Christian Anton Kopf BSc.

Volumetric 3D Reconstruction for the Industrial Inspection of Transparent Materials

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme
Information and Computer Engineering

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl.-Ing. Dr.techn. Thomas Pock

Institute of Computer Graphics and Vision

Advisor

Dr. Mag. Svorad Štolc

Austrian Institute of Technology

Graz, February 2020

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das Textdokument auf TUGRAZonline ist identisch zur vorliegenden Arbeit.

Ort	Datum	Unterschrift

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Place	Date	Signature

Abstract

Computing 3D depth information through stereo methods is a well researched topic of computer vision and has lead to countless variations and approaches over the years. However the applicability to transparent objects is still considered as one of the few unresolved disciplines to this date. The conventional principle of computing depth cues from finding correspondences in two or more images leads to poor results due to ambiguities caused by transparent materials. In this thesis we thus explore an alternative approach which is based on the local orientation analysis of light field data in the domain of epipolar plane images. In this context we discuss the principles of structure tensor approaches and their intrinsic properties. Starting with the basic case for opaque Surfaces we will subsequently explain how tensor models can be extended to higher order, such that depth ambiguities from transparent surfaces can be resolved. Furthermore we show our findings and improvements to the method. Based on a voxel volume projection approach we present two variants of a refinement method which allows us to obtain improved results for scenes including transparent objects. Since this work is part of a project which originates from an industrial context, we assume certain conditions regarding the acquisition setup and the data throughout the course of this thesis. However the presented method is equally applicable to a more generalized setting.

Kurzfassung

Die Berechnung von 3D-Tiefeninformationen mit Hilfe von Stereo-Methoden ist ein gut erforschtes Themengebiet der Computer Vision und hat im Laufe der Jahre zu unzähligen Variationen und Ansätzen geführt. Die Anwendbarkeit auf transparente Objekte gilt jedoch bis heute als eine der wenigen noch ungelösten Disziplinen. Das herkömmliche Prinzip der Berechnung von Tiefeninformationen aus der Suche nach Korrespondenzen in zwei oder mehr Bildern führt aufgrund von Mehrdeutigkeiten, die durch transparente Materialien verursacht werden, zu unzufriedenstellenden Ergebnissen. In dieser Arbeit untersuchen wir daher einen alternativen Ansatz, der auf der lokalen Orientierungsanalyse von Lichtfelddaten im Bezug auf Bilder innerhalb epipolarer Ebenen basiert. In diesem Zusammenhang diskutieren wir die Prinzipien von Struktur-Tensoren und deren intrinsische Eigenschaften. Ausgehend vom Basisfall für opake Oberflächen werden wir weiterführend erklären, wie diese Tensoren auf Modelle höherer Ordnungen erweitert werden können, sodass Mehrdeutigkeiten bezüglich der Tiefenschätzung von transparenten Oberflächen behandelt werden können. Darüber hinaus präsentieren wir unsere Erkenntnisse und Verbesserungen der Methode. Basierend auf einem Projektionsansatz mittels Voxel-Volumina stellen wir zwei Varianten einer Verfeinerungsmethode vor, die es uns erlaubt, verbesserte Ergebnisse für Szenen mit transparenten Objekten zu erhalten. Da diese Arbeit Teil eines Projektes aus einem industriellen Kontext darstellt, gehen wir im Verlauf dieser Arbeit von bestimmten Bedingungen bezüglich des Aufnahmesystems und der Daten aus. Die vorgestellte Methode ist jedoch auch auf ein allgemeineres Umfeld anwendbar.

Acknowledgements

First and foremost I want to thank Prof. Thomas Pock for his continuous support and guidance in all aspects related to this thesis. I'm also very grateful for his initiatives which enabled this thesis in the first place.

Since this thesis is part of a collaboration with the Austrian Institute of Technology, special gratitude goes towards my colleagues from the team lead by Dr. Markus Clabian. I appreciate all their help and support regarding organisational as well as practical matters. Of course I am also very grateful that I had the chance to gather hands-on experience during my internship at AIT. A very special thanks goes to Dr. Svorad Štolc for his valuable input and dedication throughout this project. His support was key to the successful outcome of this thesis.

Furthermore I want to thank my family and friends for supporting me throughout my studies and enabling my devotion for what I love doing most.

At last, as a matter most dear to me, I want to thank Sandra for her love and support.

Contents

List of Figures

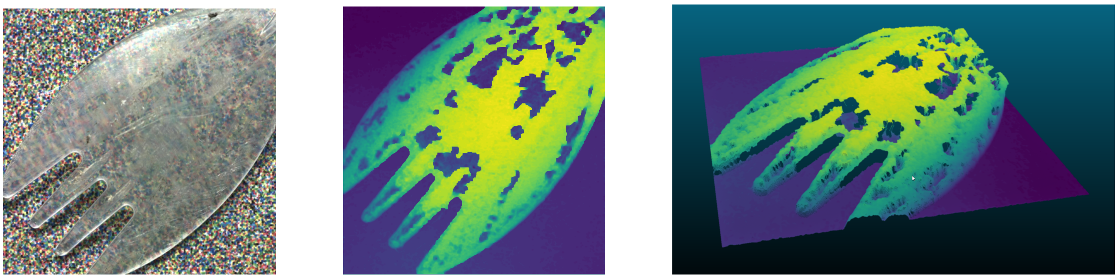
1. Introduction

1.1 Motivation

Inspired by the human visual apparatus, research in computer vision dedicates significant effort towards bringing visual perception and understanding to the digital domain. For a majority of tasks from day to day life, we naturally rely on our eyes. However, mimicking this through visual computing is far from trivial. This gave rise to numerous research fields over the past few decades. Among others, a very important one is the spatial perception of a scene in 3D. In analogy to the natural process of stereopsis, it is also possible for machines to generate 3D information from a set of digital images through stereo vision [32]. Stereo vision (also stereo matching) is an essential discipline of computer vision and is crucial for tasks such as tracking, pose estimation, object identification and scene understanding. Each of these tasks can be used to solve numerous problems in different areas, however a vast amount of research originates from applications in the context of industrial production and quality control. To ensure persistent levels of product quality, visual computing can be used for an automated inspection of objects or entire scenes on macroscopic as well as microscopic levels. Detecting defects or anomalies in production is a powerful and efficient way to ensure industry standards. Also the validation of certain measurements can be a desired application. However, image noise, occlusions, reflections, surface conditions and many more border constraints render this a challenging problem. While there is a fair amount of research available which focuses on these problems, very little work has been published to address the acquisition of non-opaque surfaces. A successful acquisition of the 3D shape of transparent/semi-transparent objects from stereo vision is very hard to achieve since the scene can appear deceiving and ambiguous. Suppose a given scene with a transparent object in front of an opaque background, see fig. 1.1. In such a case the information obtained from acquired images is a composition of the transparent object and the background. In conjunction with other unwanted effects such as reflection or refraction it is a non-trivial task to separate this composition in a way such that the depth information regarding the transparent object as well as the background can be retrieved. Even for human inspection it can sometimes be hard to determine the shape of transparent objects. Without prior knowledge and understanding of the scene, determining shape information in such a scenario would pose an equally hard task for a human.

Needless to say, the ability to acquire 3D information from transparencies is a desirable tool for research and industry alike. One reason for example is due to the fact that many modern production processes commonly involve the use of transparent materials, e.g.

transparent plastics or glass. Applications in packaging are very common, however also others such as consumer utilities, optics or medical health-care can be found frequently. The goal of this thesis is to present a method that represents a reliable solution for the depth estimation of scenes with transparent materials, that is compatible with a given inline computational imaging setup [1] developed by the Austrian Institute of Technology (AIT). Most approaches which are suitable for this setup deliver unsatisfying results when applied to a scene with transparent objects. This is why this work intends to present a new approach dedicated to compute depth information for such distinct applications.



(a) Image from image pair **(b)** Depth map **(c)** 3D point cloud

Figure 1.1: Exemplary result from a modern stereo method with a (semi-)transparent object. Notice that many estimates of the transparent object originate from the background of the scene, hence the depth map shows a lot of discontinuities between the object surface and the background.

1.2 Related work

The depth reconstruction of transparent objects is considered as one of the few unresolved disciplines of computer vision. Due to the challenging nature of the problem there has been a sparse amount of publication on the topic ever since. In a general setting there have been very few solution based on very distinct approaches over the last two decades. To tackle the basic problem of depth ambiguities, early work of Szeliski *et al.* [46] addresses stereo matching in conjunction with two layer ambiguities (foreground and background) as in the case of depth discontinuities and occlusions. Tsin *et al.* [59] present an approach to transparencies where compositions of colour are solved through spatial-temporal differencing between different stereo images. Based on active stereo, another interesting approach utilizing tomographic reconstruction in the visible light spectrum has been presented by [47]. Miyazaki *et al.* [34] present a method in which they estimate shapes of transparent surfaces using an inverse polarized raytracing approach. A very recent approach to full 3D reconstruction of transparent objects has been presented in [57]. In this paper the authors propose a method based on a full 360° acquisition in conjunction with a structured light source. Multiple surface and silhouette constraints are then used to refine the model iteratively. Another approach based on RGB-D data has been proposed by [24].

Whilst the more recent approaches are very promising all by themselves, it is necessary to point out that most of them require dedicated hardware and setups which are not applicable to our needs. Recall that it is a necessary criterion of this thesis to present a solution that is compatible with the aforementioned scanning setup [1], [50]. Thus the scope of this thesis is limited to passive methods only. Unfortunately, most of the few available solutions are based on active methods.

There have been promising publications by Wanner and Goldlücke on their work on structure tensors for 4D light fields [51]. Their principle is closely related to multi-view stereo matching. However in their approach, scene information is obtained from a 4D light field description instead of a multitude of individual images. In their later work in [52] the authors extend their approach to scenes with transparent and reflective surfaces. Instead of the traditional procedure of computing a single depth estimate from a given dataset, they extend the structure tensor approach such that it is possible to obtain depth estimates for transparent surfaces. Note that although their light field description is made up of two spatial- and two angular dimensions it is still applicable to our case. With our inline scanning setup we are able to acquire light field data in two spatial- and one angular dimension. Although we have less angular information we chose this structure tensor based method as the basis for our proposed solution. This will become clearer in the course of the upcoming sections. Note that there has also been a related learning based solution by Johannsen *et al.* [25], where the problem is tackled by solving sparse coding problems based on a dictionary of patterns. However solving a number of local sparse coding problems is computationally extremely demanding which likely proves to be impractical for industrial applications.

We will later introduce a method to refine depth estimates through the projection into a discretized voxel volume. This refinement method was inspired by other projection methods for combining multiple depth maps as for example through signed distance functions [12], [21], [55], [15].

1.3 Outline

As mentioned previously, the goal of this thesis is to highlight the complex of problems of depth estimation for scenes with transparent materials. Additionally we present a new approach to solve this problem under the requirements given by the acquisition setup [1], [50] in an industrial context. Thus this thesis will start by discussing preliminaries related to the principles of depth estimation from multiple images and will gradually explain the extension to light field descriptions. Furthermore we will coarsely explain the principle of the used acquisition setup and the light field data acquired from it. Moreover we will also show how depth information can be obtained from a given light field.

Since our presented solution is based on depth estimation through structure tensor methods as in [51], we will subsequently discuss the principles and intrinsics of the basic structure tensor model thoroughly. In the course of this discussion we will highlight the derivations of the model for better understanding and review different aspects of the method along with intermediate results. Moreover we will also present relevant findings and improvements.

From there we will shift the focus of the discussion to higher order structure tensor models as in [3]. In the same manner as for the basic counterpart, we will extensively examine these tensors. The key properties concerning the depth reconstruction for transparent materials will also be evaluated along this line. Since each of these different models assumes a certain number of depth layers, we will also show the effects of different models on data with different amounts of layers. With these model based solutions at hand we will later present a model independent refinement scheme in two distinct variants. Based on voxel-volume projection methods, we are going to show how we combine depth estimates in a way such that we obtain a smooth solution for scenes which include transparent materials. Additionally we will explain our implementation in greater detail. We finalize this thesis with a discussion of the end results and point out findings regarding different aspects.

2. Preleminaries on Light Fields and Depth Computation

Before we focus on the main body of this thesis, we want to give an introduction to the basic principles and ideas relevant for the subject. This shall create a good foundation for a better understanding of the content which will be presented in the subsequent chapters.

2.1 Depth from Stereo Matching

In this section we are going to briefly discuss the origins of depth computation through stereo. Note that this work is strictly speaking based on computing 2.5D estimates, however, since 2.5D is a special case of 3D and for the sake of simplicity, we will continue to refer to 3D depth estimates instead.

The key idea behind stereo matching is to compute each point of a scene in 3D space by correlating pixels from different images. The relative rotation and translation between both camera orientations must be known. While this principle is valid for arbitrary relations from one image to another, the computation of depth is vastly simplified if there is no rotation and the translation between camera centres is aligned with one image axis (parallel stereo). This constraint is used in stereo matching to simplify the computation of disparity from local neighbourhoods around each image pixel [19], [14], [18], [37].

Consider the example in fig. 2.1. With the given (stereo) setup, the depth estimate z of any point in the scene can be easily computed by the relative horizontal disparity d, the focal length f and the baseline b. The epipolar plane intersects the image plane and forms an epipolar line [19]. Along this epipolar line the disparity d can be computed through $(y_1 - y_2)$. Note that $x_1 = x_2$. The distance z from the scene point SP_1 to the image plane can thus be computed through the following geometric rule:

$$z = f\left(\frac{b}{d} - 1\right) = f\left(\frac{b}{y_1 - y_2} - 1\right) \tag{2.1}$$

Notice that disparity and depth are inversely proportional. The principle of stereo matching from two images has been a fruitful research topic ever since [60], [33], [45], [26]. Of course there have been numerous extensions and variations to the principle. One of the most common is the extension to the case of multiple images instead of only two. The

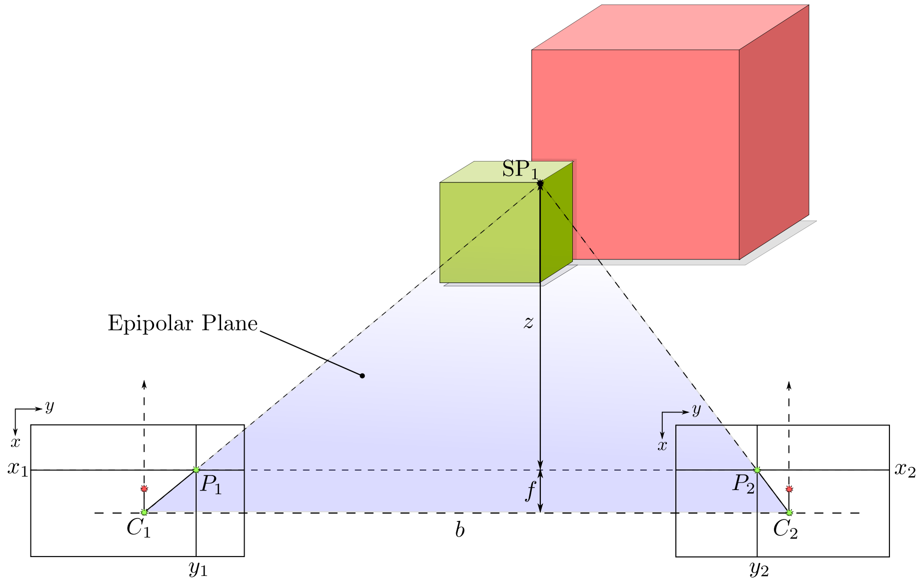


Figure 2.1: Principle illustration of stereo matching with two aligned cameras. A scene is imaged from two different camera-centres C_1 and C_2. The corresponding feature point from the scene (SP$_1$) is projected to P_1 and P_2 in the respective image plane. The focal length is depicted by f, the baseline by b and the sought depth estimate by z.

basic idea is to capture more images in-between to increase the robustness w.r.t finer details or image noise [36], [48]. This is referred to as multi-view matching. The principle of computing depth by finding correspondences between multiple image pairs is closely related to the method of computing depth estimates from a light field. In the next section we will show this relation in further detail.

2.2 Light Fields

Based on the idea of computing depth estimates from multiple images, there is also a more sophisticated method. If we interpret each given image as a single layer comprised of voxels instead of pixels by giving it a unitary 3^{rd} dimension and from there proceed to stack said layers along this new dimension, we obtain a volume as in fig. 2.2. This volume can be described through a so called light field [58], [30]. A light field (LF) describes

the function of light at each point in 3D space from different perspectives and was first introduced to computer vision in [30]. The most general case of a light field is described by the plenoptic function [4] in 7D:

$$L(\theta, \phi, \lambda, t, V_x, V_y, V_z)$$

where θ, ϕ describe the ray direction of observation, λ is the wave length, t the time and V_x, V_y, V_z describe the coordinates of the camera centre in space which consequently is the location from which each image is captured. In conventional imaging application with

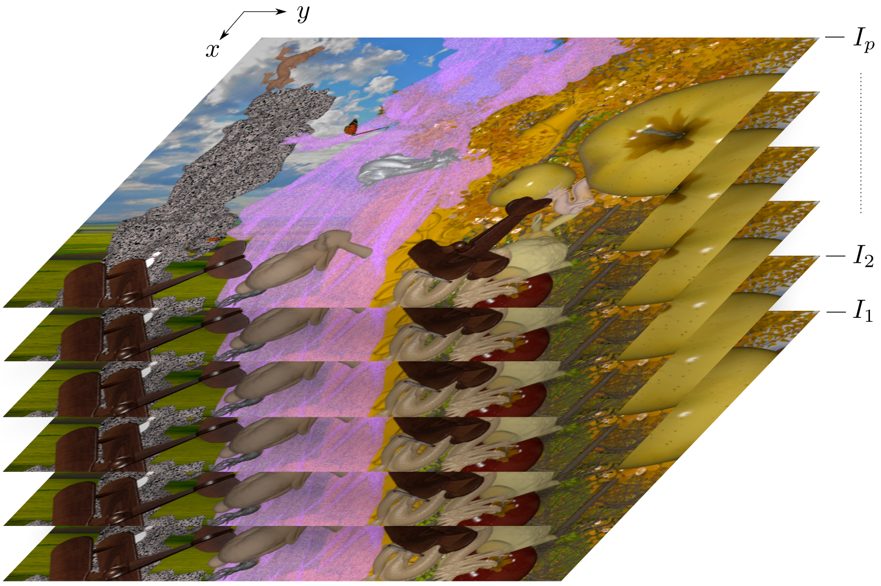


Figure 2.2: Collection of P input images stacked on top of each other. The depicted images originate from the scene 45 dataset from [20].

a static setting where only grayscale or RGB images are described, the complexity of the light field reduces to 5D. Instead of describing each point of a scene by the observed direction of the two angles θ, ϕ one can alternatively use image coordinates x and y. The light field thus is described by

$$L(x, y, V_x, V_y, V_z)$$

Depending on the degrees of freedom of the camera movement (3D/2D/1D) there is a multitude of methods to acquire such light fields. One option is to use plenoptic cameras as presented by Perwass *et al.* [38], [39] or using a conventional camera, mounted onto linear rails driven by a stepper motor [51]. A different approach has been presented by Štolc *et al.* [50] where a stationary multi-line-scan camera captures a 3D light field from an object, which is moved on a linear rail. The scene acquired from this setup is described through a light field of the form

$$L(x, y, s). \tag{2.2}$$

Here x denotes the vertical sensor resolution of the camera and y denotes the acquisition during movement. Furthermore s denotes the index of the sensors scanlines. A simplified depiction of this acquisition setup is shown in fig. 2.3. Since each scanline forms one image, this definition of the light field is equivalent to the stacked voxel volume defined in the beginning of this section. Because this acquisition setup will be the basis for all of the lab data recorded and evaluated in this work, we will focus on this LF definition from now on.

Throughout the course of this thesis the phrase LF stack may also appear. Note that this is just another term for the light field description or the voxel volume interpretation.

