{lci}, \quad (7.2)$$

with s being the scattering parameter of the lens and \bar{I}_{lci} the mean incident light without the scattered part. Accordingly, the unscattered signal can be calculated as

$$I_{lci} = I_{lc} - s \cdot \bar{I}_{lci}. \quad (7.3)$$

I_{lci} is the value of interest, while I_{lc} can be obtained with the calibration from chap. 6. Locally corrected intensities can not be obtained without an estimate of s but the mean value of the corrected data \bar{I}_{lci} can be calculated from Eq. 7.3 with known s :

$$\begin{aligned} \frac{1}{n} \sum_{\Omega} I_{lci} &= \frac{1}{n} \sum_{\Omega} (I_{lc} - s \cdot \bar{I}_{lci}) \\ \Leftrightarrow \bar{I}_{lci} &= \frac{1}{n} \sum_{\Omega} I_{lc} - s \cdot \bar{I}_{lci} \\ \Leftrightarrow \bar{I}_{lci}(1 + s) &= \bar{I}_{lc} \\ \Leftrightarrow \bar{I}_{lci} &= \frac{\bar{I}_{lc}}{1 + s}. \end{aligned} \quad (7.4)$$

With Eq. 7.4 and Eq. 7.3 the sharp image without scattering can be calculated from the calibrated data I_{lc} data with only the scattering parameter s unknown:

$$I_{lci} = I_{lc} - s \cdot \frac{\bar{I}_{lc}}{1 + s}. \quad (7.5)$$

s has to be determined separately. A possible method is suggested in the following section.

7.2 Measuring the Scattering Parameter

What remains to be done is a determination of the scattering coefficient s . To do so, a scene is set up with a target in the background and a white board in the foreground, covering about half of the frame (cf. fig. 7.2) as a large scattering object. The room is darkened and the whole area around the scene is covered in black cloth, to avoid changing ambient light and invisible scattering sources.

If there is a scene with a bright object between the camera and the rest of the scene, the scattering effects can be observed quite well. If the bright object is covered with a dark, non-reflective tissue for a second recording, the scattering can be determined from the two datasets by comparing the identical parts of the scene:

$$\begin{aligned} I_{lci,1} &\stackrel{!}{=} I_{lci,2} \\ \Leftrightarrow I_{lc,1} - s \cdot \bar{I}_{lci,1} &= I_{lc,2} - s \cdot \bar{I}_{lci,2} \\ \Leftrightarrow I_{lc,1} - I_{lc,2} &= s \cdot \bar{I}_{lci,1} - s \cdot \bar{I}_{lci,2} \\ \Leftrightarrow s &= \frac{I_{lc,1} - I_{lc,2}}{\bar{I}_{lci,1} - \bar{I}_{lci,2}} \\ \Leftrightarrow s &= \frac{I_{lc,1} - I_{lc,2}}{\bar{I}_{lci,1} - \bar{I}_{lci,2}} \end{aligned} \quad (7.6)$$

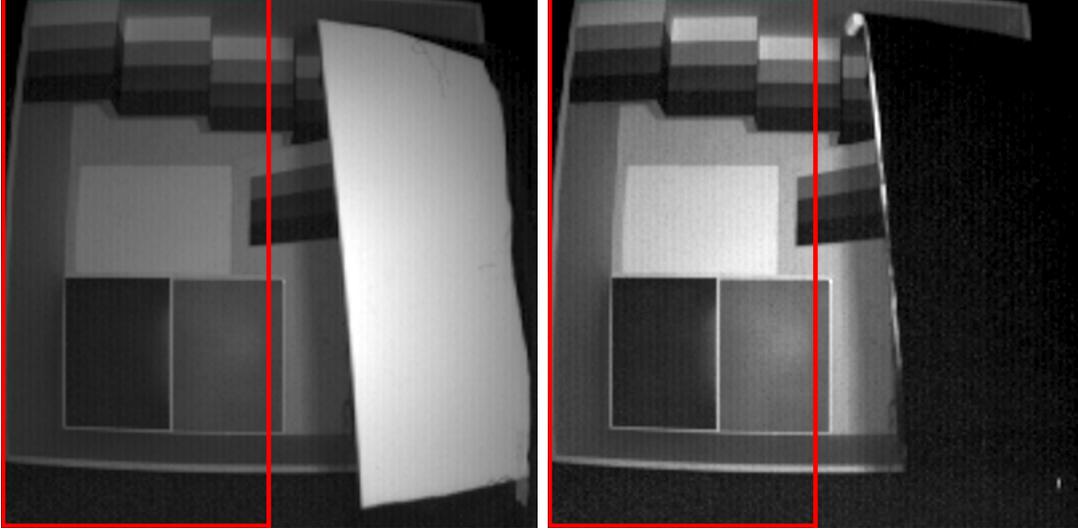


Figure 7.2: Scene to determine the scattering parameter s . The large white scattering object (left image) is covered with a black cloth (right image), while the rest of the scene is left unchanged. From the differences in the left half of the image the scattering parameter can be calculated. The white board is positioned at an angle to avoid overexposure by direct specular reflections of the light sources. Previously published in [46].

Since homogeneous scattering is assumed for the model, Eq. 7.6 can be averaged over the whole identical part of the image (left half in fig. 7.2), denoted as Ω' :

$$\begin{aligned}
 s &= \frac{\frac{1}{n'} \sum_{\Omega'} (I_{lc,1} - I_{lc,2})}{\bar{I}_{lci,1} - \bar{I}_{lci,2}} \\
 \Leftrightarrow s &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lci,1} - \bar{I}_{lci,2}}. \tag{7.7}
 \end{aligned}$$

And with Eq. 7.6, the scattering coefficient s can be calculated directly from the two calibrated measurements in fig. 7.2 as:

$$\begin{aligned}
 s &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\frac{1}{s+1} (\bar{I}_{lc,1} - \bar{I}_{lc,2})} \\
 \Leftrightarrow s - s \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lc,1} - \bar{I}_{lc,2}} &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lc,1} - \bar{I}_{lc,2}} \\
 \Leftrightarrow s \left(1 - \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lc,1} - \bar{I}_{lc,2}} \right) &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lc,1} - \bar{I}_{lc,2}} \\
 \Leftrightarrow s &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{\bar{I}_{lc,1} - \bar{I}_{lc,2}} \cdot \frac{\bar{I}_{lc,1} - \bar{I}_{lc,2}}{(\bar{I}_{lc,1} - \bar{I}_{lc,2}) - (\bar{I}'_{lc,1} - \bar{I}'_{lc,2})} \\
 \Leftrightarrow s &= \frac{\bar{I}'_{lc,1} - \bar{I}'_{lc,2}}{(\bar{I}_{lc,1} - \bar{I}_{lc,2}) - (\bar{I}'_{lc,1} - \bar{I}'_{lc,2})}. \tag{7.8}
 \end{aligned}$$

It is found to be

$$s = 0.017 \pm 0.002, \quad (7.9)$$

which means that about 1.7% of the light recorded by the sensor are diffusely scattered.

7.3 Experiments

To confirm the scattering model the HCI box is now recorded with different scattering sources in the foreground. Bright objects are placed on a tripod close to the camera (cf. fig. 7.3).

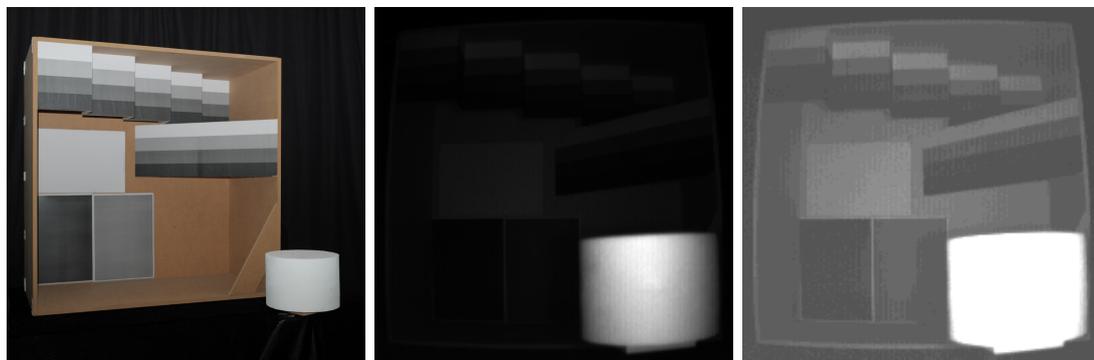


Figure 7.3: Experimental setup to verify the scattering model (left). And camera view. The two images on the right show the same raw data with different linear gray scales to illustrate the strong brightness difference between foreground and background, due to the camera mounted light source. In the following figures the scales will be adjusted nonlinearly to make the whole scene visible.