Recalling the epipolar properties from section 2.1 we can now apply this knowledge to the

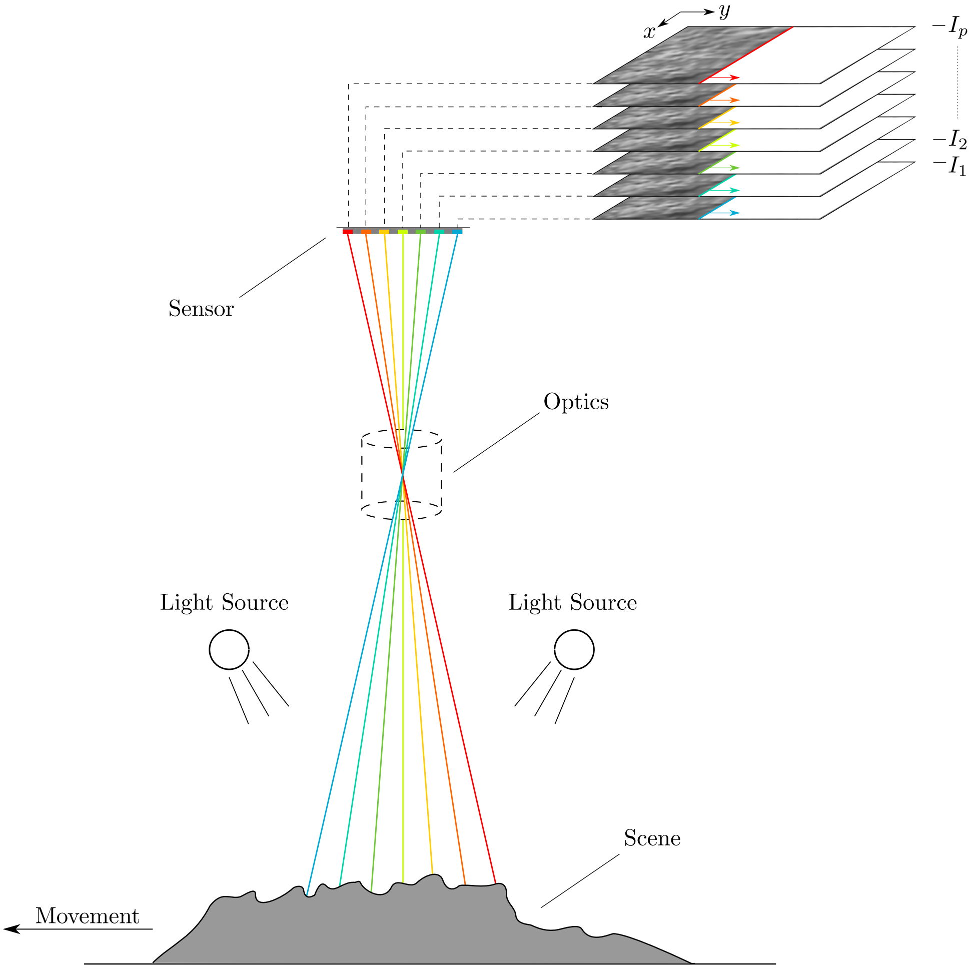


Figure 2.3: A conceptual illustration of the inline acquisition setup from [50] and [1]. The principle is based on the relative movement between the camera and the scene. Different scanlines on the camera sensor acquire the scene from different angles. Each scanline forms an image of the light field stack.

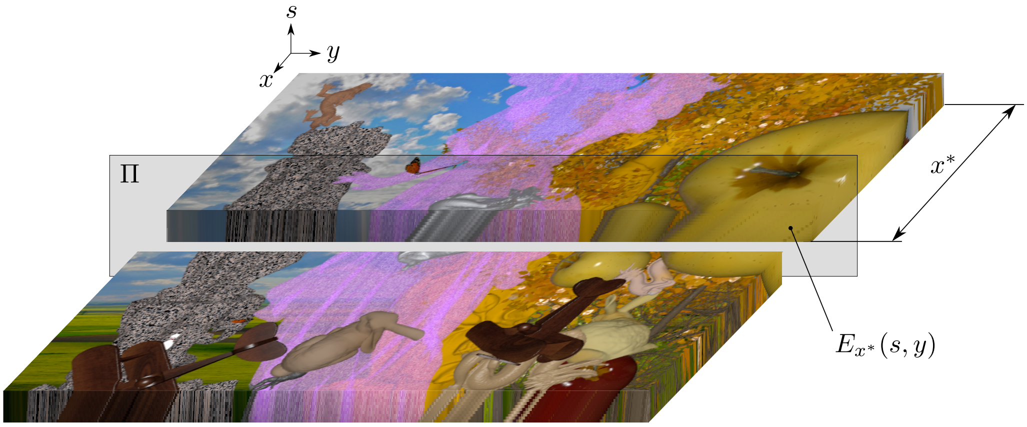


Figure 2.4: Light field L(x, y, s) depicted as a volume in x, y, and s. The plane Π intersects the volume at x^* to illustrate the epipolar plane image $E_{x^*}(s, y)$. Images from scene 45 dataset from [20].

light field $L(x, y, s)$. From the aforementioned acquisition setup [50] we know that due to the constrained multi-line principle, all images share the same epipolar planes w.r.t the movement direction. By intersecting these epipolar planes with each image plane, this therefore results in the same epipolar lines in all images. Thus by fixing x at any value x^* within the range of image height across all images, we obtain an image that is known as an epipolar plane image (EPI), see [51]. In the notational form of the light field description it thus can be denoted

$$L(x^*, y, s) := E_{x^*}(s, y). \tag{2.3}$$

Fig. 2.4 depicts such an light field as a volume with the dimensions $M \times N \times P$. M and N denote the original image height and width. Notice that the faces on the sides of the volume show the corresponding EPIs of the light field. From the earlier example it can be seen, that the EPI at x^*, $E_{x^*}(s, y)$ is depicted as well. As an easier way of comprehending epipolar plane images, one can also imagine dissecting the volume at plane Π. A detailed look at the EPI pattern reveals, that corresponding scene points across all input images form a traceable line or pattern in the EPI. Notice that different lines at each location in $E_{x^*}(s, y)$ have different slopes which indicates different depths from the 3D scene, see fig. 2.5. By computing the slope of the pattern for each location it is thus possible to compute depth information.

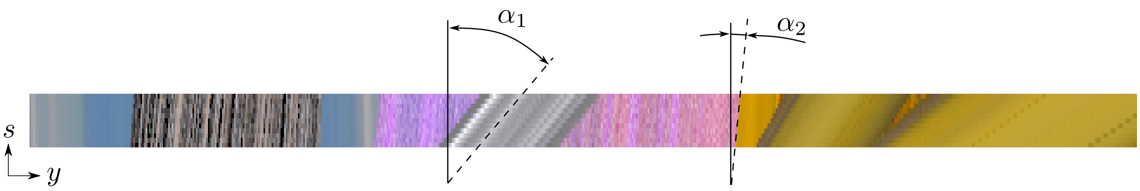


Figure 2.5: Epipolar plane image $E_{x^*}(s, y)$. Notice how different lines/patterns have different angles α_1, α_2 what indicates different depth. Images from scene 45 dataset from [20].

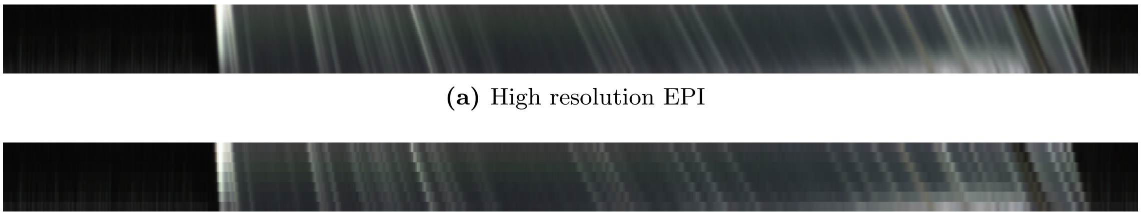
In section 2.1 it was shown, that depth can be computed from the disparity of a feature in two images. With the light field description discussed in this section, equivalent can be achieved by estimating the line/pattern angles of the epipolar plane images. Note that although both approaches allow to compute depth estimates, they are not identical. The numerical relation between both approaches is briefly evaluated in section 3.3.
To summarize, by using this light field representation it follows that it is possible to infer the depth of any scene point by estimating the angle of the local pattern structure in the corresponding epipolar images. The actual angle estimation from a local neighbourhood can be achieved in different ways. A local approach is the testing of hypotheses as presented by Soukup *et al.* [44]. The principle is that a fixed number of angular hypotheses are tested against a local pattern and evaluated based on some cost measure. The best hypothesis is then used to infer the depth estimate. Note that this approach assumes constant radiance across all P images. Problems may occur in case the constant radiance assumption is violated by e.g. reflections or transparencies/occlusions. A different approach has been presented by Wanner and Goldlücke [51] based on the use of so called

structure tensors. The key idea is that their approach is able to estimate local pattern orientations based on a statistical analysis of gradient information in the EPI domain. Because an estimate can be computed directly, there is no need for evaluating a hypothesis. As mentioned earlier, the use of structure tensors will be the key principle throughout this thesis. In the later course of this thesis we will explain how this principle can be extended to the application of light field descriptions with transparent materials.

2.3 Data

This section will briefly discuss the structure and properties of data used throughout this paper. Most of the results presented in this thesis originate from acquisitions through the inline computational imaging (ICI) setup [1], [6], based on the principle introduced earlier in [50]. One point to emphasize is, because the referenced setup is equipped with two light sources it is possible to acquire one light field dedicated to each light source individually. This will be a useful advantage as it will be shown in a later section. The light field dimensions from this setup are denoted $M \times N \times C \times P$. The spatial dimension M corresponds to the vertical resolution of the camera sensor, N is the number of scanline acquisitions along the direction of movement and defines the spatial resolution. C denotes the number of colour channel times the available illuminations. Since the data acquired for evaluation is in RGB and we use two light sources, one from the left and one from the right, the parameter $C = 6$. The RGB light fields are thus stacked along this dimension. As denoted in section 2.2, P denotes the number of images, which implies the number of inline scanlines used for acquisition, i.e. the angular resolution. More specifically the resulting light field dimension thus is $M \times N \times 6 \times P$.

Most setups which allow light field data to be acquired are limited w.r.t. to angular views (i.e. images taken from different angles). Usually this is conjoined by a trade-off between spatial and angular resolution. Due to this shortcoming, effort has been invested into computing super-resolution light fields from sparser sampled acquisitions. There are multiple methods to achieve this. Rossi *et al.* [40], [41] provide a graph-based method, Yoon *et al.* [61] and Gul *et al.* [16] present methods based on CNNs. However, angular and spatial resolution is crucial to acquire fine details and to combat noise. From our findings, this has also another significant impact on the results for scenes including transparent/semitransparent materials. Due to the fact that transparent surfaces impose very little to no structures in EPIs, which are usually very fine, a high angular and spatial resolution is crucial to computing depth estimates through light field based methods. Fig. 2.6 depicts a comparison of epipolar plane images with varying resolution. Because of the lower spatial and angular resolution in one of the images, it is difficult to make out continuous structure or traceable lines which is of high importance for the used method. This will be clearer upon the explanation of the used method in the upcoming sections. S. Wanner [54] provides a more detailed insight about limitations in disparity between adjacent views and shows that if disparity gets too high, it is no longer possible to perform an orientation estimation.



(a) High resolution EPI

(b) Low resolution EPI

Figure 2.6: Comparison between an EPI with high spatial- and angular resolution and a low resolution EPI. Note how the lines in the low resolution EPI show a staircasing effect.

One of the benefits of using the aforementioned inline setup (ICI) is, that the spatial as well the angular setup can be selected through enabling and disabling scanlines and choosing the acquisition interval. This convenient fact lets us thus choose P and N. In our acquitions we found that a high angular resolution ($P > 30$) works well. The vertical resolution given by the sensor upper bounds the resolution in x. Comparable datasets which are available online e.g. [23], [53], [20] usually work on lower spatial and angular resolutions. An example for densely sampled light fields is given by [5]. However, since research on light fields in conjunction with transparent materials has been very sparse, there is very little we can compare to. We found only one dataset which is dedicated towards transparent materials ('Maria' dataset from [53]). This dataset was published by the same authors that presented [52]. Note however that in their work they only show results on data with planar reflective or transparent surfaces. In this thesis however we will study depth estimation for transparent objects with more general shapes and forms.

3. Depth Estimation for Lambertian Surfaces

During the course of the previous chapter we conveyed the most crucial basics related to this thesis. In this section we are going to discuss the core principles and findings based on the aforementioned method. More specifically, we will focus on the depth estimation for scenes where it is assumed that each scene point fulfils the Lambertian assumption. For this assumption to hold true, it is obligatory that a scene point as part of a (Lambertian) surface reflects radiance in all directions equally [28]. Because we use the aforementioned light field description, this assumption means that the imposed lines and structures in the EPI domain are clean and continuous. Another property of this assumption is that pattern patches in the EPI domain will have exactly one orientation. In case this assumption is violated, these properties will not hold. This can be the case for transparencies or occlusions in a scene, however, we are going to discuss this in a later chapter of this thesis. In the course of this chapter we instead will focus on the the applicability of structure tensors to the task of depth estimation. As briefly mentioned in section 2.2, the structure tensor can be used to determine pattern or line orientations in an image. Since the first order structure tensor model, which will be discussed in the upcoming parts, is capable of estimating exactly one orientation, it will also be referred to as the single orientation structure tensor (SOST). Before we dive into the derivation of the first order model in further detail, we will briefly discuss the interpretation of pattern orientations and the corresponding angular estimates. Since our solution relies on angular estimates instead of disparities, we have spent plenty of time investigating the problems that may arise with this representation. Most known structure tensor based implementations assume disparities for their results. However it is crucial to have continuous angular estimates for the refinement method as presented by our extension in chapter 5. In addition the results need to be consistent with those from higher order structure tensors which will be introduced in chapter 4.

3.1 Pattern Orientations and Angular Estimates

The range of a valid angle ξ for the denotation is constrained to be $\xi \in \left(\frac{\pi}{2}, -\frac{\pi}{2}\right)$, as this is the range for which the tangent function is unambiguous. It follows the definition of the orientation vector $w(\xi) = [\cos(\xi), \sin(\xi)]^{\mathrm{T}}$. The drawback of this is that lines in EPIs are effectively limited to the interval $\xi \in \left[0, \frac{\pi}{2}\right]$. Because we use angular estimates instead of disparities as our result, this representation will lead to problematic discontinuities in the structure tensor estimates. In our framework we aim to distribute our full angular resolution over the whole available range as best as possible, to get the maximum out of our data. Thus it is necessary to map the angular estimates from EPI lines or patterns to our desired range. This means that we have to utilize those angles in $\xi \in \left[\pi, \frac{\pi}{2}\right]$ as well. If flipped by a factor of π, the angles in this interval correspond to $\xi \in \left[0, -\frac{\pi}{2}\right]$. With our acquisition setup we are able to acquire light fields where EPI lines with angles over the full interval of $\xi \in \left[\frac{\pi}{2}, -\frac{\pi}{2}\right]$ can occur, see fig. 3.1. The angular range can be shifted by changing the focal point of our setup. Details about this will not be discussed in this thesis - for further reading we refer the reader to the adequate references [50], [44]. The problem with selecting the angular range in our lab data in this way is that depths that correspond to $\xi \approx \frac{\pi}{2}$ are directly neighbouring those corresponding to $\xi \approx -\frac{\pi}{2}$ which leads to these unwanted discontinuities in our results. Note that this is not the case for example in the synthetic data of Heber and Pock [20] as depicted in fig. 2.5, since in their data only angles $\xi \in \left[0, \frac{\pi}{2}\right]$ occur.

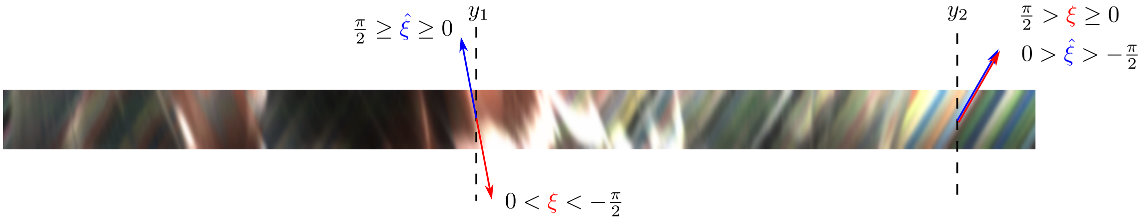

$\frac{\pi}{2} \geq \hat{\xi} \geq 0$
y_1
$0 < \xi < -\frac{\pi}{2}$
y_2
$\frac{\pi}{2} > \xi \geq 0$
$0 > \hat{\xi} > -\frac{\pi}{2}$

Figure 3.1: EPI with different pattern orientations. Notice that the depth for the local pattern at y_1 is close to $\xi = -\frac{\pi}{2}$ and at the same time is close to the pattern orientation at y_2. The orientation vectors for $\hat{\xi}$ point only 'up'.

Although sticking with this behaviour would not render further processing steps inapplicable, it is hard to interpret the correctness and quality of our results. This is why we employ the following measures to enforce smoothness and continuity of our results.
As a first step, we flip all angular estimates $-\frac{\pi}{2} < \xi < 0$ by π. This gives us our new continuous range $(\pi, 0]$. The need for flipping the orientation can easily be determined by checking the sign of the second component of the orientation vector $w(\xi)$. Additionally because we want our estimates to reside around 0, we offset $\hat{\xi}$ by a factor of $-\frac{\pi}{2}$. This then leads to our angular range as illustrated by the pattern patches in fig. 3.2.

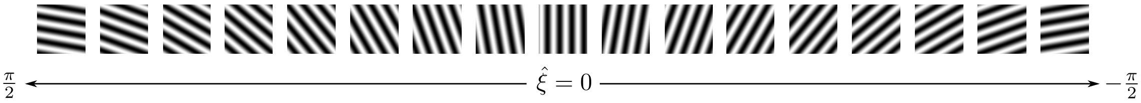
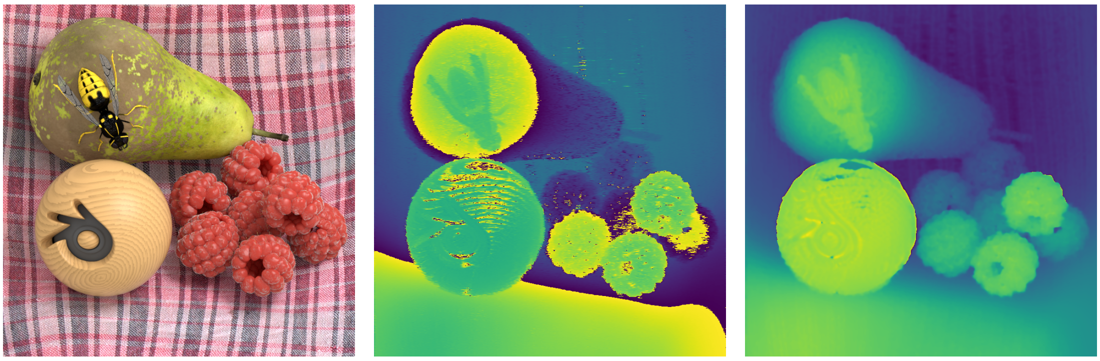


Figure 3.2: Pattern patches with the range of $\hat{\xi}$ annotated correspondingly.

A direct comparison with the EPI given above (fig. 3.1) shows clearly, that because of orientations in both quadrants of $\xi \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ this is the most effective method of defining the notation for our angular estimates.

Fortunately for the results from the first order case this does not require any mapping whatsoever. Instead through the direct computation, as it will be shown in the upcoming section, it is possible to enforce the resulting vector to fulfil $w(\xi)^{\mathrm{T}} \cdot [0 \;\; 1]^{\mathrm{T}} > 0$. Possible effects of this discontinuity behaviour are depicted in fig. 3.3.



(a) Central input image **(b)** Angular estimate ξ **(c)** Angular estimate $\hat{\xi}$

Figure 3.3: Effects of the angular interval correction as introduced in section 3.1. Notice that for subfigure b) the results are discontinuous and flipped around $\xi = 0°$. The angular result $\hat{\xi}$, as presented by us, can handle data sets with orientations across the whole range $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$.

3.2 Single Orientation Structure Tensor (SOST)

Next we are going to discuss the mathematic principle behind the first order or single orientation structure tensor. The derivations below will follow [3] and extend it for further clarity. Recalling our goal of estimating the orientation $w(\xi)$ of a local pattern in the EPI domain, we define a patch within a region Ω from a grayscale image $f : \mathbb{R}^2 \to \mathbb{R}$ given by a function

$$f(x) = f(x + kw) \qquad \forall x \in \Omega \tag{3.1}$$

where $k \in \mathbb{R}$ denotes a step size and $x \in \mathbb{R}^2$ denotes the bivariate image coordinates in y and s. From previously, we know that $w(\xi) = [\cos(\xi), \sin(\xi)]^{\mathrm{T}}$ denotes the orientation of our pattern. In the context of the Lambertian assumption as described in the beginning of chapter 3, it is also the direction along which the radiance in an EPI patch stays constant. This means that the gradient along this direction must vanish. Thus the following can be denoted [27], [8]

$$\frac{\partial f(x)}{\partial w(\xi)} = w(\xi)^{\mathrm{T}} \cdot \nabla f(x) = 0 \qquad \forall x \in \Omega \tag{3.2}$$

The given constraint is fulfilled by an energy term. This energy term can be formulated as

$$E_1(\xi) = \int_\Omega \left(w(\xi)^{\mathrm{T}} \nabla f(x)\right)^2 d\Omega = 0 \tag{3.3}$$

As a general side note, the gradient which is applied in the EPI domain is denoted by $\nabla f(x) = \left[\frac{\partial f(x)}{\partial y}, \quad \frac{\partial f(x)}{\partial s}\right]^{\mathrm{T}}$. Condition eq. (3.3) can be reformulated in the following way

$$E_1(\xi) = w(\xi)^{\mathrm{T}} \left(\int_\Omega \nabla f(x) \nabla f(x)^{\mathrm{T}} d\Omega \right) w(\xi) = w(\xi)^{\mathrm{T}} S \ w(\xi) \tag{3.4}$$

Through this reformulation the first order structure tensor S is introduced. The formal definition of the tensor in a continuous setting thus follows

$$S = \int_\Omega \begin{bmatrix} d_y^2 & d_y d_s \\ d_y d_s & d_s^2 \end{bmatrix} d\Omega. = \int_\Omega \begin{bmatrix} S_{11} & S_{12} \\ S_{12} & S_{22} \end{bmatrix} d\Omega. \tag{3.5}$$

Note that through its composition this must be a symmetric, positive semi-definite matrix. d_y denotes the gradient in y while d_s denotes the gradient in s.
Our general goal now is to find a solution to $w(\xi)$ such that $E_1(\xi)$ becomes 0. Note that the energy function $E_1(\xi)$ can only be zero if the whole image patch has a perfect single orientation pattern structure with no noise or variations along $w(\xi)$ whatsoever. In real world data this is practically never the case, thus $E_1(\xi)$ will never reach 0. The next best thing that can be done though is to keep the energy as minimal as possible. As a side note, this is similar to the idea of a principle component analysis (PCA) where the orientation with the least variability after projection is sought. We are thus searching for the optimal solution ϕ to the target function

$$\underset{\phi}{\operatorname{argmin}}\, E_1(\phi) = w(\phi)^{\mathrm{T}} S \ w(\phi) \quad s.t. \quad w(\phi)^{\mathrm{T}} \cdot w(\phi) = 1 \tag{3.6}$$

Constraining the direction vector as denoted above serves the purpose of excluding the trivial solution $w(\phi) = 0$. The solution to this problem can be solved through the Lagrange function

$$L(w, \lambda) = w(\phi)^{\mathrm{T}} S \ w(\phi) + \lambda(1 - w(\phi)^{\mathrm{T}} \cdot w(\phi)) \tag{3.7}$$

This leads to the optimality conditions

$$\begin{aligned} \nabla_w L(w, \lambda) &= S \ w(\phi) - \lambda w(\phi) = 0 \\ \nabla_\lambda L(w, \lambda) &= w(\phi)^{\mathrm{T}} \cdot w(\phi) - 1 = 0 \end{aligned} \tag{3.8}$$

From this notation it can be derived, that finding the optimal solution to the problem is equivalent to performing an eigenvalue analysis of the structure tensor S

$$S\ w(\phi) = \lambda w(\phi) \qquad w(\phi)^{\mathrm{T}} \cdot w(\phi) = 1 \tag{3.9}$$

where $w(\phi)$ represents the right hand side eigenvector. Thus we have shown, that the resulting eigenvectors indicate the orientations of the pattern patch within the region Ω. The upcoming section will explain the eigenvalue analysis further.

3.2.1 Eigenvector Analysis

In the previous section we derived, that the orientation of an image patch in the region Ω can be computed by an eigenvalue analysis. Let λ_1 and λ_2 be the eigenvalues of S where $\lambda_1 > \lambda_2$ and let v_1 and v_2 be their corresponding eigenvectors. It is clear that the eigenvector v_1 corresponding to the largest eigenvalue λ_1 of the structure tensor S points in the direction of the strongest variability, i.e. the direction of the gradient. Since the second (smaller) eigenvector v_2 must be orthogonal to v_1 one can compute the orientation of a pattern from the smaller eigenvector v_2. This is exactly the vector that minimizes eq. (3.6). The residual of the energy at the optimal solution ϕ^* thus is given by

$$E_1(\phi^*) = \lambda_2. \tag{3.10}$$

This means that a low residual indicates a high confidence for the single orientation assumption [3] of the image patch. Equivalently this must indicate an image patch in the vicinity of Ω that has a consistent, low noise orientation. On the contrary we can formulate the statement, that a high value for λ_2 indicates a high variation along the optimal orientation estimate which in turn might imply noise, a multi-orientation pattern or violation of the constant radiance assumption (i.e. reflection, occlusion etc.).

Since S is a symmetric 2×2 matrix, the relevant parameters can easily be computed in closed form. By definition they must fulfil

$$(S - \lambda_{1,2} I) v_{1,2} = 0. \tag{3.11}$$

Which leads to the characteristic polynomial

$$\det(S - \lambda I) = 0. \tag{3.12}$$

By solving for λ we get:

$$0 = \lambda^2 - \lambda(S_{11} + S_{22}) + S_{11}S_{22} - S_{12}^2$$

$$\lambda_{1,2} = \frac{(S_{11} + S_{22})}{2} \pm \sqrt{\frac{(S_{11} + S_{22})^2}{4} - (S_{11}S_{22} - S_{12}^2)}$$

$$\lambda_{1,2} = \frac{\text{trace}(S)}{2} \pm \sqrt{\frac{\text{trace}(S)^2}{4} - \det(S)}$$

From eq. (3.11) it follows, that an eigenvector $v = \begin{bmatrix} v_y \\ v_s \end{bmatrix}$ must fulfil

$$(S_{11} - \lambda)v_y - S_{12}v_s = 0 \tag{3.13}$$

Both eigenvectors can be derived from this, however for computing the angle of the pattern patch we are only interested in the eigenvector corresponding to the smaller eigenvalue which is v_2. This eigenvector points in the direction perpendicular to the gradient direction. Inserting λ_2 into eq. (3.13) gives

$$0 = \left(S_{11} - \frac{(S_{11} + S_{22})}{2} - \sqrt{\frac{(S_{11} + S_{22})^2}{4} - (S_{11}S_{22} - S_{12}^2)} \right) v_y - S_{12}v_s$$

$$v_s = \frac{S_{11} - S_{22} + \sqrt{(S_{11} - S_{22})^2 + 4S_{12}^2}}{2S_{12}} v_y$$

$$v_2 = \begin{bmatrix} v_y \\ v_s \end{bmatrix} = \frac{v_y}{2S_{12}} \begin{bmatrix} 2S_{12} \\ S_{11} - S_{22} + \sqrt{(S_{11} - S_{22})^2 + 4S_{12}^2} \end{bmatrix}$$

Since only the relation between v_y and v_s is known, it is up to interpretation on how to exactly compute the orientation vector. [3] and [54] restrict the vector to unit length through normalization and set one of either v_y or v_s to 1. With hindsight to section 3.1 we conveniently have the option to enforce $v_s > 0$, by setting $v_s = 1$. This ensures that the angular result does not need to be flipped by π and is already in the correct range for computing our angular result. The angular estimate $\hat{\xi}$ can thus be computed through

$$\hat{\xi} = \tan^{-1}\left(\frac{v_s}{v_y}\right) - \frac{\pi}{2} \tag{3.14}$$

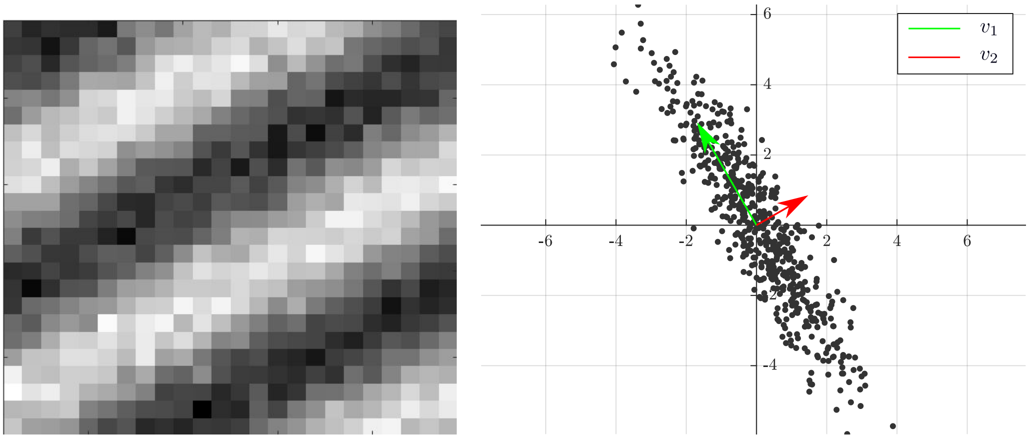
As an interesting alternative, the next section will present a different though process that leads to solving the problem of pattern orientation through a least squares problem.

3.2.2 Pattern Orientation through LSP

An alternative idea of finding the same orientation is to formulate a least squares problem (LSP) from all pixel-wise gradients of an image patch in a region Ω. Since the gradient points into the direction of the highest variability, we need to find the orientation that is most orthogonal to the cumulative gradient. Thus to find the optimal solution w^*, it would be also possible to find the solution to the least squares problem

$$\min_{w \in \mathbb{R}^2} \frac{1}{2} \sum_{i}^{\Omega} (g_i^{\mathrm{T}} w)^2 \tag{3.15}$$

Note that this notation does not include the structure tensor S but instead uses gradient information directly. An example for this principle is depicted in fig. 3.4. The second plot in this figure depicts the point cloud of gradients which also represents the principle of the structure tensor in some indirect way. Since the structure tensor is comprised of gradient information this point cloud may serve as a bridge to the related eigenvalue analysis. However note that the least squares cost function is convex and thus cannot be extended to multiple orientations. Therefore it will not be investigated further. This section shall only serve the purpose of helping the reader to get a better understanding w.r.t. pattern orientations.



(a) Pattern patch Ω **(b)** Gradients and eigenvectors

Figure 3.4: The left image depicts an example of a pattern patch sampled from region Ω (24×24). The orientation of this patch is $\hat{\xi} = -60°$. 20% of Gaussian distributed noise has been added. The right image shows a point cloud of pixel-wise gradients $g_i = [d_y, d_s]^{\mathrm{T}}$ (black) and the normalized eigenvectors v_1, v_2 from the eigen analysis of the structure tensor. Note that v_2 aligns with the pattern orientation on the left and has an estimated orientation of $-59,73°$ in this example.

3.2.3 Robustness and Smoothing

Up to this point the mathematical principles of the structure tensor have been introduced. This section will be dedicated to the discussion regarding the robustness of the structure tensor with respect to different metrics. The structure tensor S in eq. (3.5) is defined as a symmetric 2 × 2 matrix. When using the structure tensor for estimating pattern orientations in epipolar plane images, it is usually the case, that a weighting function on an inner-/ and an outer-scale is used [51]. This is done to make the structure tensor better scalable such that it can be adapted specifically to the scale of the pattern which helps to prevent aliasing affects. Another reason is, that because the structure tensor is based on the product of image gradients, sensitivity to noise is a concern. The weighting functions help to suppress sensitivity to noise. With these weighting functions the structure tensor can be denoted as

$$S = \int_{\Omega} G_{\sigma_o} * \begin{bmatrix} \hat{d_y}^2 & \hat{d_y}\hat{d_s} \\ \hat{d_y}\hat{d_s} & \hat{d_s}^2 \end{bmatrix} d\Omega. \tag{3.16}$$

Here $\hat{d_y}$ and $\hat{d_s}$ denote the gradient in y and s, smoothed by a Gaussian kernel with a inner scale of σ_i, i.e. $\hat{d_y} = \sigma_i * d_y$. Since in our case the gradients are computed from central differences and direct differences at the image borders, we use a convolution with a Gaussian row- and column-vector respectively. For a 3×1 or 1×3 kernel the (intermediate) inner smoothing operation would look like

$$\hat{d_y} = d_y * \begin{bmatrix} 0.25 & 0.5 & 0.25 \end{bmatrix}$$

$$\hat{d_s} = d_s * \begin{bmatrix} 0.25 \\ 0.5 \\ 0.25 \end{bmatrix}$$

This way we only smooth along the direction of the gradient operation and not across it. Another way of computing robust image gradients is through Gaussian derivatives [2]. This combines Gaussian smoothing with the computation of image gradients in one step.

$$\mathcal{N}(y, s|\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

$$\frac{\partial \mathcal{N}(y, s|\sigma)}{\partial y} = -\frac{y}{2\pi\sigma^4} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

$$\frac{\partial \mathcal{N}(y, s|\sigma)}{\partial s} = -\frac{s}{2\pi\sigma^4} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

If $f(x)$ denotes the input image, then the gradient image could be computed from the convolution

$$\hat{d_y} = f(x) * \frac{\partial \mathcal{N}(y, s|\sigma)}{\partial y} \tag{3.17}$$

$$\hat{d}_s = f(x) * \frac{\partial \mathcal{N}(y, s|\sigma)}{\partial s} \tag{3.18}$$

The Gaussian derivatives can also be used to compute the second order gradients in the same manner.

$$\frac{\partial^2 \mathcal{N}(y, s|\sigma)}{\partial y^2} = \frac{y^2 - \sigma^2}{2\pi\sigma^6} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

$$\frac{\partial^2 \mathcal{N}(y, s|\sigma)}{\partial y \partial s} = \frac{sy}{2\pi\sigma^6} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

$$\frac{\partial^2 \mathcal{N}(y, s|\sigma)}{\partial s^2} = -\frac{s^2 - \sigma^2}{2\pi\sigma^6} e^{-\frac{1}{2}\frac{y^2+s^2}{\sigma^2}}$$

In [29], Köthe proposes multiple improvements on the regular structure tensor for applications in the context of corner-/feature- detection. One of the propositions is to up-sample given image patches to avoid aliasing. The author states that this can be combined with Gaussian derivatives. Through Gaussian deconvolution the Gaussian derivatives thus can be used to compute the smoothed derivatives of an up-sampled image patch all at once. We evaluated both methods, however we could not determine an improvement for the orientation analysis. Due to the increased computational effort for up-sampling to a higher resolution we decided to use regular computation of derivatives with subsequent smoothing as described above. This also gave similar results as compared to using Gaussian derivatives (without up-sampling). Wanner [54] also investigated Sobel [56] and Scharr [7] operators and compared them to Gaussian derivatives. From the authors findings it can be deduced, that Gaussian derivatives perform best across different structure tensor scales. In general, from the findings during our evaluations we observed, that higher spatial resolutions require also higher angular resolutions and thus the structure tensor needs to be defined on larger scales. This is consistent with the findings from the reference.

In the notation eq. (3.16) from above, G_{σ_o} denotes the weighting operation on an outer scale. This outer operation is usually achieved through convolution with a Gaussian Kernel with standard deviation σ_o. However, because we found that residual border artefacts from computing the gradient in the EPI domain influence our estimates heavily, we decided to test other methods as alternatives. The idea is to mitigate the influence of those border artefacts. The results from this experimental evaluation will be presented in section 3.2.5. Since this outer scale operation in conjunction with the integration resembles sort of a cumulative aggregation over the elements of the structure tensor, it will also be referred to as the aggregation step.

It is necessary to point out, that from our findings this inner scale intermediate weighting in conjunction with the outer scale aggregation step is crucial to the stability of the method. Without both steps the method turns out to operate on a scope that is too local. As mentioned above, the method is prone to noise due to the fact that the structure tensor is a construct from products of gradients. This sensitivity to noise especially applies for structure tensor models of higher order, which will be clear once we discuss them in a later part of this thesis. Through the robustness study presented in this section, we can be certain that our chosen metrics are optimal with respect to our structure tensor implementation. Nevertheless, we require some way of measuring certainty that validates orientation estimates locally. This leads to the discussion about confidence in the single orientation structure tensor hypothesis which will follow in the subsequent section.

3.2.4 Single Orientation Confidence

In the previous sections we described how to compute robust depth estimates from analysing pattern- and line-orientations in the EPI domain based on the structure tensor. Since this is a model based approach, we need to choose a confidence estimate that supports our model hypothesis. As mentioned in section 3.2.1, the reliability of a single orientation hypothesis is supported by the residual of the energy term as in eq. (3.6). Recall that the residual is given by the smaller eigenvalue λ_2.

As an additional metric we want to have strong gradients orthogonal to the orientation of the image patch within Ω. This is the case if the larger eigenvalue λ_1 is large relative to λ_2. It follows, that in order to make a statement about the confidence in the single orientation hypothesis, we can analyse the results from the eigenvalue analysis. Based on this, Bigün *et al.* [8] propose to simply compute the difference between both eigenvalues

$$C_1 = \lambda_1 - \lambda_2 \tag{3.19}$$

This confidence measure all by itself gets the correct idea, however the measure proves to be problematic in case $\lambda_2 = 0$. The range of confidences in this case is reliant on the value of λ_1. Proposed by the same paper there is also another measure that is applicable in the same manner.

$$C_2 = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} = \frac{(\sqrt{S_{11} - S_{22})^2 + 4S_{12}^2}}{(S_{11} + S_{22})^2} \tag{3.20}$$

This measure is called coherence. Due to the simple structure of the closed form solution for the eigenvalues it can be computed from the structure tensor S directly. This confidence measure scales perfectly on the interval $[0, 1]$. We also chose this confidence measure for our implementations. As it will be shown in the later course of this thesis, it is very important that different model hypotheses are comparable to make reliable decisions for transparencies. Our choice for C_2 is additionally supported by the convenient scale and good expressiveness on the hypothesis. Examples depicting the confidence according to the presented confidence measure C_2 are presented in fig. 3.9.

3.2.5 Evaluation of Different Aggregation Methods

During the discussion in section 3.2.3 we mentioned an experimental evaluation of different aggregation methods on the outer scale of the structure tensor. This evaluation will be highlighted in this section. The goal of this evaluation is to find the most stable aggregation method w.r.t noise and border artefacts. For this evaluation we created a small test framework, in which we are able to compare structure tensor estimates on a small scale based on different aggregation methods. As a reminder, the aggregation method denotes the weighting and summation on an outer scale, see eq. (3.16). In this framework we create a window of size $9 \times 9 \times 1 \times 21$ ($x \times y \times 1 \times s$) and synthesize a pattern with a given orientation $\hat{w}(\hat{\xi})$. An illustration is shown in fig. 3.5. From this synthesized window we compute the orientation $w(\hat{\xi})$ and compare the result to $\hat{w}(\hat{\xi})$.