The camera is set to record at a certain frame rate and integration time, chosen low enough to avoid overexposure. Once a steady temperature is reached, the offset and dark current are measured to perform a calibration according to chap. 6. Immediately after that raw data of the three different scenes are recorded (cf. fig. 7.4). To reduce the Poisson noise 200 frames are recorded for each dataset.

With the offset, the dark current and the previously measured exponents and scattering parameter, the data can be calibrated and corrected for scattering, according to Eq. 6.2 and 7.4. Fig. 7.5 shows the phase shifts calculated from the uncorrected data, from the calibrated data and calculated after the scattering correction in the bottom row. As expected, the scattering object in the foreground causes the back of the scene to appear closer to the camera. The effect is also more obvious in the areas of the scene with bad reflectivity and increases with the size of the scattering object. The results are a very noticeable depth offset of the dark patches in the lower left corner, most obvious in the setup with the cylinder. But also the dark stripes of the stair and ramp structure show a decreased phase shift in the different uncorrected images.

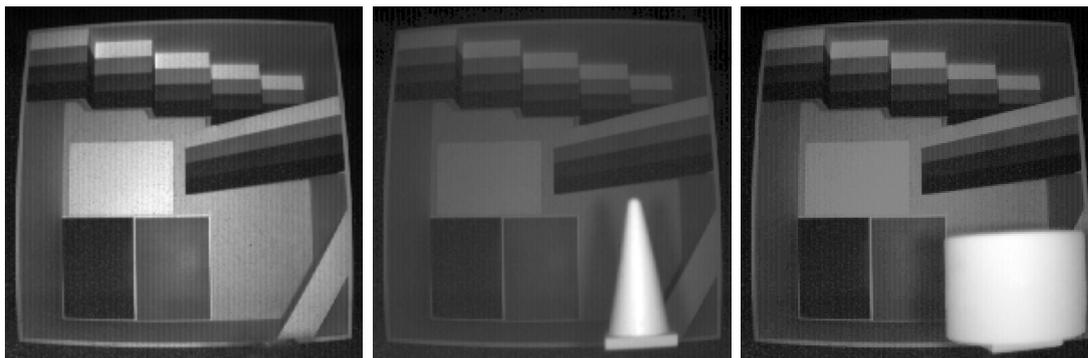


Figure 7.4: The three different variations of the scene to verify the scattering model.

The depth inconsistencies in the corners for example are due to in-scene scattering or multi-reflections, which are not discussed in this thesis. The reader is referred to [16, 8, 30].

Fig. 7.6 shows a crop of the back wall off the box, where the patches with different reflectivity meet. In these images the different phase shift of the darker patches is much more obvious, as well as the qualitatively very good correction by the scattering compensation. The increase in depth or phase shift towards the top left is due to the angle at which the surface is towards the camera, as well as multi-reflection artifacts in the corners and under the stair structure.

Tab. 7.1 gives the mean phase shifts of the crops in fig. 7.6. The offset introduced here by scattering is quite large, shifting the dark area in the cylinder scene about 0.07 rad towards the camera, which corresponds to about 8 cm. For the cone setup the shift is less severe because of the smaller size of the cone.

Data / Scene	Empty	Cone	Cylinder
Uncalibrated	2.512	2.503	2.481
Calibrated	2.510	2.501	2.479
Corrected	2.511	2.512	2.511

Table 7.1: Mean values of the phase shift (in Radian) from the crops in fig. 7.6. The empty scene can be taken as a reference without scattering.

7.4 Discussion

It must be noted, that the correction of the scattering in raw data is impossible without the calibration presented in chap. 6, especially the isolation of the optical signal from the dark current and the signal offset. And the calibration on the other hand would not be possible without access to the raw data. This also offers the advantage of very simple processing steps, compared to a scattering correction from the intensity, amplitude and phase data. The only weakness of the approach

is the possible presence of invisible scattering sources (cf. fig. 7.1) and of course the need to access the raw data.

Fig. 7.7 shows the presence of a scattering object causes very similar behavior also in other camera models.

The in camera scattering model presented here is also the first theoretical explanation and model based correction of an intensity related distance error. The performed experiments support this claim. A thorough calibration and correction of internal scattering will also facilitate further research on in-scene multi-reflections. The internal scattering basically causes an additional contribution to the phase diagram (cf. fig. 5.16). Although the corresponding light is regarded as constant for the entire sensor in the presented model, the different pixel properties can cause a slight variation of the resulting signal for each pixel.

The multi-reflection error ([8, 15, 30]), that was not addressed in this thesis is the largest remaining error source. The research performed in the previous chapters strongly suggests and incorporation of raw data analysis into the solution of these problems as well.

A very interesting value in this regard is the quotient I/A of intensity I (cf. Eq. 2.6) and amplitude A (cf. Eq. 2.7). The theoretical value of this quotient is constant on the whole image domain for single return signals. But with uncalibrated data there is no point in evaluating this because of the high offset (cf. sec. 5.4) in the intensity data. After the calibration and the scattering correction however, the quotient is much more informative (cf. fig.7.8). An increased value is now a good indication for multiple returns, for example at the borders of the white cylinder in the data. There is also a more subtle increase of the values towards the area beneath the stairs or in the top right corner, areas where increased in-scene scattering would be expected.

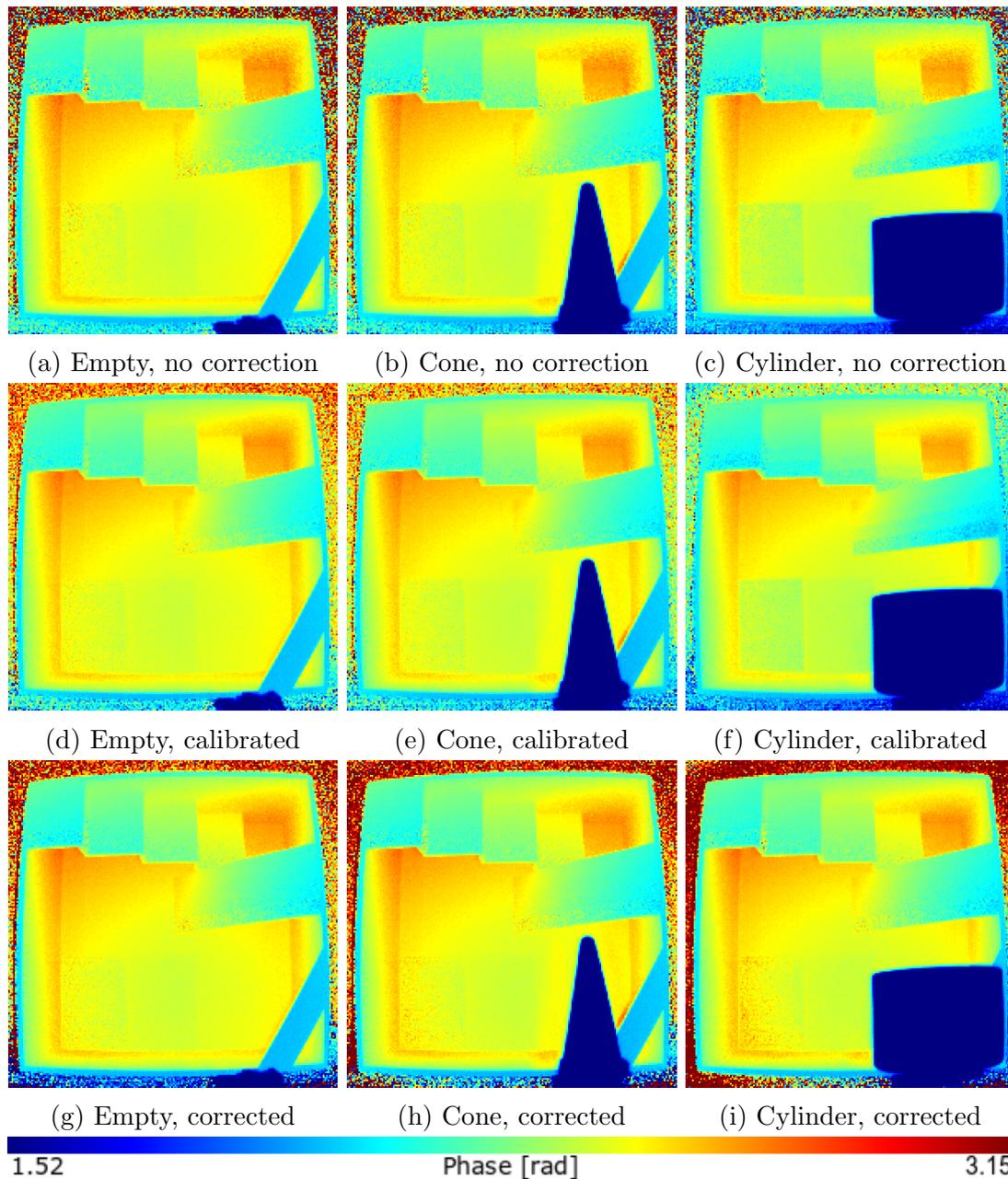


Figure 7.5: Phase image of the scenes in fig. 7.4. The colors are normalized to fig. 7.5a without considering the background, which is very noisy due to the extremely low reflectivity. In this representation of the scene it can be seen already, how the calibration reduces the noise, especially in the dark areas (cf. fig. 7.4 for intensity data) but the dark areas still appear closer to the camera. The deviation also increases with the size of the scattering object. A crop of the large patches in the bottom left area are shown in fig. 7.6.