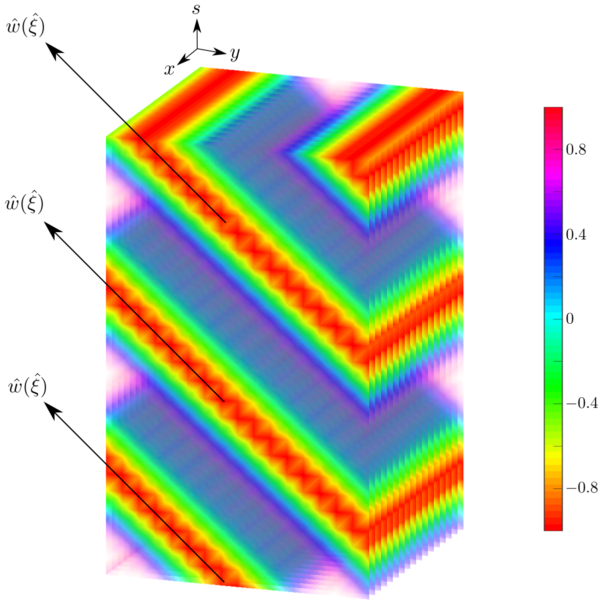


Figure 3.5: Synthesized window for testing different aggregation methods. In this figure the orientation points into a direction with an angle $\hat{\xi} = 45°$

Our evaluation is conducted in a way where we compute $w(\hat{\xi})$ with 50 equally distributed orientations $\hat{w}(\hat{\xi})$ on the interval $[85°, -85°]$. The elements of the structure tensor S are aggregated and evaluated by the following methods:

Sum

This method simply sums up all the estimates in the given window and parses the result to the eigenvalue analysis. Note that this method can be combined with the outer scale weighting as in section 3.2.3 since applying a box filter (i.e. sum) multiple times converges to the same result compared to applying a Gaussian filter.

Mean

Simple mean over all elements within the aggregation window.

Weighted Mean

This aggregation method is similar to the previous one, but each individual element in the sum is weighted with the corresponding confidence C_2 as in section 3.2.4.

Median

Collecting all elements in the aggregation window and computing the median of the resulting distribution.

Highest Confidence

From all the elements from within the window we choose the element with the highest corresponding confidence as the result for further computation through the eigenvalue analysis.

Mode

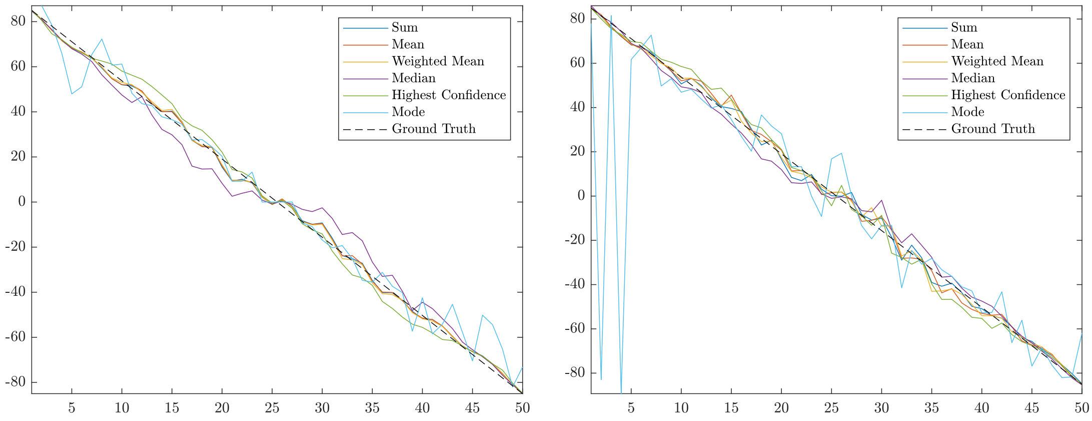
Collecting all elements in the aggregation window and computing the mode of the resulting distribution.

On top of the synthesized pattern window we add different levels of additive noise, which is sampled from a zero-mean normal distribution. From there we evaluated each method over the entire interval, see fig. 3.6. Each aggregation method is tested 3 times with different levels of noise. The averaged results of the SSE (sum of squared errors) over the entire angular range are depicted in table 3.1.

From the results it can be seen, that the most reliable method is given by summing up all elements from the aggregation window. Other methods like mean or weighted mean perform comparably, however this no surprise since mean and sum are closely related. Furthermore through further empirical testing with different aggregation windows and line widths we were able to confirm the results presented in this section. The results support the standard aggregation method as it is defined in eq. (3.16). Another argument supporting the chosen aggregation method is, that for whole light fields, border regions make out a smaller percentage as compared to the small framework in this evaluation. This is especially true for border regions in s- and less so in y-direction due to the relatively high spatial resolution and lower angular resolution. Overall the affected regions resemble only a small portion of the whole light field. Since the most robust method is given by the regular sum, we chose to use a box filter on the same scale as the outer smoothing kernel and aggregate 3 times. This gives us combined approximation of a Gaussian kernel and thus combines the weighting and the aggregation.

PSNR in dB	**SSE**					
	Sum	Mean	Weighted Mean	Median	Highest Confidence	Mode
26.02	253.31	259.50	272.26	2117.79	731.77	2017.18
16.48	274.94	258.64	277.86	1950.74	643.60	8138.10
12.04	288.95	331.21	387.54	965.24	754.27	57324.56

Table 3.1: The averaged results (over 3 runs) from the conducted tests on aggregation methods. The numbers represent the sum of squared errors over the entire tested angular range in degrees (°).



(a) PSNR= 26.02 dB

(b) PSNR= 12.04 dB

Figure 3.6: Exemplary results for tested aggregation methods over the evaluated angular interval $[85°, -85°]$ at different noise levels.

3.3 Relation to Stereo Matching

In section 2.2 we mentioned that orientation analysis in light fields and classical stereo matching are linked. The following section shows a naive analytical link between both approaches.

Recall eq. (3.5) and consider a theoretical region Ω of 1 pixel, i.e. computing the eigenvector for each individual pixel, then it is guaranteed that the smaller eigenvalue $\lambda_2 = 0$ since

$$det\left(\begin{bmatrix} d_y^2 & d_y d_s \\ d_y d_s & d_s^2 \end{bmatrix}\right) = 0$$

This simplifies the computation of the eigenvector v_2 to

$$v_2 = \begin{bmatrix} v_y \\ v_s \end{bmatrix} = v_y \begin{bmatrix} 1 \\ \frac{d_y}{d_s} \end{bmatrix} \tag{3.21}$$

A simplified illustration for this can be seen in fig. 3.7. This figure shows an EPI with some pixel values forming lines with some arbitrary orientation. Let D_{stereo} denote the disparity from stereo matching based on the mean absolute error within a certain region between $s = 1$ and $s = P$ in some local neighbourhood N. For the sake of simplicity, consider the winner takes all (WTA) result as in

$$D_{Stereo} \in \operatorname*{argmin}_{d} \sum_{x}^{N} |L(x^*, y, P) - L(x^*, y, 1)|\,. \tag{3.22}$$

Note that the disparity is reciprocally linked to the angle of the line. Let N_Ω be the number of pixels in the sampled region Ω and recall the eigenvector from eq. (3.21). Then the following can be denoted

$$\frac{1}{N_\Omega}\sum_\Omega \frac{v_s}{v_y} = \frac{1}{N_\Omega}\sum_\Omega \frac{d_s}{d_y} \simeq \frac{P-1}{D_{Stereo}} \tag{3.23}$$

It thus can be seen that the principles behind both approaches are linked. This relation is not so straight forward in case a larger region Ω is considered. A major difference between both methods is, that stereo matching is fairly independent from angular and spatial resolution. For stereo matching, higher resolutions mostly benefit properties like accuracy or robustness to noise. However for the orientation analysis in the EPI domain, sufficient spatial and angular resolution is a necessary prerequisite.

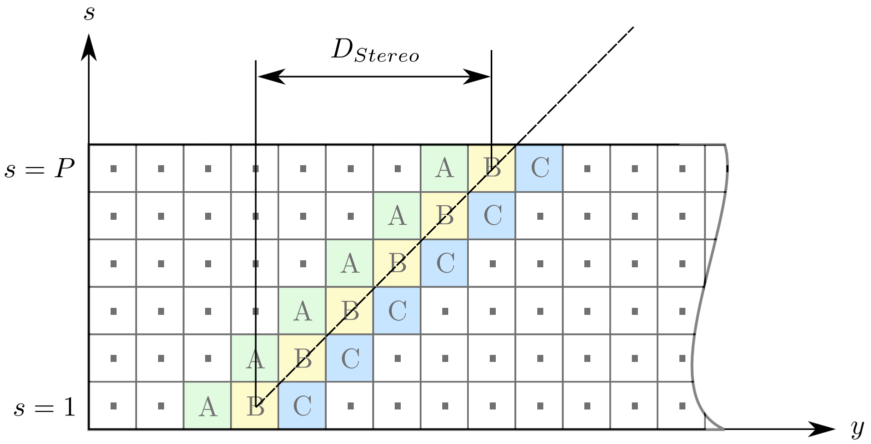


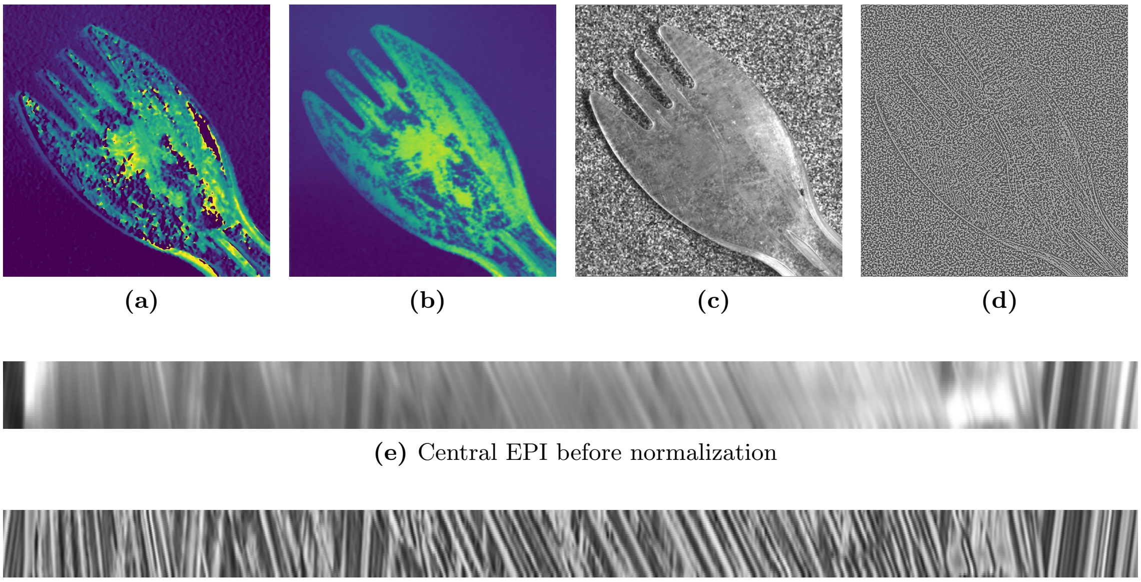
Figure 3.7: EPI illustration depicting the link between slope and disparity.

3.4 Contrast Normalization

Contrast normalization is a preprocessing step to make input data better suited for subsequent analysis. For our application we decided to preprocess our light field data to improve applicability of orientation analysis with structure tensors. Recall from section 3.2 that a given image patch benefits from strong homogeneous patterns within the observed region Ω. Thus we decided to employ local contrast normalization by a high-pass filtered variant of the self quotient image (SQI) normalization method presented by Wang *et al.* [17]. Our variation of the algorithm can be found in algorithm 1. The motivation for this preprocessing step is the improvement towards reflective and/or texture-less surfaces. Also very weak pattern lines that originate from small structures on transparent surfaces, shall be enhanced such that the application of the structure tensor to data with transparent objects is improved. This preprocessing is applied at the very beginning of our implementation pipeline. See fig. 3.8 for a depiction of the effects on the light field data before and after the contrast normalization.

Algorithm 1 Contrast normalization using a Gaussian filter kernel σ_{kp}

```
procedure HIGH-PASS MEDIUM-CONTRAST FILTERING($I$) ▷ $I, I_n, I_d, I_o \in \mathrm{I\!R}^{\mathrm{MxNx3xP}}$
    $I_n \leftarrow I - \sigma_{kp} * I$                                ▷ High-pass filtering
    $I_d \leftarrow (\sigma_{kp} * I_n^2)^{\frac{1}{2}} + \epsilon$               ▷ Division by 0 offset $\epsilon = 10^{-6}$
    $I_o \leftarrow \frac{I_n}{I_d}$
    $I_o \leftarrow \frac{I_o}{6} + 0.5$                                 ▷ Scaling
    return $I_o$
end procedure
```



(a) **(b)** **(c)** **(d)**

(e) Central EPI before normalization

(f) Central EPI after normalization

Figure 3.8: Effects of the proposed contrast normalization. a) shows the angular estimate without prior contrast normalization. b) shows the angular estimate with prior contrast normalization. c) depicts a grayscale input image. d) depicts the corresponding output image. e) and f) depict epipolar plane images from before and after the contrast normalization step.

From fig. 3.8 observe that the imbalanced contrast in the EPI images between the object and the background is equalized across the whole EPI. Also notice how specular regions which are heavily saturated before normalization, impose a balanced contrast after this preprocessing step. As denoted in algorithm 1, the size of the equalized local region is determined by σ_{kp}. For our application we use filter kernels that are large in the spatial directions x and y but small in the direction of s. As an example for a light field of the size $(1177 \times 1152 \times 1 \times 17)$ we use a 3D kernel with the dimensions $(9 \times 9 \times 1 \times 3)$. This exemplary selection has been made empirically. Kernel sizes may be scaled according to the dimensions of the light field. We have not conducted a test to evaluate a whole range of different kernel sizes, however we could observe, that the overall improvement is fairly constant across different scales of the normalization kernel. In fig. 3.8 the impact on the resulting estimates can be seen. Notice how the results are much smoother as opposed to the result without our contrast normalization technique. Also observe that there are far less regions where estimates are at the limits of the total angular range $\frac{\pi}{2}$ and $-\frac{\pi}{2}$.

3.5 Single Orientation Results

This section presents some results which we obtain from the single orientation structure tensor model. Recall that a general input light field is given by $L(x, y, s)$ with the dimensions $[M \times N \times 3 \times P]$ where the 3^{rd} dimension denotes the channel index, i.e. RGB. Since we compute the single order structure tensor estimate as discussed in this chapter, we obtain an angular estimate $\hat{\xi}$ for every element of the light field. Thus the resulting stack of estimates is $\Lambda_{\hat{\xi}}(x, y, s)$, with the same spatial and angular dimensions $[M \times N \times 1 \times P]$ as the input light field stack. In section 3.2.4 we discussed the confidence measure to support our model hypothesis. In the same manner as for the angular estimates, our implementation also yields a confidence result stack $C_2(x, y, s)$ with the same dimensions. In all these stacks we refer to $s_{\text{ref}} = \frac{P+1}{2}$ as the central view or reference view. Fig. 3.9 depicts the central view $L(x, y, s_{\text{ref}})$ for different datasets with corresponding estimates $\Lambda_{\hat{\xi}}(x, y, s_{\text{ref}})$.

Recall section 2.3 where we stated, that our lab data has the dimensions $[M \times N \times 6 \times P]$ where 6 channels come from two RGB light field stacks (left- and right-illumination). Naturally it follows that we obtain two results for $\Lambda_{\hat{\xi}}(x, y, s)$ and $C_2(x, y, s)$ instead of one. This will be important for our later explanation in section 5.1.1.

This concludes our discussion on the depth estimation for Lambertian surfaces. As we have shown in the course of this chapter, the regular single orientation structure tensor delivers insufficient results if subjected to transparent surfaces. In the next chapter we will investigate the extension of the basic single orientation model to models of higher order. Through this, we will show how we can improve upon the depth estimation of transparent surfaces.

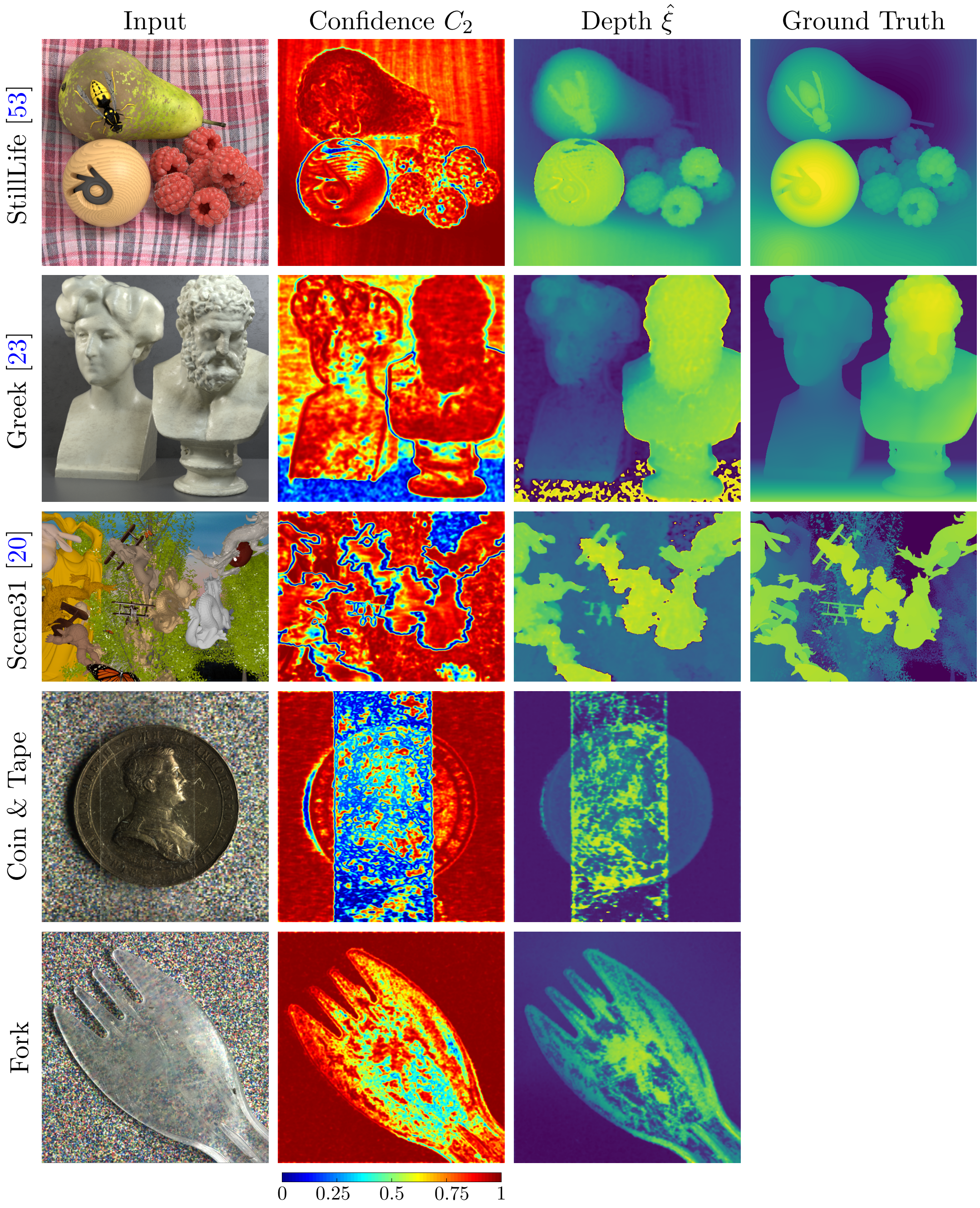


Figure 3.9: Angular depth results $\hat{\xi}$ for different data sets. The images in the first column depict the central view of a given LF $L(x, y, s_{\mathrm{ref}})$. The third column depicts the corresponding estimate $\Lambda_{\hat{\xi}}(x, y, s_{\mathrm{ref}})$ and column 4 the corresponding ground truth for all synthetic datasets. Furthermore the second column shows the confidence $C_2(x, y, s_{\mathrm{ref}})$. Notice how non-Lambertian surfaces, i.e. the tape in 'Coin & Tape' and the transparent fork in 'Fork' cause the estimate to alternate between the background and the foreground. It can be seen that the depth estimate fails in regions where the confidence C_2 is low.