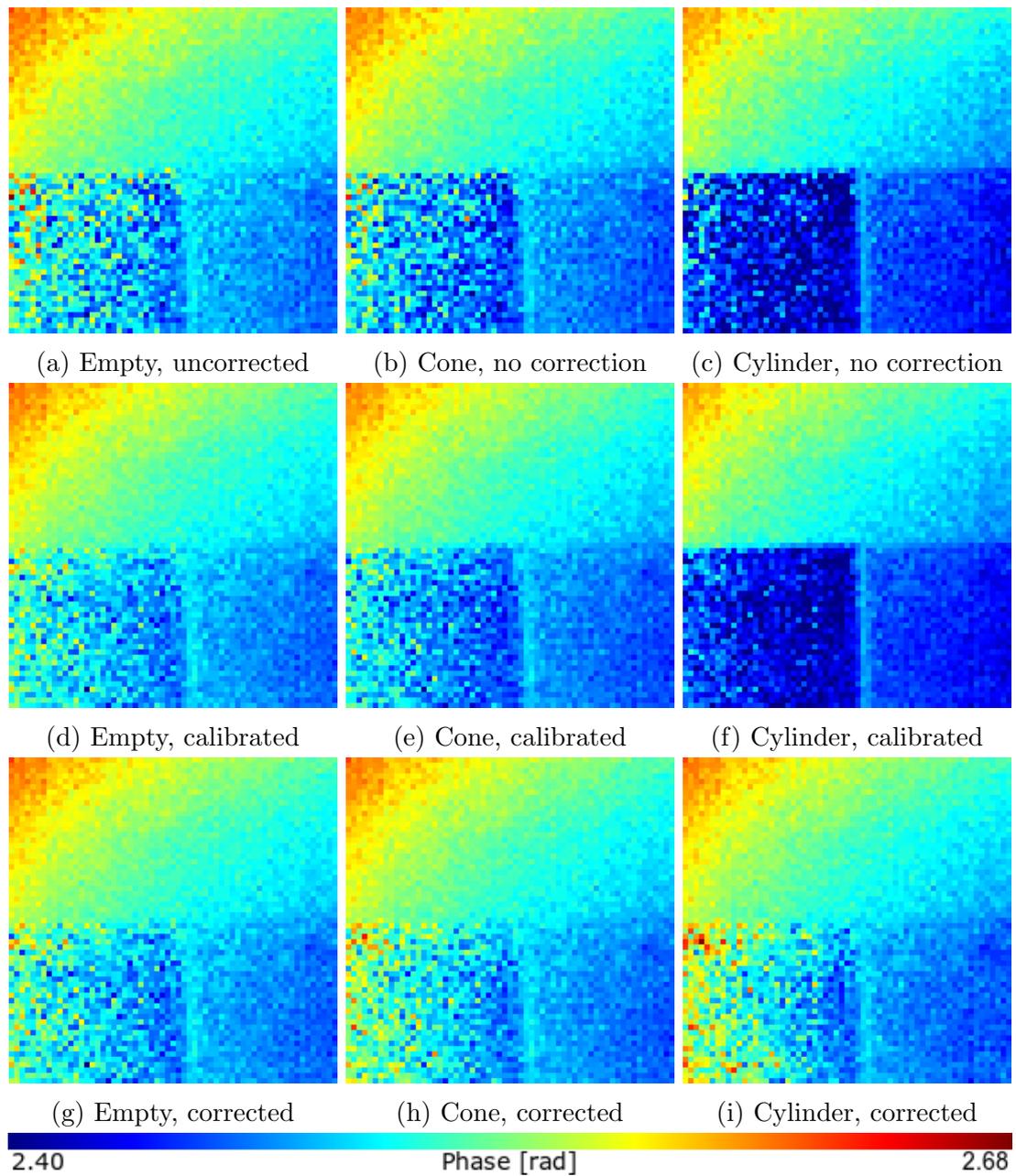


Figure 7.6: Crop from the phase data in the lower left area of fig. 7.5. The color range is normalized to fig. 7.6a. The area displayed is flat but has different intensities, which leads to a strong offset of the phase shift in the uncorrected data, most obvious with the cylinder (cf. fig. 7.6c). The calibration reduces the noise but not the systematic offset of the dark areas. Only the scattering correction effectively removes this systematic error.

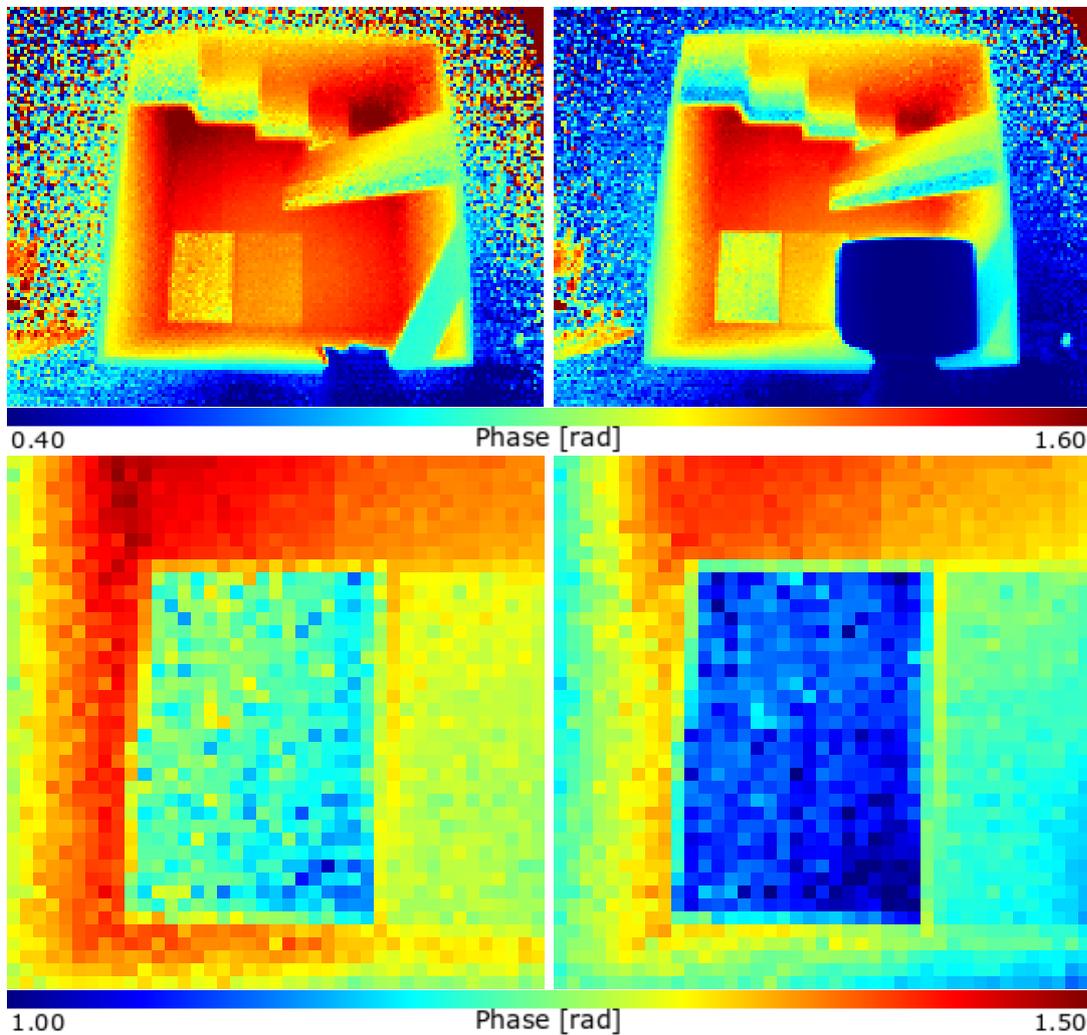


Figure 7.7: For comparison the same scene as in fig. 7.3 recorded with the Argos 3D ToF camera (cf. fig. 1.2b). It shows similar behavior with a scattering object in the foreground of the scene (cf. fig. 7.5c), shifting the whole background towards the camera, especially the dark parts. But like most models the camera does not allow access to the single subframes, making a calibration as proposed in chap. 6, necessary for accurate scattering correction, impossible. The bottom row shows a crop of the lower left area of the box.