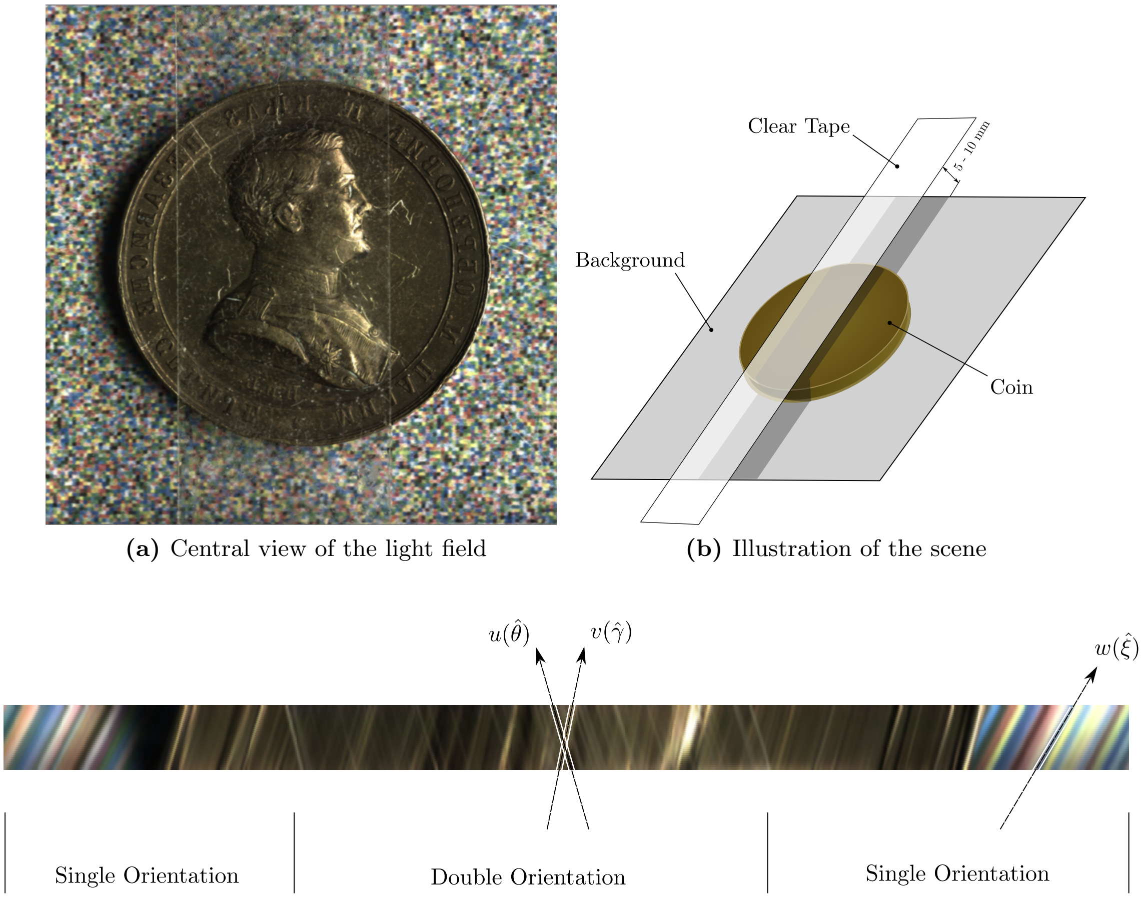
4. Depth Estimation for Transparent Surfaces

In the previous chapter we explained the use of structure tensors for the depth estimation of Lambertian surfaces. In this context we also discussed the first order model as our chosen method for this kind of task. We now want to extend this principle for scenes where part of the scene does not fulfil the Lambertian assumption, i.e. scenes with non-Lambertian surfaces. Examples for non-Lambertian surfaces are reflective or transparent surfaces. From the definition of the Lambertian assumption in chapter 3, it can be deduced, that for those surfaces the reflected radiance is non-uniform w.r.t different viewing angles. Due to the violation of this constant radiance assumption, which was the basis for our principles in the previous chapter, we have to reconsider our model. In this chapter we are going to discuss structure tensor methods that can be utilized in the case of non-Lambertian surfaces. Therefore we are going to extend the single orientation- structure tensor by introducing double orientation- structure tensors as presented by Aach *et al.* [3]. Wanner *et al.* [52] show how to apply this principle to light field data for the depth estimation of transparent and reflective surfaces. Since the reconstruction of depth for reflective surfaces is not an objective for this thesis, we are going to limit our discussions to transparent surfaces only. Note however that the principles for both types are closely related.

In the upcoming sections we will discuss the extension of the first order structure tensor to higher order models and show how this is relevant for the computation of depth estimates of transparent materials.

4.1 Transparent Materials for Light Fields

In [52] Wanner and Goldlücke describe how transparent objects in front of an opaque background impose structures in epipolar images. With the assumption of a two layered scene, i.e. a transparent surface at the front and an opaque layer behind, we want to apply a method to estimate two depth hypotheses at once. The most crucial observation made by the referenced paper is that transparent materials impose multi-orientation patterns lines in the EPI domain and that the depth estimation can be solved through multi-orientation analysis in EPI patches, as for example in fig. 4.1. Since the task of computing more than two estimates exceeds the scope of this thesis, we will hold onto the assumption of double layered scenes. One method for estimating multiple overlaid orientations is given



(a) Central view of the light field

(b) Illustration of the scene

(c) Epipolar plane image

Figure 4.1: Subfigure a) depicts the central view of a light field with a non-Lambertian (transparent) surface, i.e. clear tape. Since it is hard to comprehend a scene with transparencies, b) illustrates the scene for a better understanding. The scene consists of a strip of clear tape spanned over a coin. Subfigure c) depicts an epipolar plane image of this scene. Note how the transparent surface imposes double orientation patterns in the EPI.

by extending the previously discussed structure tensor approach to a higher order model. The derivation of the method will be given in section 4.2. This current section shall demonstrate how transparent materials influence acquired data and how we can use this to our advantage.

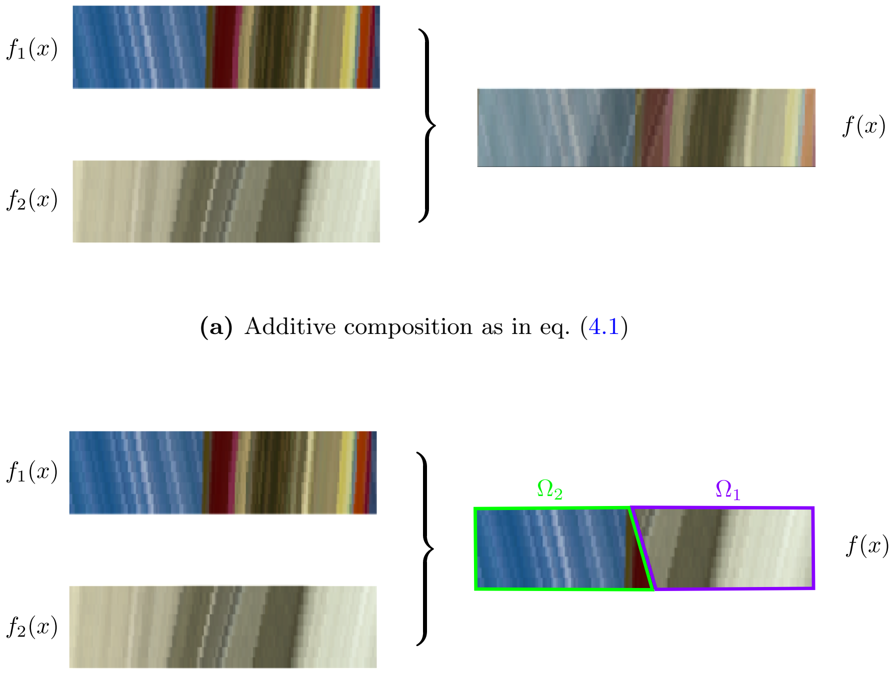
As depicted in fig. 4.2, suppose a given image patch from an epipolar image is described by a function $f(x)$ within a region Ω. Furthermore assume that the given function is a composition of signals $f_1(x)$ and $f_2(x)$ which are either additively overlaid

$$f(x) = f_1(x) + f_2(x) \qquad \forall x \in \Omega \tag{4.1}$$

or disjunct w.r.t. to some subregions Ω_1 and Ω_2 [3]

$$f(x) = \begin{cases} f_1(x) & \forall x \in \Omega_1 \\ f_2(x) & \forall x \in \Omega_2 \end{cases} \tag{4.2}$$

The composition as in eq. (4.2) is typically found in occluded regions. Occlusions occur where the visibility assumption of certain scene points is violated, i.e. these scene points can not be found in all of the P images of the LF. This kind of scenario is usually imposed by steep edges and objects close to the camera.
Since it is transparent/semi-transparent surfaces which impose an additive multi-orientation pattern in EPIs, the more interesting composition for our application is the one from eq. (4.1). Nevertheless we will discuss one structure tensor model for each case separately since both models contribute to our subsequent steps.



(a) Additive composition as in eq. (4.1)

(b) Disjunct composition as in eq. (4.2)

Figure 4.2: Composition of two image patches in an additive and disjunct manner.

With our double orientation assumption we can now observe that scenes with transparent surfaces impose additively overlaid double-orientation-patterns in EPIs. This is illustrated in fig. 4.1 c). Instead of one orientation $w(\hat{\xi})$, two orientations $u(\hat{\theta})$ and $v(\hat{\gamma})$ are present. As mentioned earlier, we will discuss double orientation analysis from EPI patches in the upcoming sections. Thus we will show how to compute two orientations $u(\hat{\theta})$ and $v(\hat{\gamma})$ for regions Ω through double orientation structure tensors.

4.2 Double-orientation Analysis

This section will give the mathematical derivation and intuition of structure tensor models for the angular estimation of double orientation patterns. Again our derivations are based on [3] and will be extended in multiple ways.

4.2.1 Second Order Double Orientation Structure Tensor (SODOST)

First we are going to focus on the case of additively overlaid signals as in fig. 4.2. This specific case can be tackled through a second order double orientation structure tensor model (SODOST). Assume $x \in \mathbb{R}^2$ again denotes the bivariate image coordinates. Given a double-orientation image patch $f : \mathbb{R}^2 \to \mathbb{R}$ within a region Ω and the assumption from eq. (4.1), then it follows that a pattern can be separated into two single orientation patches $f_1(x)$ and $f_2(x)$ with orientations $u(\theta)$ and $v(\gamma)$. For these single orientation patches thus the following conditions must hold true

$$\begin{aligned}\frac{\partial f_1(x)}{\partial u(\theta)} &= u(\theta)^{\mathrm{T}} \cdot \nabla f_1(x) = 0 \qquad \forall x \in \Omega \\ \frac{\partial f_2(x)}{\partial v(\gamma)} &= v(\gamma)^{\mathrm{T}} \cdot \nabla f_2(x) = 0 \qquad \forall x \in \Omega\end{aligned} \tag{4.3}$$

From these conditions follows the transparency constraint [42]. The proof is given in section A.1.

$$\frac{\partial^2 f(x)}{\partial u(\theta)\partial v(\gamma)} = 0 \qquad \forall x \in \Omega \tag{4.4}$$

Recall from eq. (3.2) the identities for the operators

$$\begin{aligned}\left(\frac{\partial}{\partial u(\theta)}\right) &= \cos(\theta)\left(\frac{\partial}{\partial y}\right) + \sin(\theta)\left(\frac{\partial}{\partial s}\right) \\ \left(\frac{\partial}{\partial v(\gamma)}\right) &= \cos(\gamma)\left(\frac{\partial}{\partial y}\right) + \sin(\gamma)\left(\frac{\partial}{\partial s}\right)\end{aligned} \tag{4.5}$$

With these identities we can denote

$$\begin{aligned}&\left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right) f(x) = \\ &= \left[\cos(\theta)\left(\frac{\partial}{\partial y}\right) + \sin(\theta)\left(\frac{\partial}{\partial s}\right)\right]\left[\cos(\gamma)\left(\frac{\partial}{\partial y}\right) + \sin(\gamma)\left(\frac{\partial}{\partial s}\right)\right] f(x) \\ &= \left\{\cos(\theta)\cos(\gamma)\left(\frac{\partial^2}{\partial y^2}\right) + [\sin(\theta)\cos(\gamma) + \sin(\gamma)\cos(\theta)]\left(\frac{\partial^2}{\partial y \partial s}\right) + \sin(\theta)\sin(\gamma)\left(\frac{\partial^2}{\partial s^2}\right)\right\} f(x) \\ &= m_2(\theta,\gamma)^{\mathrm{T}} \cdot \delta f(x) = 0\end{aligned}$$

where $m_2(\theta,\gamma)^{\mathrm{T}} = \left[\cos(\theta)\cos(\gamma), \quad \sin(\theta)\cos(\gamma) + \sin(\gamma)\cos(\theta), \quad \sin(\theta)\sin(\gamma)\right]^{\mathrm{T}}$ denotes the mixed-orientation-parameter (MOP) vector [3] and $\delta f(x) = \left[\frac{\partial^2}{\partial y^2}, \quad \frac{\partial^2}{\partial y \partial s}, \quad \frac{\partial^2}{\partial s^2}\right]^{\mathrm{T}}$. In analogy to eq. (3.6) the constraint is equivalent to the energy term

$$
\begin{aligned}
E_2(m_2(\theta,\gamma)) &= \int_\Omega \left(m_2(\theta,\gamma)^{\mathrm{T}} \cdot \delta f(x)\right)^2 d\Omega = \\
&= m_2(\theta,\gamma)^{\mathrm{T}} \left(\int_\Omega (\delta f(x))(\delta f(x))^{\mathrm{T}} d\Omega \right) m_2(\theta,\gamma) = \\
&= m_2(\theta,\gamma)^{\mathrm{T}} \cdot T \cdot m_2(\theta,\gamma) = 0
\end{aligned}
\tag{4.6}
$$

From this equation follows the definition of the second order double orientation structure tensor (SODOST)

$$
T = \int_\Omega G_{\sigma_o} * \begin{bmatrix} \hat{d}_{yy}^2 & \hat{d}_{yy}\hat{d}_{ys} & \hat{d}_{yy}\hat{d}_{ss} \\ \hat{d}_{yy}\hat{d}_{ys} & \hat{d}_{ys}^2 & \hat{d}_{yy}\hat{d}_{ys} \\ \hat{d}_{yy}\hat{d}_{ss} & \hat{d}_{yy}\hat{d}_{ys} & \hat{d}_{ss}^2 \end{bmatrix} d\Omega = \int_\Omega \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{12} & D_{22} & D_{23} \\ D_{13} & D_{23} & D_{33} \end{bmatrix} d\Omega. \tag{4.7}
$$

The structure tensor is a symmetric construct comprised of second order gradients filtered on an inner scale σ_i, hence the 'second order' prefix. Regarding the notation, $\hat{d}_{yy} = \sigma_i * \frac{\partial^2 f(x)}{\partial y^2}$ denotes the filtered second order gradient in y. The same applies for $\hat{d}_{ss}$ in s-direction. Equivalently $\hat{d}_{sy} = \hat{d}_{ys} = \sigma_i * \frac{\partial^2 f(x)}{\partial y \partial s}$ denotes the filtered y-gradient of the gradient in s direction of $f(x)$ and vice versa. G_{σ_o} defines the weighting function of T on an outer scale.

There is also a double orientation structure tensor based on first order derivatives which will be discussed in section 4.2.2.

4.2.2 First Order Double Orientation Structure Tensor (FODOST)

For the double orientation analysis of pattern patches we previously assumed that both sub-images $f_1(x)$ and $f_2(x)$ are additively overlaid, i.e. see eq. (4.1). The second case focuses on the composition of $f_1(x)$ and $f_2(x)$ on disjunct subregions Ω_1 and Ω_2. Note that $\Omega = \Omega_1 \cup \Omega_2$. If we assume that the orientations of the two disjunct regions are given by $r(\theta)$ and $t(\gamma)$, then we can denote

$$
\frac{\partial f_1(x)}{\partial r(\theta)} \frac{\partial f_2(x)}{\partial t(\gamma)} = 0 \tag{4.8}
$$

Following a similar derivation as in section 4.2.1, this then leads to the energy term

$$
\begin{aligned}
E_3(m_2(\theta,\gamma)) &= \int_\Omega \left(m_2(\theta,\gamma)^{\mathrm{T}} \cdot \delta_0 f(x)\right)^2 d\Omega = \\
&= m_2(\theta,\gamma)^{\mathrm{T}} \left(\int_\Omega (\delta_0 f(x))(\delta_0 f(x))^{\mathrm{T}} d\Omega \right) m_2(\theta,\gamma) = \\
&= m_2(\theta,\gamma)^{\mathrm{T}} \cdot T_f \cdot m_2(\theta,\gamma) = 0
\end{aligned}
\tag{4.9}
$$

where $\delta_0 f(x) = \left[\left(\frac{\partial}{\partial y}\right)^2, \; \frac{\partial}{\partial y}, \; \frac{\partial}{\partial s}, \; \left(\frac{\partial}{\partial s}\right)^2 \right]^{\mathrm{T}}$ and $m_2(\theta,\gamma)$ denotes the previously introduced MOP vector. Since $\delta_0 f(x)$ is comprised of first order derivatives only, the definition of the first order double orientation structure tensor (FODOST) is given by

$$
T_f = \int_\Omega G_{\sigma_o} * \begin{bmatrix} \hat{d}_y^4 & \hat{d}_y^3 \hat{d}_s & \hat{d}_y^2 \hat{d}_s^2 \\ \hat{d}_y^3 \hat{d}_s & \hat{d}_y^2 \hat{d}_s^2 & \hat{d}_y \hat{d}_s^3 \\ \hat{d}_y^2 \hat{d}_s^2 & \hat{d}_y \hat{d}_s^3 & \hat{d}_s^4 \end{bmatrix} d\Omega \tag{4.10}
$$

Similar to its second order counterpart in eq. (4.7), this structure tensor model can be used to estimate two orientations, however, with the difference that $f_1(x)$ and $f_2(x)$ are define on disjunct subregions, which for example is the case for occlusions. To avoid confusion with the previous model, we denote the orientations of this model $r(\theta)$ and $t(\gamma)$. As the FODOST model is less relevant for our applications, we will continue to reference the SODOST model as our default double orientation model. Note however that both variants are interchangeable since both are symmetric 3×3 constructs. In the upcoming section we will discuss how to compute both orientation of $f(x)$ from the double orientation structure tensor models.

4.2.3 MOP- and Orientation-Computation

Recall that we want to find two orientations with angles θ and γ such that the eq. (4.6) and eq. (4.9) are fulfilled. This can only be the case if we assume perfectly noise free, double-oriented image patches in a region Ω. However, in reference to the single orientation analysis in section 3.2 we already stated that this is practically never the case for real world data. Thus we again define a constrained optimization problem as in eq. (3.6).

$$
\underset{m_2(\theta,\gamma)}{\operatorname{argmin}} \;\; m_2(\theta,\gamma)^{\mathrm{T}} T \; m_2(\theta,\gamma) \quad s.t. \quad m_2(\theta,\gamma)^{\mathrm{T}} \cdot m_2(\theta,\gamma) = 1 \tag{4.11}
$$

Further we can derive the solution to this optimization problem similar to section 3.2.1 and obtain the right hand side eigenvector notation

$$T\ m_2(\theta,\gamma) = \lambda m_2(\theta,\gamma) \qquad m_2(\theta,\gamma)^{\mathrm{T}} \cdot m_2(\theta,\gamma) = 1 \tag{4.12}$$

Let $\lambda_1 > \lambda_2 > \lambda_3$ denote the eigenvalues of the structure tensor T and let v_1, v_2 and v_3 be the corresponding eigenvectors, then it can easily be seen that eq. (4.11), due to the constraint on the MOP vector, is minimal if $m_2(\theta,\gamma) = v_3$. Thus we can find the MOP vector $m_2(\theta,\gamma)$ by performing an eigenvalue analysis on the double orientation structure tensor.

Since the double orientation models are based on 3×3 matrices, the closed form solution for the eigenvector v_3 is impractical. However due to the fact that the tensors are symmetric, it is possible to simplify and parallelize the computation for whole light fields. Based on the principle in [43], Wanner *et al.* [52] present an algorithm in their auxiliary document (additional material).