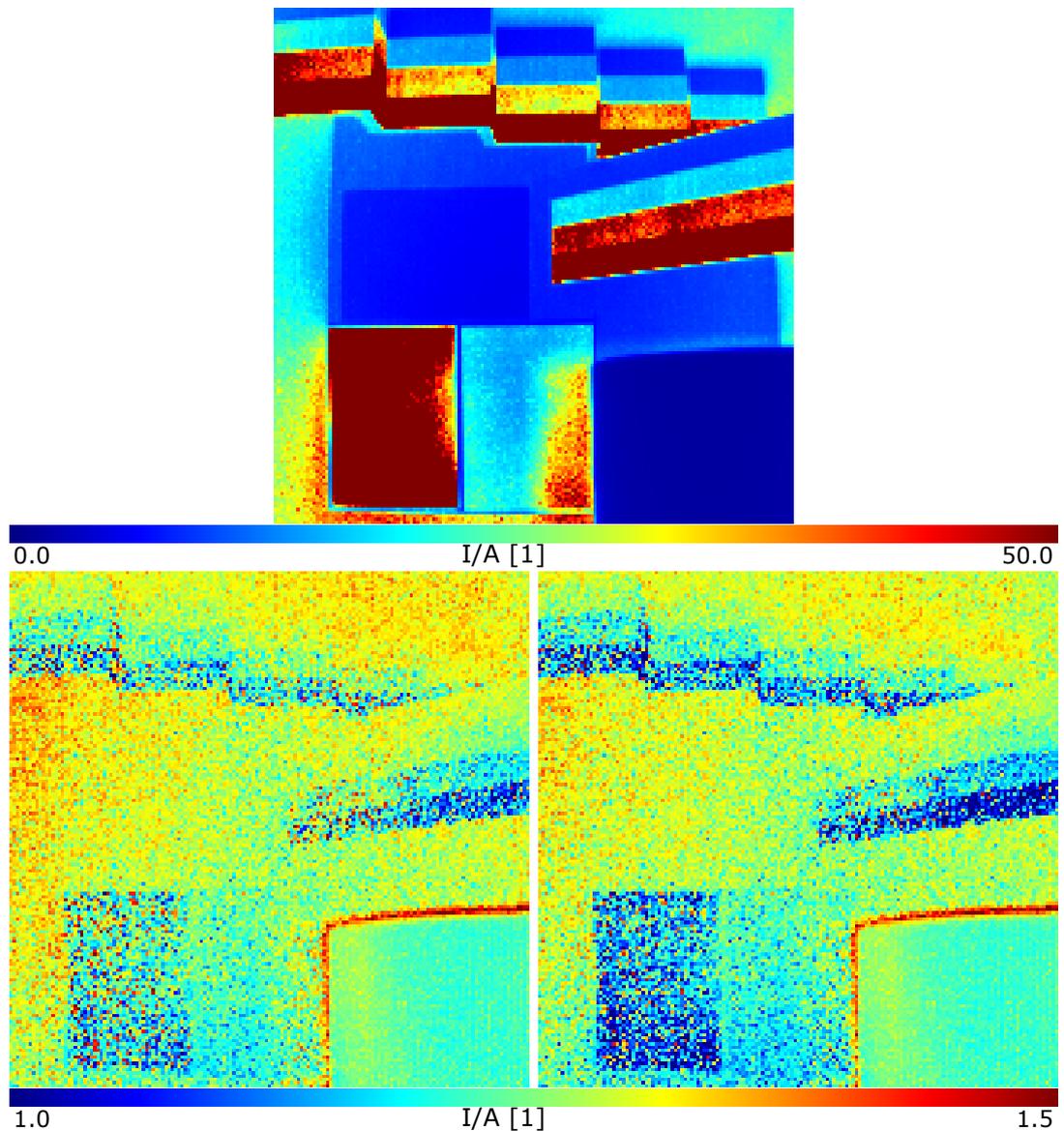


Figure 7.8: Quotients of intensity and amplitude I/A for the uncalibrated (top), calibrated (left) and scattering corrected data (right) of a crop of the respective values from fig. 7.5 c, f and i. The values in the uncalibrated image mostly resemble the change in amplitude. The calibrated and scattering corrected quotients return much more reasonable values which increase in areas where mixed signals can be expected.

“Salviati: What a sea we are gradually slipping into without knowing it! With vacua and infinities and indivisibles and instantaneous motions, shall we ever be able, even by means of a thousand discussions, to reach dry land?”

from “Dialogues Concerning Two New Sciences”
by Galileo Galilei

8

Conclusion

In the presented thesis a couple of new methods to enhance the depth images of ToF cameras were introduced.

An adaptation of image processing algorithms for image smoothing and edge detection was provided, tailored for the special properties and the additional information available from ToF cameras.

With a thorough analysis of the sensor raw data all previously unknown sources of systematic errors could be identified and corrected.

8.1 Summary

After a quick introduction of the basic theory for continuous wave ToF cameras in chap. 2, the performance of the bilateral filter for edge preserving smoothing of the depth data was analyzed in chap. 3. It was shown how this popular method for smoothing ToF depth data struggles with the strong variation of the noise level. An incorporation of the intensity data for the calculation of the local kernels improved the performance, especially in areas of very strong noise. But there remained some configurations of data where the filter caused a smoothing of depth edges due to the inconsistent causality of depth and intensity edges, because both discontinuities can sometimes appear without the other.

To improve the handling of depth discontinuities, edge detection algorithms were investigated in chap. 4. The common method by means of the structure tensor exhibited bad results in noisy areas, just like the bilateral filter before. Also here a combined approach incorporating the intensity data greatly improved the results but also added unwanted intensity edges.

This led to the incorporation of the edge detection algorithm by Canny. Using the output of the structure tensor applied to either the intensity or the depth data as an input, Canny’s algorithm returned a binary map with edge lines only one

pixel wide. The Canny part was extended with a couple of additional parameters to make it more robust to the noise in ToF data.

With separate edge maps from the intensity and depth data, the geometric information of the light sources and camera location as well as the depth measured with the ToF camera, the different intermediate results could be fused into one final edge image. The introduced process detects shadows cast by the light sources and incorporates them as additional information into the fusion process. It is the first approach for edge detection specifically tailored for ToF depth data and gives much better results than the standard algorithms on their own.

The goal of the edge detection was to provide reliable information about discontinuities for adaptive smoothing algorithms. But any smoothing operation is just forming a local average and can therefore only correct noise but no systematic errors in the data.

To gain a better understanding of the errors, the raw data was thoroughly analyzed in chap. 5. A heat model was introduced and parameterized to decouple the temperature dependency of the dark current and the integration time. This allowed the measurement of the non-linearity of the sensor with a dark signal recording.

Further analysis revealed offsets in the dark signal of consecutive subframes, recorded by the same pixel and tap. The variation of these offsets with respect to integration time and frame rate, which both influence the sensor temperature, is the first reasonable explanation for the temperature dependency of the measured depth values. It also suggests that the integration time dependency is just a secondary temperature effect.

The new insight led to the proposal of a dark signal based calibration method in chap. 6. It is scene independent and can be performed within less than a minute, making it much more practical than any of the previously suggested methods. It significantly reduces the depth noise, especially for dark areas in the frame by more than 30% (cf. tab. 6.3). It is not a complete calibration though, as the deviations of the light source modulation is not yet incorporated.

Although the pixel based depth variation that is fixed by the calibration is intensity dependent, the so called intensity related distance error that shifts darker areas in total is not affected by it. But the calibration enabled the measurement of a scattering parameter for the camera in chap. 7. The scattering model presented is different from previous publications, as the assumption is a homogeneous diffuse spread of the scattered light due to various reasons. Prior work assumed only multiple reflections between the sensor and the lens as a source, which should be strongly localized.