Once the MOP vector has been computed, it is possible to compute the orientations $u(\theta) = [\cos(\theta)\ \ \sin(\theta)]^{\mathrm{T}}$ and $v(\gamma) = [\cos(\gamma)\ \ \sin(\gamma)]^{\mathrm{T}}$. To do this, Wanner *et al.* propose to construct another 2×2 construct which again can be solved through an eigenvalue analysis. However, this step can equivalently be achieved by solving a simple root finding problem [3]. Therefore we set a polynomial

$$z^2 - m_2^{(2)} z + m_2^{(1)} m_2^{(3)} = 0. \tag{4.13}$$

where $m_2^{(i)}$ denotes the i-th element of the MOP vector. With the roots of the polynomial it is possible to compute θ and γ through

$$\tan(\theta) = \frac{z_1}{m_2^{(1)}}, \qquad \tan(\gamma) = \frac{z_2}{m_2^{(1)}} \tag{4.14}$$

The proof for this is provided in section A.2. Note that the orientations are given by

$$u(\theta) = \begin{bmatrix} m_2^{(1)} \\ z_1 \end{bmatrix}, \qquad v(\gamma) = \begin{bmatrix} m_2^{(1)} \\ z_2 \end{bmatrix} \tag{4.15}$$

To map the angles to agree with our angular notation from section 3.1, we first correct both orientations by flipping them if the sign of the second component is non-positive

$$u(\hat{\theta}) = u(\theta)\,\mathrm{sign}(z_1), \qquad v(\hat{\gamma}) = v(\gamma)\,\mathrm{sign}(z_2) \tag{4.16}$$

In the same manner as for the single orientation case we then can compute the angles directly through

$$\hat{\theta} = tan^{-1}\left(\frac{u(\hat{\theta})^{(2)}}{u(\hat{\theta})^{(1)}}\right) - \frac{\pi}{2}, \qquad \hat{\gamma} = tan^{-1}\left(\frac{v(\hat{\gamma})^{(2)}}{v(\hat{\gamma})^{(1)}}\right) - \frac{\pi}{2} \tag{4.17}$$

This shows that we can compute two orientations by performing an eigenvalue analysis followed by a root finding problem. Both of these steps are implemented in a parallel GPU implementation, which is a must for a reasonable runtime even at moderate light field sizes.

4.2.4 Double Orientation Confidence

In section 3.2.4 we discussed confidence measure for single orientation estimates. Since we want to bring double orientation estimates in relation to single orientation estimates, our goal is to find a confidence measure based on metrics from the double orientation structure tensors that allows comparability with the aforementioned coherence of the structure tensor S, i.e. eq. (3.20). Aach *et al.* [3] propose to use a confidence measure that is an extension of their single orientation confidence (coherence)

$$0 < C_3 = \frac{(\lambda_1\lambda_2\lambda_3)^2}{(\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_1\lambda_3)^2} \leq \frac{1}{27} \tag{4.18}$$

However, in practice this confidence measure doesn't scale well over the full range $0 < C_3 \leq \frac{1}{27}$. Even if scaled to the interval $[0, 1]$, this is not an expressive confidence measure and hardly comparable to C_2.

Instead we decided to present our own confidence estimate based on the following thoughts. Opposed to the single orientation case it is hard to interpret the eigenvectors and eigenvalues from the double orientation models in the sense of geometric intuition of the pattern. From our empiric observations we found that the relations between eigenvalues correspond to certain properties of the image patch in Ω. Assume the eigenvalues of the double orientation structure tensors to be given by $\lambda_1, \lambda_2, \lambda_3$ subject to $\lambda_1 > \lambda_2 > \lambda_3$.

Pattern class	Double Orientation Eigenvalues
Homogeneous	$\lambda_1 \approx 0, \quad \lambda_2 \approx 0, \quad \lambda_3 \approx 0$
Single-Orientation	$\lambda_1 \gg \lambda_2 \approx \lambda_3, \quad \lambda_2 \approx 0, \quad \lambda_3 \approx 0$
Double-Orientation	$\lambda_1 \approx \lambda_2 \gg \lambda_3, \quad \lambda_3 \approx 0$
N-Orientation/Noise	$\lambda_1 \gg 0, \quad \lambda_2 \gg 0, \quad \lambda_3 \gg 0$

Table 4.1: Observed relations of eigenvalues for different pattern classes. The pattern 'N-Orientation/Noise' denotes all multi-orientation pattern patches with more than two orientations.

In case of a homogeneous, structureless pattern patch we would find that the gradients are very weak, thus resulting in eigenvalues that are close to 0. If we apply the second order structure tensor analysis to a pattern patch with only one orientation, we can observe that the two smaller eigenvalues λ_2 and λ_3 are small relative to the largest eigenvalue λ_1. When adding one orientation, we obtain the double orientation case where the two larger eigenvalues λ_1 and λ_2 are much larger than the smallest eigenvalue λ_3. In section 4.2.3 we stated that for double orientation models the residual energy for eq. (4.11) at the optimal solution is equivalent to the smallest eigenvalue λ_3. Naturally it follows that for highly confident double orientation results we want λ_3 to be low. See table 4.1 for a comparison of different pattern classes. Because of these conditions for high confidence in the double-orientation models and the fact that $\lambda_1 > \lambda_2 > \lambda_3 \geq 0$ we define a new confidence measure

$$C_4 = \left(\frac{\lambda_1 - \lambda_3}{\lambda_1 + \lambda_3}\right)\left(1 - \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right) \tag{4.19}$$

The first term in C_4 gives a high confidence if the difference between λ_1 and λ_3 is large. We also want both larger eigenvalues λ_1 and λ_2 to be large and close to each other. If this is the case, the second term of C_4 shall contribute to a high confidence. Note that $C_4 \in [0, 1]$. From our evaluations we have found, that the utilization of the full range is much improved compared to C_3. We also found that this confidence measure is very well suited for comparing it with the single orientation confidence C_2. To prove this, we computed the correlation between the inverse of the single orientation confidence $(1 - C_2)$ and our confidence measure C_4 according to

$$r = \frac{\sum_x \sum_y (C_4(x,y) - \overline{C_4})((1 - C_2(x,y)) - \overline{(1 - C_2)})}{\sqrt{\left(\sum_x \sum_y C_4(x,y) - \overline{C_4}\right)^2 \left(\sum_x \sum_y (1 - C_2(x,y)) - \overline{(1 - C_2)}\right)^2}}$$

and found a correlation of 94% to 97% for the given confidence images in fig. 4.3. All expressions of the form $\overline{x}$ denote the mean over the whole image.

Going back to the interpretation of the eigenvalues, there is still one case that hasn't been discussed. In case all three eigenvalues λ_1, λ_2 and λ_3 are large we can deduce that our double orientation hypothesis is too simple. Such cases imply an image patch with more than two orientations. This can also be the case if high amounts of noise are present.

With the findings regarding the confidence in the double orientation models SODOST and FODOST we conclude this section. We highlighted different confidence measures to indicate weakly supported estimates and presented a confidence measure that delivers high confidence for double orientation patterns. We nevertheless want to know what to expect from our models when we apply them to unsuitable pattern patches. Thus we will give an evaluation on unfitting model assumptions in the next section.

4.3 Estimates for Unfitting Model Assumptions

This section will highlight how estimates from structure tensor models of different order perform if subjected to image patches $f(x)$ with unfitting pattern classes. For this discussion we exclude trivial cases such as homogeneous image patches since the orientation in

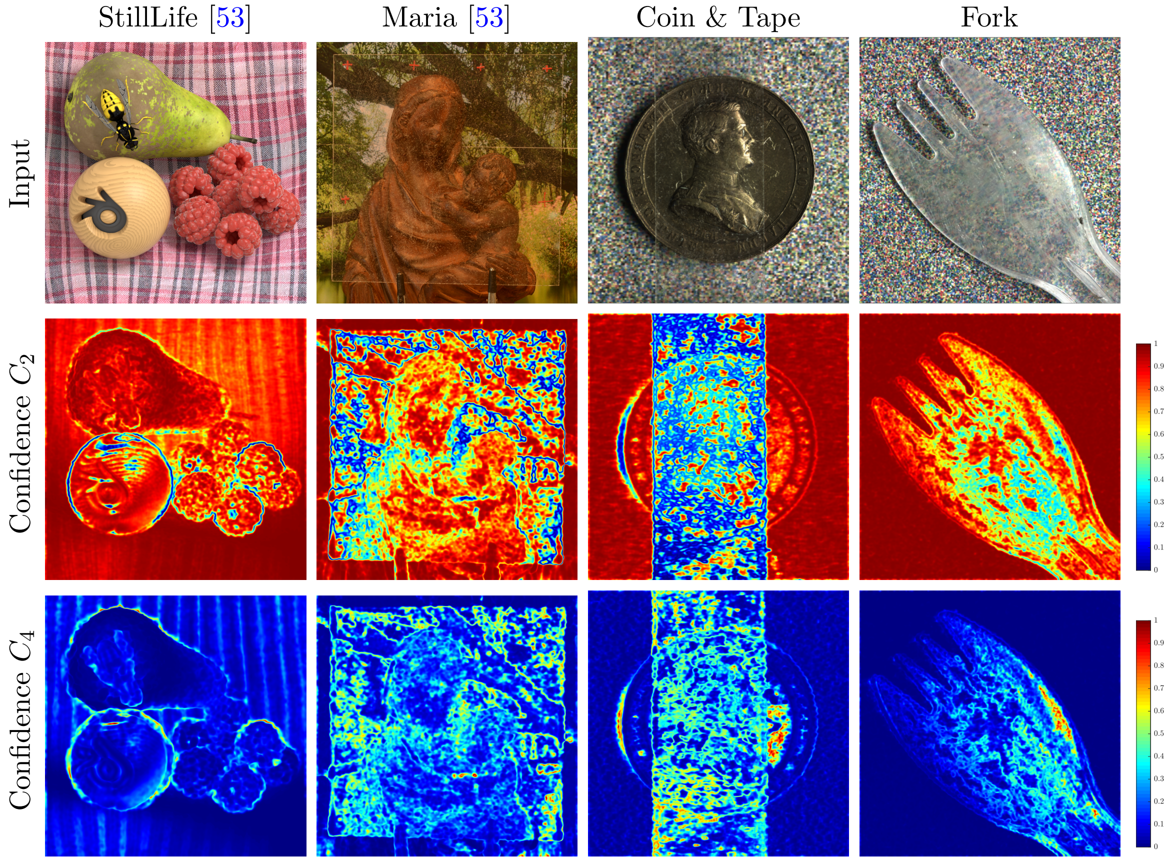


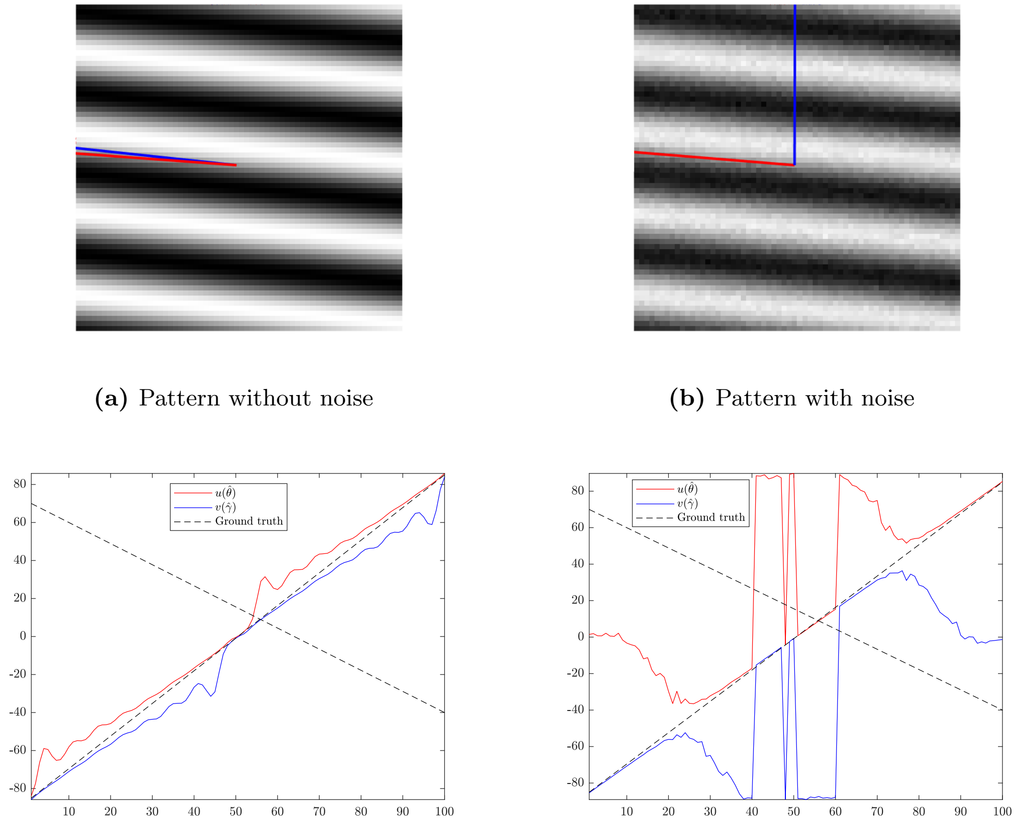
Figure 4.3: Result of the confidence measures C_2 and C_4 for different data sets. The images in the first row depict the central view of a given LF. The second row depicts the corresponding confidences C_2. Likewise the third row depicts C_4. It can be observed, that the double-orientation hypothesis is highly supported in areas where the single orientation confidence C_2 is low. This implies that areas with two EPI orientations, i.e. areas with transparencies or occlusions, lead to a high confidence in the double orientation hypothesis C_4.

such cases is determined by the noise.

4.3.1 Unfitting Double Orientation Hypothesis

As the first case assume we compute the double orientation estimates $u(\hat{\theta})$ and $v(\hat{\gamma})$ in a local neighbourhood Ω with only one pattern orientation. Therefore we reused the framework described in section 3.2.5 and simplified it to simple 2D pattern patches. Similar to this framework, we set up synthetic image patches and computed the double orientation estimates for the interval $[-85°, 85°]$. Note that the estimates are sorted $u(\hat{\theta}) > v(\hat{\gamma})$, following our assumption of having one estimate for the front layer and one for the back layer. In fig. 4.4 the results over the whole interval and exemplary pattern patches with an indication of the estimates can be seen.

The figure shows that for a noise free image patch, both estimates follow the ground truth angle well. However, once noise is introduced it can be observed that both estimates start to alternately follow the correct value. One of both estimates follows the second strongest orientation in the patch which is given by the noise. Note the high sensitivity to noise



(a) Pattern without noise

(b) Pattern with noise

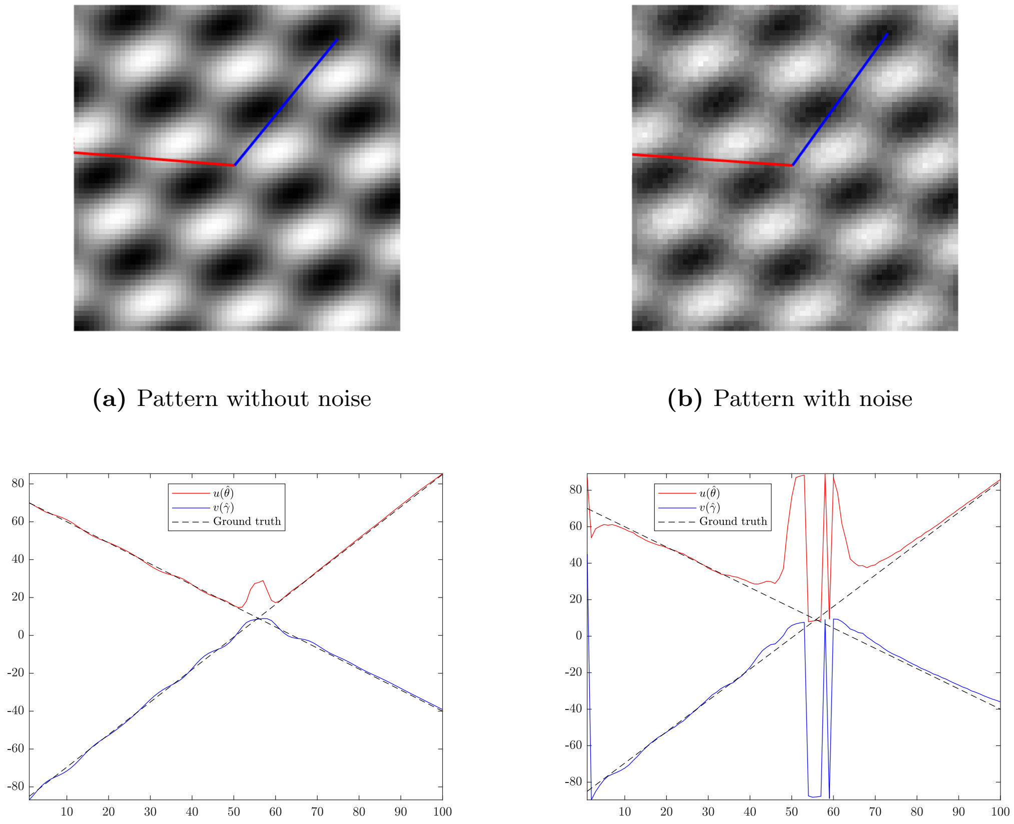
(c) Result without noise

(d) Result with noise

Figure 4.4: a) and b) show pattern patches with both orientations $u(\hat{\theta})$ and $v(\hat{\gamma})$ in colour. The sub-figures c) and d) depict the estimates compared to the ground truth angle from which the pattern patch has been synthesized.

since only a small amount is sufficient to induce this effect.
Once a second orientation is added, we obtain composite pattern patches as depicted in fig. 4.5. This added second orientation is synthesized for angles in the interval $[-40°, 70°]$ and is evaluated in descending order.

The results show that in the case of two strong orientations in a patch, the double orientation hypothesis proves to be very reliable. As expected, due to the correct model hypothesis, noise has much less of an impact on the analysis. Since we sweep through two angular intervals in opposing directions, we get to a point where both orientations align, which is at $\approx 10°$. In this case we obtain a single orientation image patch. Regarding the estimates we can observe that one is close to the ground truth and the other one is determined by the noise. Once this alignment occurs, noise has a much larger impact because the noise determines the orientation of the second estimate.
To demonstrate the effects if the pattern patch has more than two orientations, we tested three orientation patterns as depicted in fig. 4.6. Therefore we overlaid another pattern patch with angles in the interval $[-10°, 10°]$.



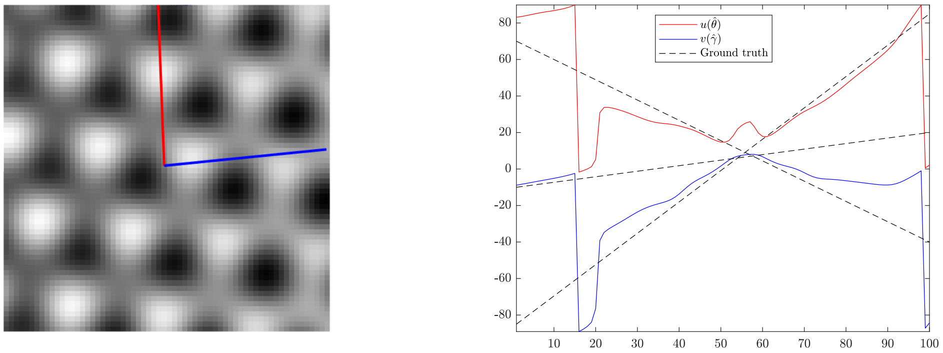
(a) Pattern without noise

(b) Pattern with noise

(c) Result without noise

(d) Result with noise

Figure 4.5: a) and b) show pattern patches with both orientations $u(\hat{\theta})$ and $v(\hat{\gamma})$ in colour. The sub-figures c) and d) depict the estimates compared to the ground truth angles from which the pattern patch has been synthesized.