The measurements suggest that in-camera light scattering actually is the source of the intensity related distance error. Despite the simple model, which can be considered a first order approximation of the scattering properties of the optics, the compensation applied directly to the raw data gave very good results and removed the scattering effects, i.e. the intensity dependent error, almost completely (cf. tab. 7.1).

8.2 Outlook

The new insight into the raw data and the provided new explanations and solutions for the systematic errors can not only increase the accuracy of the measurements, but also greatly facilitate the research on other challenging topics in the ToF imaging field.

The experiments during this thesis suggest a raw data calibration should be performed for any kind of measurement where the accuracy and consistency is of importance. But that would first of all require access to the camera raw data, which is not possible with most models.

The problem of multiple returns for example is caused by in-scene scattering or mixed pixels at depth edges. A compensation requires a decomposition of the signal into the different parts, according to the scene geometry. The decomposition will be a lot easier and more accurate if the additional components due to the subframe offset and the in-camera scattering have already been subtracted, not to mention the non-linearity that differs even among the two taps of a single pixel.

Manufacturers try to solve the problem of low amplitude and intensity areas especially in the background of a scene with new cameras that record at multiple integration times, like the Argos 3D (cf. fig. 1.2b), which reduces the statistical error of the single pixels. The investigation of the subframe offsets presented here suggests it will also increase the temperature variations and induce new systematic deviations.

With a physical shutter in the cameras the dark signal measurements could be greatly facilitated. It could even enable an automatic performance of the calibration suggested here. With detailed information about the sensor design and one or more internal temperature sensors maybe even a dynamic modeling of the temperature dependency is possible.

With the sources of all the systematic errors now revealed, it will be very interesting to see where Time-of-Flight cameras will take us in the future.

A

Deductions

A.1 Intensity, Amplitude and Phase

The minimum least squares for Intensity, Amplitude and Phase of the residual

$$\begin{aligned} r(I, A, \phi) = \min_{I, A, \phi} \sum_{n=0}^{N-1} (I(\theta_n) - I_n)^2 = \\ \min_{I, A, \phi} \sum_{n=0}^{N-1} ((I + A \cos(\phi + \theta_n)) - I_n)^2 \end{aligned} \tag{A.1}$$

is found with the derivative to the respective variable. This gives the solution for the Intensity I :

$$\begin{aligned}
& \frac{\partial}{\partial I} r(I, A, \phi) = 0 \\
& \Leftrightarrow \frac{\partial}{\partial I} \sum_{n=0}^{N-1} ((I + A \cos(\phi + \theta_n)) - I_n)^2 = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} 2((I + A \cos(\phi + \theta_n)) - I_n) = 0 \\
& \Leftrightarrow 4I - \sum_{n=0}^{N-1} I_n + A(\cos(\phi) + \cos(\phi + \pi) + \cos(\phi + \pi/2) + \cos(\phi + 3\pi/2)) = 0 \\
& \Leftrightarrow 4I = \sum_{n=0}^{N-1} I_n - A(\cos(\phi) - \cos(\phi) + \cos(\phi + \phi/2) - \cos(\phi + \phi/2)) \\
& \Leftrightarrow I = \frac{1}{4} \sum_{n=0}^{N-1} I_n \\
& \Leftrightarrow I = \frac{1}{4}(I_0 + I_1 + I_2 + I_3).
\end{aligned} \tag{A.2}$$

The derivative with respect to ϕ :

$$\begin{aligned}
& \frac{\partial}{\partial \phi} r(I, A, \phi) = 0 \\
& \Leftrightarrow \frac{\partial}{\partial \phi} \sum_{n=0}^{N-1} ((I + A \cos(\phi + \theta_n)) - I_n)^2 = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} 2((I + A \cos(\phi + \theta_n)) - I_n)(-A \sin(\phi + \theta_n)) = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} 2(AI_n \sin(\phi + \theta_n) - A^2 \cos(\phi + \theta_n) \sin(\phi + \theta_n) - AI \sin(\phi + \theta_n)) = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} A(I_n \sin(\phi + \theta_n) - A \cos(\phi + \theta_n) \sin(\phi + \theta_n) - I \sin(\phi + \theta_n)) = 0 \\
& \Leftrightarrow A \sum_{n=0}^{N-1} (I_n \sin(\phi + \theta_n) - A \cos(\phi + \theta_n) \sin(\phi + \theta_n)) = 0 \\
& \Leftrightarrow A \sum_{n=0}^{N-1} I_n \sin(\phi + \theta_n) - A^2 \sum_{n=0}^{N-1} \frac{1}{2} \sin(2\phi + 2\theta_n) = 0 \\
& \Leftrightarrow A(\sin(\phi)(I_0 - I_2) + \cos(\phi)(I_1 - I_3)) = 0 \\
& \Leftrightarrow \sin(\phi)(I_0 - I_2) = \cos(\phi)(I_3 - I_1) \\
& \Leftrightarrow \frac{\sin(\phi)}{\cos(\phi)} = \frac{I_3 - I_1}{I_0 - I_2} \\
& \Leftrightarrow \tan(\phi) = \frac{I_3 - I_1}{I_0 - I_2}.
\end{aligned} \tag{A.3}$$

A can be eliminated from the equation, because $A = 0$ is not a relevant solution.

The straight forward approach would now be to use the $\arctan()$ to calculate ϕ . However, this would return only values in the interval $[-\pi/2; \pi/2]$, while the actual values should be in a range of 2π . Using the $\arctan2()$ gives the desired value in $[-\pi; \pi]$:

$$\phi = \arctan2((I_3 - I_1), (I_0 - I_2)). \tag{A.4}$$

The derivative with respect to A gives an equation dependent on ϕ :

$$\begin{aligned}
& \frac{\partial}{\partial A} r(I, A, \phi) = 0 \\
& \Leftrightarrow \frac{\partial}{\partial A} \sum_{n=0}^{N-1} ((I + A \cos(\phi + \theta_n)) - I_n)^2 = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} 2((I + A \cos(\phi + \theta_n)) - I_n) \cos(\phi + \theta_n) = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} (I \cos(\phi + \theta_n) + A \cos^2(\phi + \theta_n) - I_n \cos(\phi + \theta_n)) = 0 \\
& \Leftrightarrow \sum_{n=0}^{N-1} (A \cos^2(\phi + \theta_n) - I_n \cos(\phi + \theta_n)) = 0 \tag{A.5} \\
& \Leftrightarrow A \sum_{n=0}^{N-1} \cos^2(\phi + \theta_n) - \sum_{n=0}^{N-1} I_n \cos(\phi + \theta_n) = 0 \\
& \Leftrightarrow A (2 \cos^2(\phi) + 2 \sin^2(\phi)) = \sum_{n=0}^{N-1} I_n \cos(\phi + \theta_n) \\
& \Leftrightarrow 2A = I_0 \cos(\phi) - I_2 \cos(\phi) + I_1 \cos(\phi + \pi/2) + I_3 \cos(\phi + 3\pi/2) \\
& \Leftrightarrow 2A = \cos(\phi)(I_0 - I_2) + \cos(\phi + \pi/2)(I_1 - I_3) \\
& \Leftrightarrow A = \frac{1}{2} (\cos(\phi)(I_0 - I_2) + \sin(\phi)(I_3 - I_1)).
\end{aligned}$$

If this is combined with eq. A.3, the solution for A is found to be:

$$\begin{aligned}
& A = \frac{1}{2} (\cos(\phi)(I_0 - I_2) + \sin(\phi)(I_3 - I_1)) \\
& \Leftrightarrow A = \frac{1}{2} \left(\cos \left(\arctan \left(\frac{I_3 - I_1}{I_0 - I_2} \right) \right) (I_0 - I_2) + \right. \\
& \quad \left. \sin \left(\arctan \left(\frac{I_3 - I_1}{I_0 - I_2} \right) \right) (I_3 - I_1) \right) \\
& \Leftrightarrow A = \frac{1}{2} \left(\frac{I_0 - I_2}{\sqrt{1 + \left(\frac{I_3 - I_1}{I_0 - I_2} \right)^2}} + \frac{\frac{(I_3 - I_1)^2}{I_0 - I_2}}{\sqrt{1 + \left(\frac{I_3 - I_1}{I_0 - I_2} \right)^2}} \right) \tag{A.6} \\
& \Leftrightarrow A = \frac{1}{2} \frac{(I_0 - I_2)^2 + (I_3 - I_1)^2}{\sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}} \\
& \Leftrightarrow A = \frac{1}{2} \sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}.
\end{aligned}$$