(a) Pattern without noise

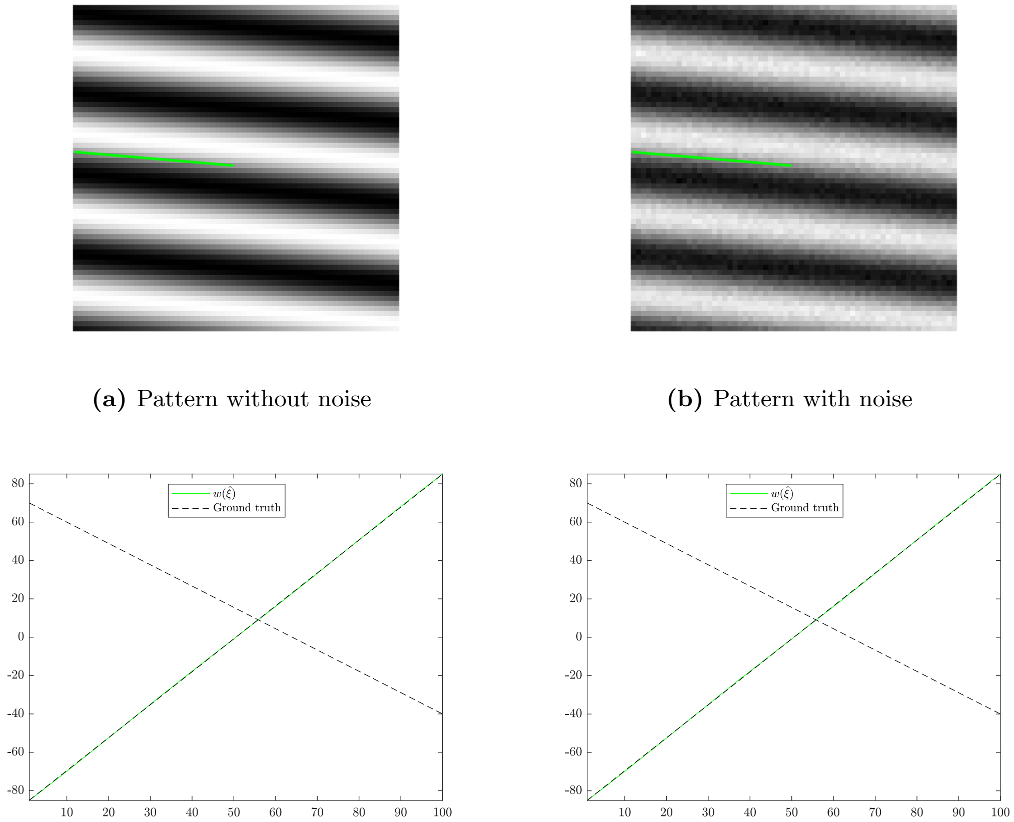
(b) Result without noise

Figure 4.6: a) shows a triple-orientation pattern patch with the double orientation estimates $u(\hat{\theta})$ and $v(\hat{\gamma})$ in colour. The sub-figure b) depicts the estimates compared to the ground truth angles from which the pattern patch has been synthesized.

From the results on the triple orientation patch it can be seen, that since all orientations intersect at $\approx 10°$, the pattern briefly reduces to the single orientation case. However overall the model assumption is too simple to cope with more than two orientations.

4.3.2 Unfitting Single Orientation Hypothesis

In section 4.3.1 we discussed the outcome of the double orientation analysis on different types of pattern patches. In this section we are going to do the same but with the assumption of a single orientation. The results for the first case, based on the orientation analysis from the aforementioned framework, can be seen in fig. 4.7.



(a) Pattern without noise

(b) Pattern with noise

(c) Result without noise

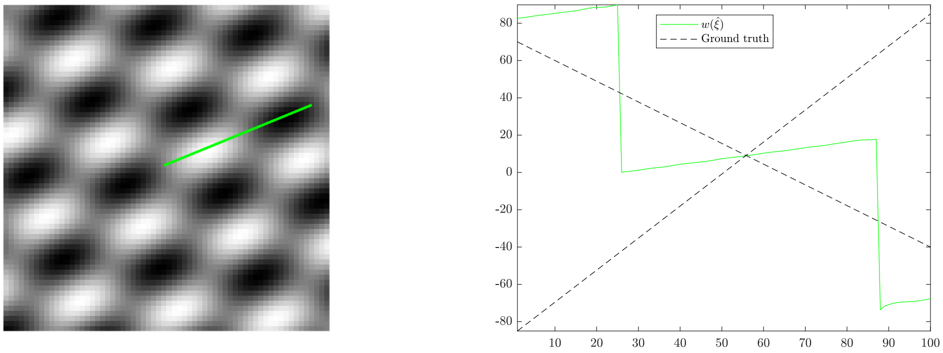
(d) Result with noise

Figure 4.7: a) and b) show single orientation pattern patches with a single orientation estimate $w(\hat{\xi})$ in colour. The sub-figures c) and d) depict the estimate compared to the ground truth angle from which the pattern patch has been synthesized.

What can be noticed right away is that the estimate $w(\hat{\xi})$ follows the ground truth value very closely. Noise does not have as much of a significant effect on the result as compared to the double orientation hypothesis. This also applies for higher levels of noise. However, due to the fact that the single orientation structure tensor S in eq. (3.5) is based on first order derivatives and not second order derivatives, this observation is expected.

Moving onward, we subject double orientation patches to a single orientation analysis.

The results are depicted in fig. 4.8. Due to the low sensitivity to noise, this figure only depicts the noise-free double orientation case.



(a) Pattern without noise

(b) Result without noise

Figure 4.8: Sub-figure a) shows a double orientation pattern patch with a single orientation estimate $w(\xi)$ in colour. The sub-figure b) depicts the estimate compared to the ground truth angle from which the pattern patch has been synthesized.

The results show that the single orientation estimate $w(\hat{\xi})$ follows the most prominent orientation of the pattern. Up to this point, the magnitudes of both single orientation patterns $f_1(x)$ and $f_2(x)$ were set equal. Because of this balance, both orientation patterns interfere in a way such that they form an averaged orientation except when both sub-patterns align. The subplot *b*) in fig. 4.8 also shows discontinuities in the resulting curve for the estimate $w(\hat{\xi})$. At exactly these points the pattern is synthesized from two orientations which are orthogonal to each other. This causes the estimate to flip by 90°, i.e. reorienting to the most prominent orientation.
The behaviour changes if the individual pattern magnitudes are different. The depiction in fig. 4.9 shows that if we increase the magnitude of one sub-signal relative to the other, then the single orientation estimate will strictly follow this prominent orientation. This observation is important since it leads to the typical alternating behaviour between foreground and background. In the next section we will briefly discuss how the findings from this section apply to real data.

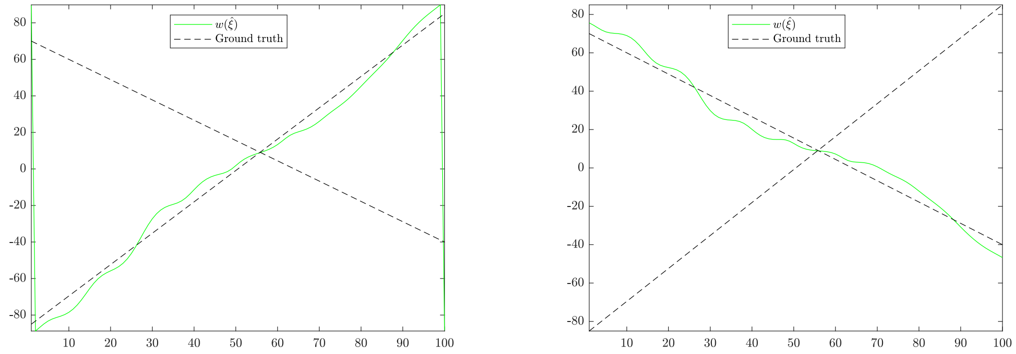
4.3.3 Impact on Real Data

Now that we have discussed the findings from within this restricted framework, we will briefly evaluate the meaning for whole data sets. For this reason fig. 4.10 depicts the single- and double-orientation estimates (SODOST) for a scene with a transparent object.

The order in which we will point out the effects of unfitting model assumption is the same as presented in the first two parts of this section.

Double Orientation Model on Single Orientation Data

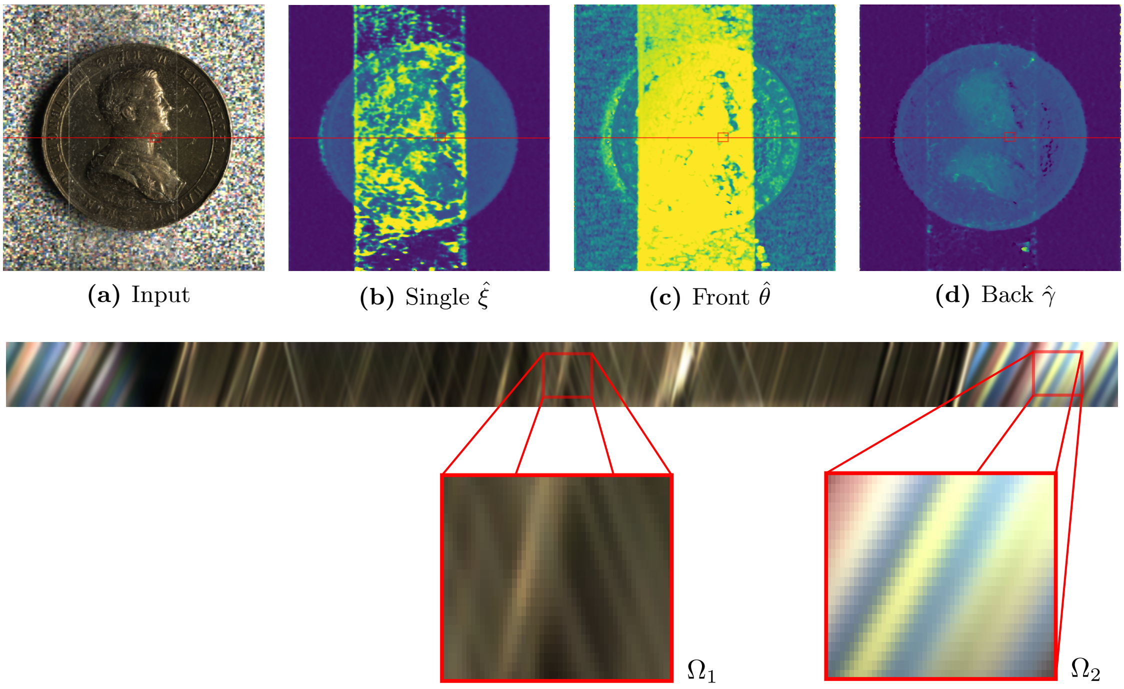
In the case that a double orientation model is applied to a local region where only



(a) $\hat{f}_1(x) > \hat{f}_2(x)$

(b) $\hat{f}_2(x) > \hat{f}_1(x)$

Figure 4.9: Both sub-figures show the resulting estimates against the ground truth of the double orientation pattern. In a) the magnitude of the first single orientation signal $\hat{f}_1(x)$ is larger than $\hat{f}_2(x)$. For b) this relation is reversed. Notice how the estimate follows the ground truth of the more prominent signal.



(a) Input

(b) Single $\hat{\xi}$

(c) Front $\hat{\theta}$

(d) Back $\hat{\gamma}$

(e) EPI + Patch

Figure 4.10: Example based on the previously introduced 'Coin & Tape' dataset. a) shows the central image form the input light field. The epipolar plane image depicted in e) is located at the indicated red line. b) depicts the estimate form the single orientation model. c) and d) show the front- and background estimate of the double orientation model (SODOST). In e) an EPI with the indication of a double orientation image patch Ω_1 and a single orientation image patch Ω_2 can be seen. The estimates computed from Ω_1 are depicted in b) c) and d) by the square marker.

one orientation is present, i.e. as in Ω_2, we can observe that one angular estimate is correct and the other one falsely estimates effects from the noise. This can be seen in background areas around the coin for example. Notice how the estimate for the foreground $\hat{\theta}$ is very noisy in these local regions.

Double Orientation Model on Double Orientation Data

In regions where the transparent surface imposes a second orientation on the EPI data, e.g. see Ω_1, the double orientation estimates $\hat{\theta}$ and $\hat{\gamma}$ estimate the foreground and background correctly. In the case of the foreground, this corresponds to a point on the surface of the tape. For the background, this corresponds to a point on the coin.

Single Orientation Model on Single Orientation Data

Examine the single orientation patch Ω_2. In this region the single orientation estimate $\hat{\xi}$ correctly estimates the depth corresponding to the background. Notice also how the background estimate from the double orientation model agrees with $\hat{\xi}$.

Single Orientation Model on Double Orientation Data

If we apply the single orientation model to a region with two orientations as in Ω_1, then we can observe that the estimate $\hat{\xi}$ alternately jumps between the foreground and the background. This behaviour has already been observed in section 4.3.2 and confirms our findings from our experimental framework. Depending on the magnitude of each respective orientation, the estimate will cling on to either the front or the back.

4.4 Double Orientation Results

In this section we are going to discuss results from the double orientation structure tensor models. For comparison we also included results from the single orientation model. An extensive set of results is depicted in fig. 4.11. As mentioned in the result section 3.5 of the single orientation case, each depicted image corresponds to the central view s_{ref} of the result stack. For the single orientation estimates this was denoted $\Lambda_{\hat{\xi}}(x, y, s_{\text{ref}})$. Similarly, the result stacks for the second order double orientation model are denoted $\Phi_{\hat{\theta}}(x, y, s)$ and $\Phi_{\hat{\gamma}}(x, y, s)$ for the front and back. Moreover $\Psi_{\hat{\theta}}(x, y, s)$ and $\Psi_{\hat{\gamma}}(x, y, s)$ equally denote result stacks for the front and back estimates but for the FODOST model.

The first depicted data set has been added for comparison with the previous results. It is the only dataset in this result section that does not include a non-Lambertian surface, i.e. the whole scene is opaque. Neither does this scene show reflections, nor does it contain transparent surfaces. The double layer assumption for this dataset is violated since there is no second layer which could impose double orientation information in the EPI domain. This can be seen in the results from the second order model. The FODOST model is mostly similar to the single orientation estimate. However, in occluded areas these double orientation estimates can resolve the occlusions implicitly by delivering estimates for pattern orientations from the occluded as well as the occluding scene points.

The most interesting results are given by the datasets including transparencies. As men-

tioned before, the 'Maria' dataset is the only transparency-dataset which could be found from external sources. Notice how for all of these datasets the single orientation estimate is an alternating composition of estimates between the foreground and the background. The second order double orientation model seems to separate both layers well. The transparent object is well estimated by $\Phi_{\hat{\theta}}(x, y, s)$ and the scene behind by $\Phi_{\hat{\gamma}}(x, y, s)$. Notice if a certain region from the transparent surface is also present in the background estimate $\Phi_{\hat{\gamma}}(x, y, s)$, then this structure occludes the background sufficiently enough such that the background estimate can no longer be determined by the double orientation model. It can also be seen that the estimate for the foreground delivers noisy results for single orientation (i.e. opaque) surfaces. For the last example, the first order double orientation model mostly resembles the same picture as in the completely opaque scene. However, the estimate for the front $\Psi_{\hat{\theta}}(x, y, s)$ tends to estimate more of the transparent surface, whereas the background estimate $\Psi_{\hat{\gamma}}(x, y, s)$ stays closer to the background.
Concluding with the investigation of the structure tensor approach, we now have 5×2 model based estimates (from left-and right-illumination), which can correctly estimate depth in subregions with an adequate hypothesis but are invalid elsewhere. According to our double layer assumption, we want to obtain two final images which are identical in regions with an opaque surface and distinct where the Lambertian assumption does not hold. For the later case we want estimates for the transparent surface and estimates for what might be underneath. To achieve this, we will present a method to combine all structure tensor estimates in the upcoming chapter.

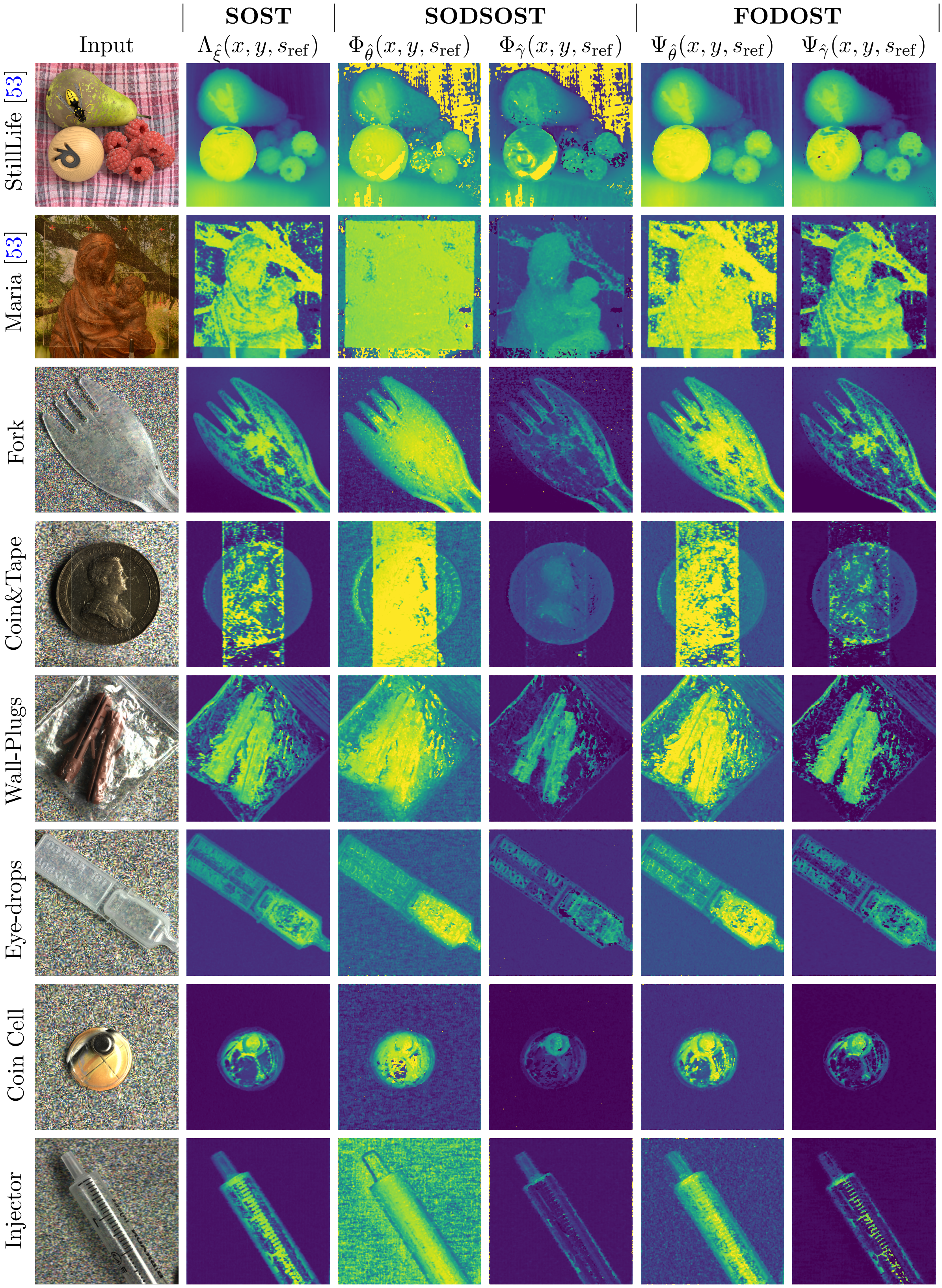


Figure 4.11: Depth estimates from the present structure tensor models (SOST, SODOST, FODOST) for a variety of different data sets. A description for all datasets that have been acquired by us is given in appendix B.

5. Double Layer Refinement

Up to this point of this thesis, we gave an introduction to the idea behind depth estimation from passive multi-view imaging and thoroughly discussed the principles of structure tensor methods in conjunction with light field data. We also gave a detailed insight of this approach in the context of different types of surfaces. Based on the principle of different structure tensor models, we showed how estimates for mixed scenes can be computed by performing orientation analysis in the EPI domain. In section 4.3 we showed that various model hypotheses only give a valid result in regions with a suitable pattern type. Recalling our double layer assumption, i.e. a front- and a back-layer, which unfortunately is not applicable everywhere, we require a reliable method to combine all depth estimates such that we obtain two surfaces in scene space. We therefore introduce two variants of a projection method which allows us to obtain exactly two surfaces. Both of these variants are based on a voting scheme to refine the prior computed structure tensor estimates. The principles and details regarding this method will be explained in the upcoming sections.