A.2 Poisson Noise

As mentioned in sec. 2.3, in [43] and [13] the variances of I, A and ϕ are already deduced with constant variances σ^2 for the subframes I_n . Because of this and some

inconsistencies in the results the process is repeated here with variable variances:

$$\begin{aligned}
 \text{Var}(\vec{f}(\vec{I})) &= J(\vec{f})\text{Var}(\vec{I})J(\vec{f})^T = \\
 &\begin{pmatrix} \frac{\partial I}{\partial I_0} & \frac{\partial I}{\partial I_1} & \frac{\partial I}{\partial I_2} & \frac{\partial I}{\partial I_3} \\ \frac{\partial A}{\partial I_0} & \frac{\partial A}{\partial I_1} & \frac{\partial A}{\partial I_2} & \frac{\partial A}{\partial I_3} \\ \frac{\partial \phi}{\partial I_0} & \frac{\partial \phi}{\partial I_1} & \frac{\partial \phi}{\partial I_2} & \frac{\partial \phi}{\partial I_3} \end{pmatrix} \begin{pmatrix} \sigma_{I_0}^2 & 0 & 0 & 0 \\ 0 & \sigma_{I_1}^2 & 0 & 0 \\ 0 & 0 & \sigma_{I_2}^2 & 0 \\ 0 & 0 & 0 & \sigma_{I_3}^2 \end{pmatrix} \begin{pmatrix} \frac{\partial I}{\partial I_0} & \frac{\partial A}{\partial I_0} & \frac{\partial \phi}{\partial I_0} \\ \frac{\partial I}{\partial I_1} & \frac{\partial A}{\partial I_1} & \frac{\partial \phi}{\partial I_1} \\ \frac{\partial I}{\partial I_2} & \frac{\partial A}{\partial I_2} & \frac{\partial \phi}{\partial I_2} \\ \frac{\partial I}{\partial I_3} & \frac{\partial A}{\partial I_3} & \frac{\partial \phi}{\partial I_3} \end{pmatrix} = \\
 &\begin{pmatrix} \sum_n \frac{\partial I}{\partial I_n} \sigma_{I_n}^2 \frac{\partial I}{\partial I_n} & & & \\ & \sum_n \frac{\partial A}{\partial I_n} \sigma_{I_n}^2 \frac{\partial A}{\partial I_n} & & \\ & & \dots & \\ & & & \sum_n \frac{\partial \phi}{\partial I_n} \sigma_{I_n}^2 \frac{\partial \phi}{\partial I_n} \end{pmatrix}. \tag{A.7}
 \end{aligned}$$

Calculating σ_I^2 is quite simple:

$$\begin{aligned}
 \sigma_I^2 &= \left(\frac{\partial I}{\partial I_0}\right)^2 I_0 + \left(\frac{\partial I}{\partial I_1}\right)^2 I_1 + \left(\frac{\partial I}{\partial I_2}\right)^2 I_2 + \left(\frac{\partial I}{\partial I_3}\right)^2 I_3 = \\
 &\quad \left(\frac{1}{4}\right)^2 I_0 + \left(\frac{1}{4}\right)^2 I_1 + \left(\frac{1}{4}\right)^2 I_2 + \left(\frac{1}{4}\right)^2 I_3 = \\
 &\quad \frac{1}{16} (I_0 + I_1 + I_2 + I_3) = \frac{I}{4}. \tag{A.8}
 \end{aligned}$$

In [43] the final result is identical, but there is a misleading error before the final step. In [13] it is denoted as $\frac{I}{2}$ because it was confused with σ_A^2 .

This also explains the inconsistency for σ_A^2 in the two publications, which is derived

as:

$$\begin{aligned}
\sigma_A^2 &= \left(\frac{\partial A}{\partial I_0}\right)^2 I_0 + \left(\frac{\partial A}{\partial I_1}\right)^2 I_1 + \left(\frac{\partial A}{\partial I_2}\right)^2 I_2 + \left(\frac{\partial A}{\partial I_3}\right)^2 I_3 = \\
&\left(\frac{(I_0 - I_2)}{2\sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}}\right)^2 I_0 + \left(\frac{(I_1 - I_3)}{2\sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}}\right)^2 I_1 + \\
&\left(\frac{(I_2 - I_0)}{2\sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}}\right)^2 I_2 + \left(\frac{(I_3 - I_1)}{2\sqrt{(I_0 - I_2)^2 + (I_3 - I_1)^2}}\right)^2 I_3 = \\
&\frac{(I_0 - I_2)^2 I_0 + (I_1 - I_3)^2 I_1 + (I_2 - I_0)^2 I_2 + (I_3 - I_1)^2 I_3}{4((I_0 - I_2)^2 + (I_3 - I_1)^2)} = \\
&\frac{(I_0 - I_2)^2(I_0 + I_2) + (I_1 - I_3)^2(I_1 + I_3)}{4((I_0 - I_2)^2 + (I_3 - I_1)^2)} = \\
&\frac{A^2(\cos(\phi + \theta_0) - \cos(\phi + \theta_2))^2(2I + A(\cos(\phi + \theta_0) + \cos(\phi + \theta_2)))}{4(A^2(\cos(\phi + \theta_0) - \cos(\phi + \theta_2))^2 + A^2(\cos(\phi + \theta_3) - \cos(\phi + \theta_1))^2)} + \\
&\frac{A^2(\cos(\phi + \theta_1) - \cos(\phi + \theta_3))^2(2I + A(\cos(\phi + \theta_1) + \cos(\phi + \theta_3)))}{4(A^2(\cos(\phi + \theta_0) - \cos(\phi + \theta_2))^2 + A^2(\cos(\phi + \theta_3) - \cos(\phi + \theta_1))^2)} = \\
&\frac{(\cos(\phi + \theta_0) + \cos(\phi + \theta_0))^2(2I + A(\cos(\phi + \theta_0) - \cos(\phi + \theta_0)))}{4((\cos(\phi + \theta_0) + \cos(\phi + \theta_0))^2 + (\cos(\phi + \theta_3) + \cos(\phi + \theta_3))^2)} + \\
&\frac{(\cos(\phi + \theta_1) + \cos(\phi + \theta_1))^2(2I + A(\cos(\phi + \theta_1) - \cos(\phi + \theta_1)))}{4((\cos(\phi + \theta_0) + \cos(\phi + \theta_0))^2 + (\cos(\phi + \theta_3) + \cos(\phi + \theta_3))^2)} = \\
&\frac{(2\cos(\phi + \theta_0))^2(2I) + (2\cos(\phi + \theta_1))^2(2I)}{4((2\cos(\phi + \theta_0))^2 + (2\cos(\phi + \theta_3))^2)} = \\
&\frac{(2I)((\cos(\phi + \theta_0))^2 + (\cos(\phi + \theta_1))^2)}{4((\cos(\phi + \theta_0))^2 + (\cos(\phi + \theta_3))^2)} = \\
&\frac{I((\cos(\phi + \theta_0))^2 + (-\sin(\phi + \theta_0))^2)}{2((\cos(\phi + \theta_0))^2 + (\sin(\phi + \theta_0))^2)} = \frac{I}{2}. \tag{A.9}
\end{aligned}$$

The variance of ϕ , probably the most interesting one, was consistend from the

beginning. What follows is the deduction with variable variances:

$$\begin{aligned}
\sigma_\phi^2 &= \left(\frac{\partial\phi}{\partial I_0}\right)^2 I_0 + \left(\frac{\partial\phi}{\partial I_1}\right)^2 I_1 + \left(\frac{\partial\phi}{\partial I_2}\right)^2 I_2 + \left(\frac{\partial\phi}{\partial I_3}\right)^2 I_3 = \\
&\quad \left(\frac{\partial \operatorname{atan}\left(\frac{I_3-I_1}{I_0-I_2}\right)}{\partial I_0}\right)^2 I_0 + \left(\frac{\partial \operatorname{atan}\left(\frac{I_3-I_1}{I_0-I_2}\right)}{\partial I_1}\right)^2 I_1 + \\
&\quad \left(\frac{\partial \operatorname{atan}\left(\frac{I_3-I_1}{I_0-I_2}\right)}{\partial I_2}\right)^2 I_2 + \left(\frac{\partial \operatorname{atan}\left(\frac{I_3-I_1}{I_0-I_2}\right)}{\partial I_3}\right)^2 I_3 = \\
&\quad \left(\frac{\left(\frac{I_3-I_1}{(I_0-I_2)^2}\right)^2}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 I_0 + \left(\frac{\left(\frac{-1}{(I_0-I_2)}\right)^2}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 I_1 + \\
&\quad \left(\frac{\left(\frac{-\frac{I_3-I_1}{(I_0-I_2)^2}}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 I_2 + \left(\frac{\frac{1}{(I_0-I_2)^2}}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 I_3 = \\
&\quad \left(\frac{\left(\frac{I_3-I_1}{(I_0-I_2)^2}\right)^2}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 (I_0 + I_2) + \left(\frac{\frac{1}{(I_0-I_2)^2}}{1 + \left(\frac{I_3-I_1}{I_0-I_2}\right)^2}\right)^2 (I_3 + I_1) = \\
&\quad \left(\frac{(I_3 - I_1)}{((I_3 - I_1)^2 + (I_0 - I_2)^2)}\right)^2 (I_0 + I_2) + \left(\frac{(I_0 - I_2)}{((I_3 - I_1)^2 + (I_0 - I_2)^2)}\right)^2 (I_3 + I_1) = \\
&\quad \left(\frac{\cos(\phi + \theta_1)}{A((\cos(\phi + \theta_1))^2 + (\cos(\phi + \theta_0))^2)}\right)^2 (2I + A(\cos(\phi + \theta_0) + \cos(\phi + \theta_2))) + \\
&\quad \left(\frac{\cos(\phi + \theta_0)}{2A((\cos(\phi + \theta_1))^2 + (\cos(\phi + \theta_0))^2)}\right)^2 (2I + A(\cos(\phi + \theta_1) + \cos(\phi + \theta_3))) = \\
&\quad \left(\frac{\cos(\phi + \theta_1)}{2A}\right)^2 (2I) + \left(\frac{\cos(\phi + \theta_0)}{2A}\right)^2 (2I) = \\
&\quad \frac{I}{2A^2} ((-\sin(\phi + \theta_0))^2 + (\cos(\phi + \theta_0))^2) = \frac{I}{2A^2}.
\end{aligned}
\tag{A.10}$$

List of Figures

1.1	Sketch of an experiment by Fizeau in 1849 to determine the speed of light.	11
1.2	A selection of current Time-of-Flight cameras.	12
1.3	Color coded depth image.	13
1.4	3D target box with interchangeable 3D structures.	15
2.1	Schematic of the data recording process of a continuous wave ToF camera.	19
3.1	Depth data filtered with standard Gaussian and bilateral filter of the same width.	23
3.2	Example for the shape of a bilateral filter kernel.	24
3.3	Exemplary results of a joint bilateral filter.	26
3.4	Problems of a joint bilateral filter.	26
3.5	Exemplary results of the adaptive bilateral filter.	28
4.1	Illustration of the frequency doubling in a pointwise multiplication of a signal.	31
4.2	Grey-value representation of an hour-glass shaped filter kernel	31
4.3	Results of the structure tensor applied to the intensity and depth data of a scene.	32
4.4	Different combinations of the structure tensor applied to depth and intensity data.	33
4.5	Different stages of Canny's algorithm.	34
4.6	The light sources mounted to the left and right of the lens cast strong shadows on the background next to an object.	35
4.7	Schematic of the shadows cast by a depth edge due to the light sources mounted left and right of the camera.	36
4.8	Example scene for detected shadow edges.	37
4.9	Overview of the different steps in the edge detection and fusion procedure.	38
4.10	Intensity data with marked intensity edges and detected shadows and an overlay on the depth data.	39
5.1	Actual intensity modulation of the PMD CC3 LED arrays, measured with a photo diode.	42
5.2	Frequency spectrum of the CC3 LED modulation, showing the higher harmonics in the signal.	43

5.3	LED-signal integrated over 1/2 period.	44
5.4	Theoretical wiggling error simulated with the measured LED modulation.	44
5.5	Change of intensity, amplitude and depth during warm-up of the camera.	45
5.6	Schematic of the camera for the heat model.	46
5.7	3D Plot of the temperature data from tab. 5.1 with fit of the heat model from eq. 5.4.	48
5.8	Dark signal of the different taps during camera warm-up.	49
5.9	Dark signal of the CC3, tap A, first subframe, averaged over 100 frames.	50
5.10	Dark current of a single pixel, averaged over 100 frames at two different camera temperatures.	51
5.11	Decrease of the dark current at $t_{int} = 100 \mu s$ for the measurements in fig. 5.8.	53
5.12	Dark signal difference of the subframes in comparison to the first subframe.	54
5.13	Single subframe and intensity image of a textured cardboard mounted to a motor to determine the time between subframes.	55
5.14	Dark signal difference of the subframes compared to the first subframe at different integration times and frame rates.	55
5.15	Modeled subframe intensities.	57
5.16	Phase diagram of combined optical signal and signal off subframe offsets.	58
5.17	Phase with and without the offset variations.	59
5.18	Phase error due to the non-linearity and additionally the subframe offset.	60
6.1	Raw data and phase before and after the calibration.	63
6.2	Crop of the phase images from fig. 6.1.	64
6.3	Small crop of the flat dark patch in the lower left corner of the phase images from fig. 6.1.	65
7.1	Diffuse scattering in a lens.	68
7.2	Scene to determine the scattering parameter s	70
7.3	Experimental setup to verify the scattering model.	71
7.4	The three different variations of the scene to verify the scattering model.	72
7.5	Phase images of the scenes in fig. 7.4.	74
7.6	Crop from the phase data of fig. 7.5.	75
7.7	For comparison the same scene as in fig. 7.3 recorded with the Argos 3D ToF camera.	76
7.8	Quotients of intensity and amplitude I/A for the uncalibrated, calibrated and scattering corrected data.	77

List of Tables

5.1	Recorded data to fit the different contributions to the thermal power of the CC3.	47
5.2	Fit results of the data in fig. 5.10. While the offset of the signal (c) and the dark current (a) increases significantly, the exponent (b) stays constant (a in $1/\mu\text{s}$).	52
5.3	Overview of how the subframes combined into one frame I_i for the calculation of ϕ as in eq. 2.8 (cf. fig. 2.1).	56
6.1	Statistics of the phase data in fig. 6.1 in [rad]. Interesting to note is the decrease in the variance of the phase by 50%, while the total range of the data actually increases. Also the mean phase changes only slightly.	64
6.2	Statistics of the phase data in fig. 6.2. While the changes are noticeable in the images to the naked eye, there is hardly a change to the statistics here because the highly noisy areas around the box are cut off.	65
6.3	Statistics of the phase data in fig. 6.3.	66
7.1	Mean values of the phase shift (in Radian) from the crops in fig. 7.6. The empty scene can be taken as a reference without scattering. . .	72

Bibliography

- [1] M. Albrecht. Untersuchung von photogate-pmd-sensoren hinsichtlich qualifizierender charakterisierungsparameter und-methoden. 2007.
- [2] V. Aurich and J. Weule. Non-linear Gaussian filters performing edge preserving diffusion. In *Proceed. 17. DAGM-Symposium*, 1995.
- [3] P. Boulanger, F. Blais, and P. Cohen. Detection of depth and orientation discontinuities in range images using mathematical morphology. In *Pattern Recognition, Proceedings, 10th International Conference on*, 1990.
- [4] B. Büttgen, M.-A. El Mechat, F. Lustenberger, and P. Seitz. Pseudonoise optical modulation for real-time 3-d imaging with minimum interference. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 54(10):2109–2119, Oct 2007.
- [5] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986.
- [6] D. Chan, H. Buisman, C. Theobalt, and S. Thrun. A noise-aware filter for real-time depth upsampling. In *Proc. of ECCV Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, pages 1–12, 2008.
- [7] S. Coleman, B. Scotney, and S. Suganthan. Edge detecting for range data using laplacian operators. *IEEE Trans. Image Process.*, 2010.
- [8] A. A. Dorrington, J. P. Godbaz, M. J. Cree, A. D. Payne, and L. V. Streeter. Separating true range measurements from multi-path and scattering interference in commercial range cameras. 2011.
- [9] EMVA 1288 Standardization Group. Emva standard 1288 - standard for characterization of image sensors and cameras, 2010.
- [10] H. Fizeau. Sur une expérience relative à la vitesse de propagation de la lumière. *Compte rendu de l'Académie des Sciences*, 29:90–92, 1849.
- [11] S. Foix, G. Alenya, and C. Torras. Lock-in time-of-flight (tof) cameras: a survey. *Sensors Journal, IEEE*, 11(9):1917–1926, 2011.
- [12] W. Förstner. A framework for low-level feature extraction. In *ECCV, LNCS*, 1994.

-
- [13] M. Frank, M. Plaue, H. Rapp, U. Köthe, B. Jähne, and F. A. Hamprecht. Theoretical and experimental error analysis of continuous-wave time-of-flight range cameras. *Optical Engineering*, 48:013602, 2009.
- [14] G. Galilei. *Discorsi e Dimostrazioni Matematiche, Intorno à Due Nuove Scienze Attenti Alla Mecanica & i Movimenti Locali*. Elsevirii, 1638.
- [15] J. Godbaz, M. Cree, and A. Dorrington. Understanding and ameliorating non-linear phase and amplitude responses in amcw lidar. *Remote Sensing*, 4(1), 2012.
- [16] S. A. GuOmundsson, H. Aanæs, and R. Larsen. Environmental effects on measurement uncertainties of time-of-flight cameras. In *Signals, Circuits and Systems, 2007. ISSCS 2007. International Symposium on*, volume 1, pages 1–4. IEEE, 2007.
- [17] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.
- [18] S. Hussmann, P. Huhn, and A. Hermanski. Systematic distance deviation error compensation for a tof-camera in the close-up range. In *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, pages 1546–1550. IEEE, 2012.
- [19] B. Jähne. *Digital Image Processing*. Springer, Berlin, 6 edition, 2005.
- [20] X. Jiang and H. Bunke. Edge detection in range images based on scan line approximation. *Computer Vision and Image Understanding*, 1999.
- [21] T. Kahlmann and H. Ingensand. Calibration and development for increased accuracy of 3d range imaging cameras. *Journal of Applied Geodesy*, 2(1):1–11, 2008.
- [22] T. Kahlmann, F. Remondino, and H. Ingensand. Calibration for increased accuracy of the range imaging camera swissrangertm. *Image Engineering and Vision Metrology (IEVM)*, 36(3):136–141, 2006.
- [23] W. Karel, S. Ghuffar, and N. Pfeifer. Modelling and compensating internal light scattering in time of flight range cameras. *The Photogrammetric Record*, 27(138):155–174, 2012.
- [24] T. Kavli, T. Kirkhus, J. T. Thielemann, and B. Jagielski. Modelling and compensating measurement errors caused by scattering in time-of-flight cameras. In *Optical Engineering+ Applications*, pages 706604–706604. International Society for Optics and Photonics, 2008.
- [25] S. Kindermann, S. Osher, and P. Jones. Deblurring and denoising of images by nonlocal functionals. *Multiscale Model. Simul.*, 4(4):1091–1115 (electronic), 2005.

- [26] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. In *ACM SIGGRAPH 2007 papers*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.
- [27] U. Köthe. Edge and junction detection with an improved structure tensor. In *Proc. of the 25th DAGM Symposium on Pattern Recognition*, LNCS. DAGM, 2003.
- [28] R. Krishnan. Scattering of light in optical glasses. In *Proceedings of the Indian Academy of Sciences-Section A*, volume 3, pages 211–220. Springer, 1936.
- [29] R. Lange. *3D Time-of-Flight Distance Measurement with Custom Solid-State Image Sensors in CMOS/CCD-Technology*. PhD thesis, Department of Electrical Engineering and Computer Science at University of Siegen, 2000.
- [30] D. Lefloch, R. Nair, F. Lenzen, H. Schäfer, L. Streeter, M. J. Cree, R. Koch, and A. Kolb. Technical foundation and calibration methods for time-of-flight cameras. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, pages 3–24. Springer, 2013.
- [31] A. Lejeune, S. Piérard, M. Van Droogenbroeck, and J. Verly. A new jump edge detection method for 3D cameras. In *International Conference on 3D Imaging (IC3D)*, 2011.
- [32] F. Lenzen, H. Schäfer, and C. Garbe. Denoising time-of-flight data with adaptive total variation. *Advances in Visual Computing*, pages 337–346, 2011.
- [33] M. Lindner. *Calibration and Real-Time Processing of Time-of-Flight Range Data*. PhD thesis, CG, Fachbereich Elektrotechnik und Informatik, Univ. Siegen, 2010.
- [34] M. Lindner and A. Kolb. Lateral and depth calibration of pmd-distance sensors. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel, and T. Malzbender, editors, *ISVC (2)*, volume 4292 of *Lecture Notes in Computer Science*, pages 524–533. Springer, 2006.
- [35] M. Lindner and A. Kolb. Calibration of the intensity-related distance error of the pmd tof-camera. In *Optics East*. International Society for Optics and Photonics, 2007.
- [36] S. Matsuda and T. Nitoh. Flare as applied to photographic lenses. *Applied optics*, 11(8):1850–1856, 1972.
- [37] S. N. R. Meister. *On Creating Reference Data for Performance Analysis in Image Processing*. PhD thesis, HCI, Fakultät für Physik und Astronomie, Univ. Heidelberg, 2014.

- [38] S. Mersmann, D. Guerrero, H.-P. Schlemmer, H.-P. Meinzer, and L. Maier-Hein. Effect of active air conditioning in medical intervention rooms on the temperature dependency of time-of-flight distance measurements. In *Bildverarbeitung für die Medizin 2012*, pages 398–403. Springer, 2012.
- [39] S. Mersmann, A. Seitel, M. Erz, B. Jähne, F. Nickel, M. Mieth, A. Mehrabi, and L. Maier-Hein. Calibration of time-of-flight cameras for accurate intra-operative surface reconstruction. *Medical physics*, 40(8):082701, 2013.
- [40] J. Mure-Dubois and H. Hügli. Real-time scattering compensation for time-of-flight camera. In *Proceedings of ICVS*, 2007.
- [41] S. Paris and F. Durand. A fast approximation of the bilateral filter using a signal processing approach. Technical report, MIT - CSAIL, 2006.
- [42] B. Parvin and G. Medioni. Adaptive multiscale feature extraction from range data. *Computer Vision, Graphics, and Image Processing*, 1989.
- [43] M. Plaue. Analysis of the pmd imaging system. Technical report, Interdisciplinary Center for Scientific Computing, University of Heidelberg, 2006.
- [44] C. Raman. The scattering of light in amorphous solids. *Journal of the Optical Society of America*, 15(4):185–188, 1927.
- [45] H. Rapp. Experimental and Theoretical Investigation of Correlating TOF-Camera Systems. Diplomarbeit, IWR, Fakultät für Physik und Astronomie, Universität Heidelberg, 2007.
- [46] H. Schäfer, F. Lenzen, and C. S. Garbe. A new model for scattering in time-of-flight cameras. *Submitted*.
- [47] H. Schäfer, F. Lenzen, and C. S. Garbe. Depth and intensity based edge detection in time-of-flight images. In *3DTV-Conference, 2013 International Conference on*, pages 111–118. IEEE, 2013.
- [48] M. Schmidt. *Spatiotemporal Analysis of Range Imagery*. Dissertation, IWR, Fakultät für Physik und Astronomie, University of Heidelberg, 2008.
- [49] M. Schmidt. *Analysis, Modeling and Dynamic Optimization of 3D Time-of-Flight Imaging Systems*. Dissertation, IWR, Fakultät für Physik und Astronomie, Univ. Heidelberg, 7 2011. started 01.07.2008.
- [50] B. Schneider. *Der Photomischdetektor zur schnellen 3D-Vermessung für Sicherheitssysteme und zur Informationsübertragung im Automobil*. PhD thesis, Department of Electrical Engineering and Computer Science, 2003.
- [51] R. Schwarte, H.-G. Heinol, Z. Xu, and K. Hartmann. New active 3d vision system based on rf-modulation interferometry of incoherent light. In *Photonics East'95*, pages 126–134. International Society for Optics and Photonics, 1995.

-
- [52] O. Steiger, J. Felder, and S. Weiss. Calibration of time-of-flight range imaging cameras. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 1968–1971. IEEE, 2008.
- [53] R. Strutt. Scattering of light by solid substances. *Proceedings of the Royal Society of London. Series A*, 95(672):476–479, 1919.
- [54] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India, 1998*.
- [55] wikipedia.org, 2014. <http://en.wikipedia.org/wiki/kinect> rev. July 31st 2014, 00:41 (81,481 bytes).
- [56] Q. Yang, R. Yang, J. Davis, and D. Nistér. Spatial-depth super resolution for range images. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [57] C. Ye and G.-P. Hegde. Robust edge extraction for swissranger sr-3000 range images. In *ICRA*. IEEE, 2009.