5.1 Voxel Volume Projection

In this section we are going to discuss our method for combining multiple depth estimates from different models. Recall that we had a single- and two double-orientation models for estimating depth from a light field description (SOST, SODOST, FODOST). In section 4.3.3 we found, that models of different order can estimate the same surface, albeit in an unpredictable fashion. The single orientation structure tensor for example computes a depth estimate for the scene point that imposes the most prominent direction in an EPI. The SOST model thus gives a single, high confident depth estimate corresponding to either the foreground or the background. Double orientation models (SODOST, FODOST) always give us two estimates, one for a possible background and one for a possible foreground. However if the double layer assumption is inapplicable, at least one of the estimates is incorrect. Thus it is hard to tell which model delivers a valid estimate for each point of a scene. We introduced confidence measures for each model, however implementing a point-wise model selection based on the single orientation confidence C_2 and the double orientation confidence C_4 gives unsatisfying results. Our plan of solving the problem gives rise to a selective refinement strategy which implicitly rules out incorrect or weakly supported estimates. Only highly correlated and confident depth estimates shall persist after this refinement process, resulting in our final double layer estimates. Recall that a major goal of this project is to deliver depth estimates for transparent objects in front of an opaque surface. Thus the front layer shall enable estimation of a transparent

surface and the back layer gives depth information about the remaining scene behind the transparent object.
From the idea of finding highly correlated depth estimates which are supported by high confidence, we came up with a voting scheme based on the projection of light field rays into a voxel volume. Before we explain the projection step all by itself we need to denote the following observations.

Note that from our acquisition setup (see fig. 2.3) we obtain P images, each from a different angular perspective. Recall that those P images make up our light field. Furthermore assume an odd number of images, i.e. an odd number of scanlines. Thus our light field stack is comprised of P images resulting from P angular views, where we denote the middle view with index $s_{\text{ref}} = \frac{P+1}{2}$ as our reference view. We also refer to it as the central view because it is this view which is acquired from the most central scanline which is aligned vertically with the axis of the camera itself. Given an arbitrary surface, assume that a given point Γ on this surface is captured by the central view at a location y in an image. This is depicted in fig. 5.1 by the blue image rays. Assume that the same scene point Γ is also captured by other scanlines with different spatial shifts Δy_1 and Δy_2 as indicated by the corresponding red and green rays.

This allows us to make out scene point correspondences across different views which is the basis for the orientation analysis of lines in the EPI domain. This means that we have up to P depth estimates for a single scene point Γ. Since we estimate the depth through angular results as explained in section 3.1, we know the angle of each ray correspondence for each estimate in our image stack. To make this principle a bit clearer, recall the single orientation results $\Lambda_{\hat{\xi}}(x, y, s)$ from section 3.5. Assume we fix an epipolar image at x^*. Since each pixel in this epipolar plane image refers to an angle $\hat{\xi}$ we can determine the relative shift in y w.r.t. the reference view s_{ref} through

$$\Delta y_{i \to ref} = \left\lfloor \tan\left(\Lambda_{\hat{\xi}}(x^*, y_i, s_i)\right)(s_i - s_{\text{ref}}) \right\rceil \tag{5.1}$$

Note that an error margin results due to the rounding to the nearest correspondence in y which is determined by the spatial resolution. An illustration for some exemplary correspondences in the EPI domain are illustrated in fig. 5.2.

With this knowledge it is thus possible to find the correspondences for all views s relative to the reference view s_{ref}. On the scale of the whole range in x and y it is thus possible to align all results from $\Lambda_{\hat{\xi}}(x, y, s)$. This is equally applicable for the second order results $\Phi_{\hat{\theta}}(x, y, s)$, $\Phi_{\hat{\gamma}}(x, y, s)$, $\Psi_{\hat{\theta}}(x, y, s)$ and $\Psi_{\hat{\gamma}}(x, y, s)$ from section 4.4.
Since we want to find out correlations and high confidence regions of our depth estimates, we came up with the idea of projecting those estimates in a voting volume. To explain this any further we need to define the voxel volume itself. Since we have just explained that the angular result volumes from the structure tensor models can be reprojected to their respective central view, it comes natural to define a voxel volume with the image height M and width N as its first two dimensions. The 3^{rd} dimension shall be given by D discretized depth-bins, into which we will project our estimates. The voxel volume thus

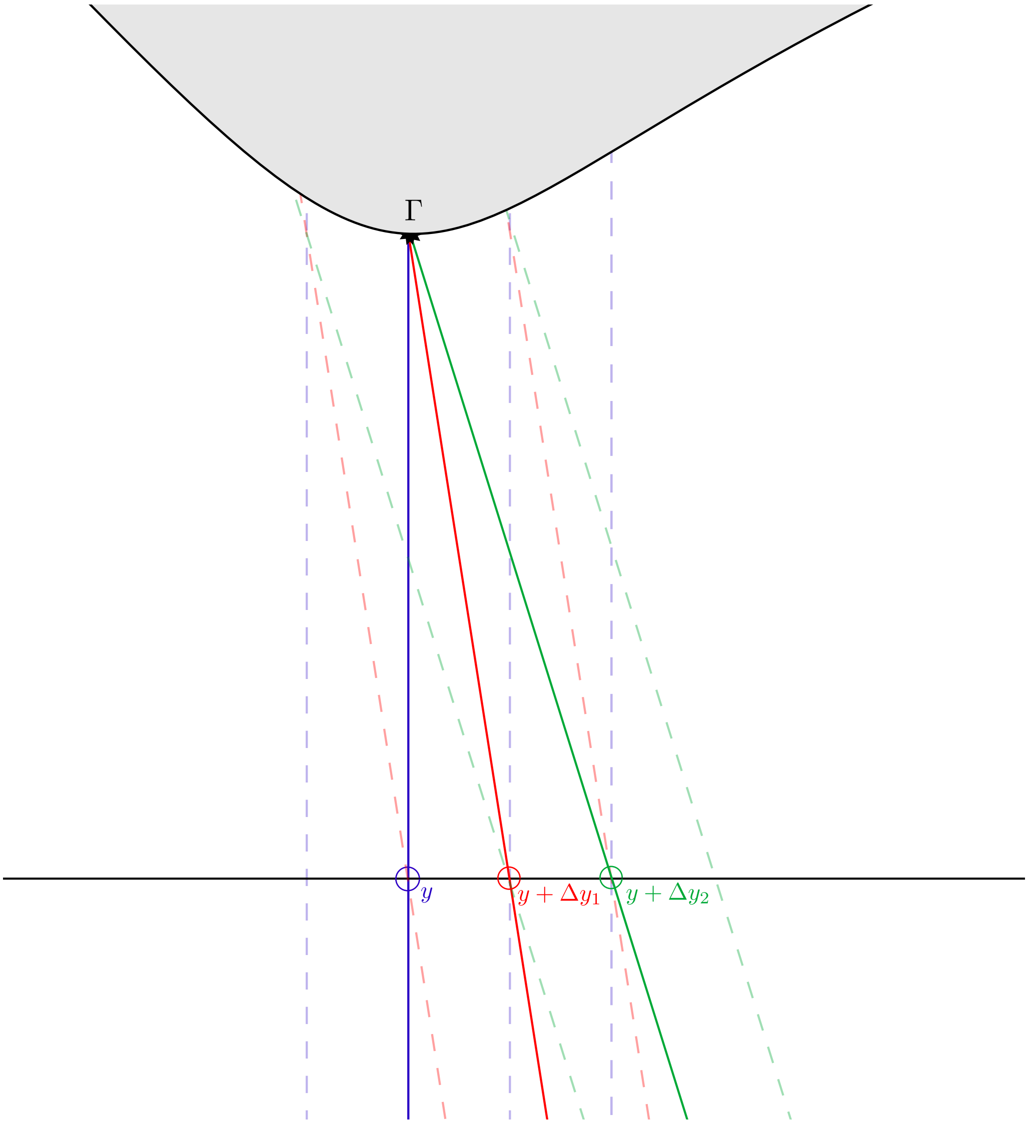


Figure 5.1: Ray correspondences between different angular views (blue, red, green). Note that through the fixed angular perspective of the different scanlines, the step Δy in transport direction determines which rays hit the same scene point Γ.

is denoted $V(x, y, d)$. See fig. 5.3 for an illustration of the projection into the voxel volume.

With the given volume, the next thing we are going to explain is how to determine the correct voxel in the direction of the projection d. Assume a given ray will be projected into the volume at the corresponding central view coordinates x^* and y^*. Recall the exemplary assumption of single orientation estimates, then the associated angle and confidence for a single estimate shall be denoted $\hat{\xi}$ and C_2. Assume further that the largest angular estimate from $\Lambda_{\hat{\xi}}(x, y, s)$ is denoted $\hat{\xi}_{max} = \max\left(\Lambda_{\hat{\xi}}(x, y, s)\right)$. Likewise the smallest estimate is denoted $\hat{\xi}_{min} = \min\left(\Lambda_{\hat{\xi}}(x, y, s)\right)$. The angular range in between $\hat{\xi}_{min}$ and $\hat{\xi}_{max}$ is mapped

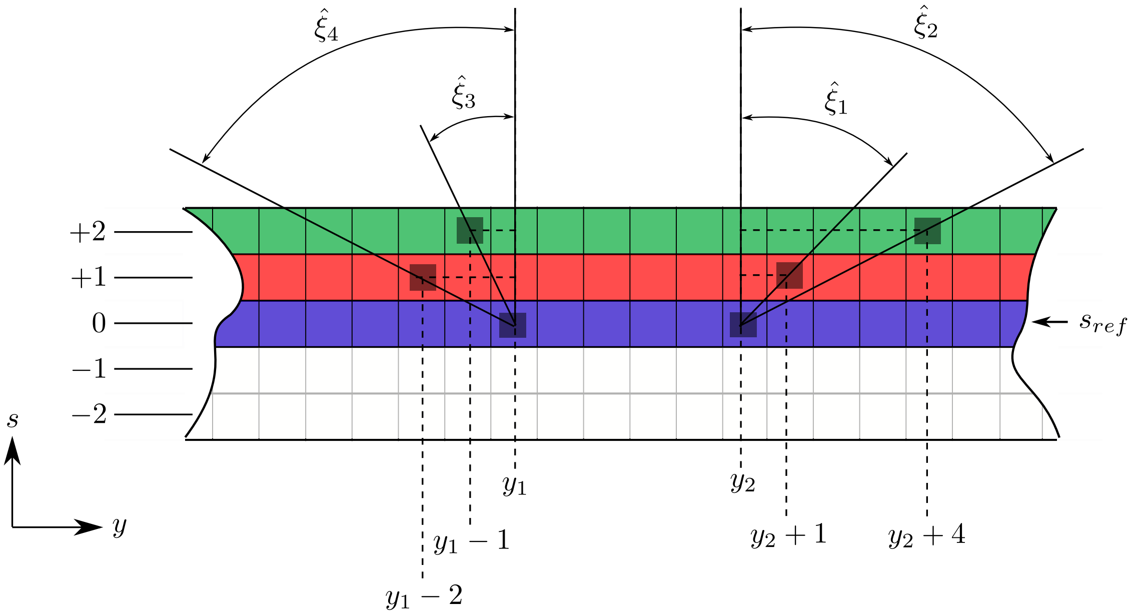


Figure 5.2: Section of an epipolar plane image $\Lambda_{\hat{\xi}}(x^*, y, s)$ with four exemplary reference view correspondences. The numbering on the left of this figure depicts the relative delta in s to the reference view s_{ref}. As an example, observe that the value of the indicated element $\Lambda_{\hat{\xi}}(x^*, y_2 + 1, s_{\text{ref}} + 1)$ is given by $\hat{\xi}_1$. Thus through eq. (5.1) it is possible to compute the corresponding shift relative to the reference view.

to the full range of bins D. It follows

$$d_i = \frac{\left\lceil \hat{\xi} - \hat{\xi}_{min} + \epsilon \right\rceil}{\hat{\xi}_{max} - \hat{\xi}_{min}} \cdot D \tag{5.2}$$

where ϵ denotes a small offset value to prevent $d_i = 0$. Note that these limit values can be chosen manually in the case of high noise levels. This way noisy outliers are clipped by the angular range. With the corresponding depth bin d_i the value in the voxel $V(x^*, y^*, d_i)$ is increased by 1. In this way all estimates are accumulated in the aforementioned voting scheme. Notice that through this projection step we became independent of all the different model hypotheses and assumptions. However since we desire to obtain accumulations of highly confident estimates, we apply a confidence based selection rule prior to projection, to rule out unconfident estimates. This will be shown in the upcoming section.

5.1.1 Ray Preselection Measures

Before we introduce our methods for extracting double layer depth estimates from the introduced voxel volume, we want to dedicate this section to selection measures which are used prior to estimate projection.

Recall from section 2.3 where we explained that we compute our depth estimates for two illumination directions (from the left and from the right) separately. This effectively doubles the amount of estimates to project. However prior to projection we validate corresponding estimates w.r.t. both illumination. An estimate pair is deemed valid if

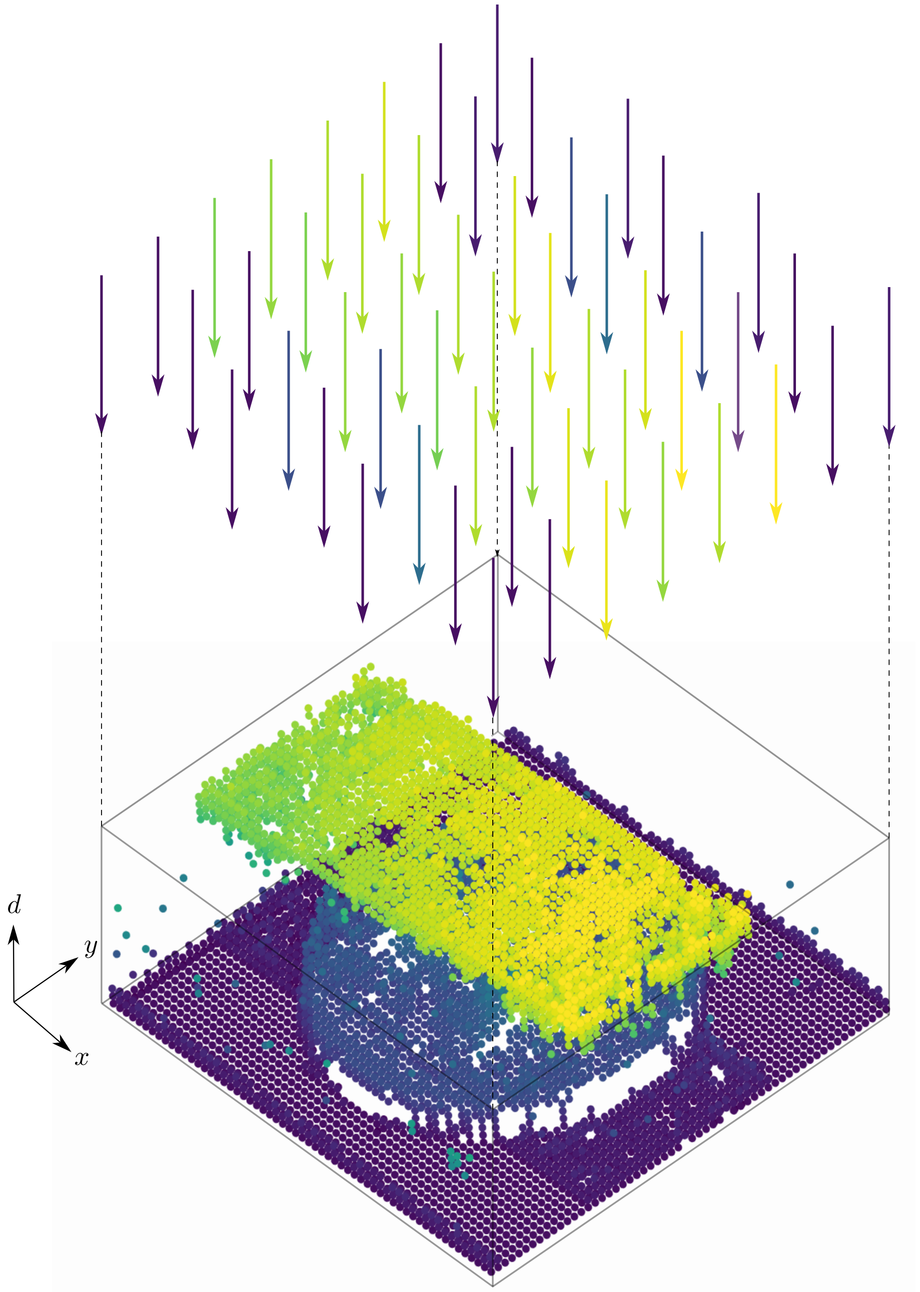
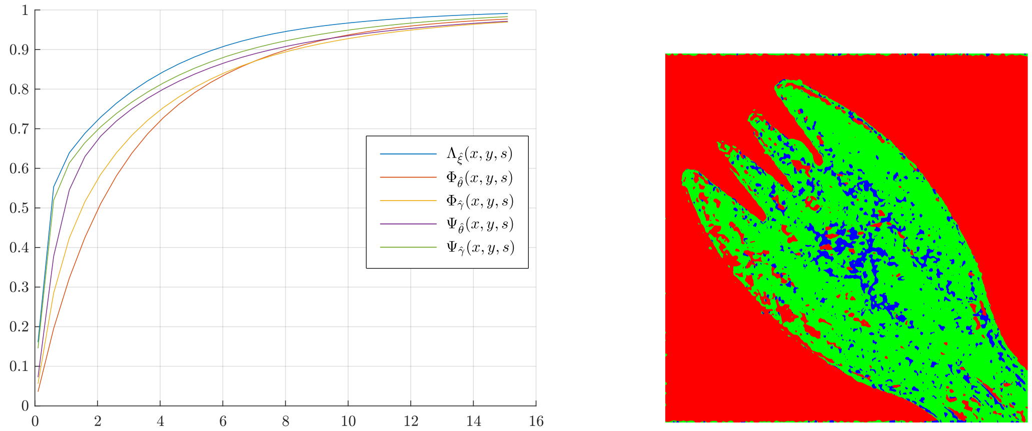


Figure 5.3: Coarsely sampled voxel volume $V(x, y, d)$ with voxels depicted as dots for the previously shown 'Coin & Tape' dataset. The voxel colours correspond to the depth in the volume similar to a depth image. The estimate projections are depicted as arrows and point in the opposite direction of the 3^{rd} dimension.

both estimates agree up to a certain threshold

$$\epsilon_{\mathrm{corr}} > |\hat{\xi}_l - \hat{\xi}_r|. \tag{5.3}$$

If the condition is not fulfilled, the estimate pair is rejected. $\hat{\xi}_l$ and $\hat{\xi}_r$ denote the left and right estimate respectively. This validation between left and right is similar to the correlation check between horizontal and vertical EPIs from a 4D light field as presented by Wanner *et al.* [52]. Fig. 5.4 a) depicts an exemplary plot for the amount of valid pairs over a given correlation threshold ϵ_{corr}.



(a) Ratio of valid pairs over ϵ_{corr} in degrees

(b) Decision Map

Figure 5.4: a) shows the ratio of estimate pairs that fulfil eq. (5.3) from a given threshold ϵ_{corr}. Subfigure b) depicts the exemplary decision map for the 'Fork' dataset. Red colour indicates the region where the SOST model is accepted. Likewise SODOST is indicated by green colour and FODOST by the blue colour.

Additionally to this consistency check between left and right we incorporate our previously introduced confidences. Since rays associated with different model estimates are projected into this voting volume, it is crucial that C_2 and C_4 reflect comparable measures to support the respective model hypothesis. Recall the discussion in section 3.2.4 and section 4.2.4. Through this step, projections with low associated confidences are ruled out from the projection process and only high confident estimates are considered. We organize the model selection according to the decision tree in fig. 5.5. To balance the distribution from all models, only one model per estimate in x, y and s is accepted. The single orientation confidence C_2 is compared against a threshold ϵ_{sost} which has to be set according to the desired balance between single- and double-orientation. If the single-orientation confidence is below threshold, the higher confident double-orientation model is accepted, i.e. either the FODOST confidence $\tilde{C}_4$ or the the SODOST confidence $\hat{C}_4$. This procedure is used for the second projection variant presented in section 5.3. For the first projection variant in section 5.2, the single orientation check against ϵ_{sost} is omitted. To get a better sens for this selection process, fig. 5.4 b) depicts an example of a color coded decision map.

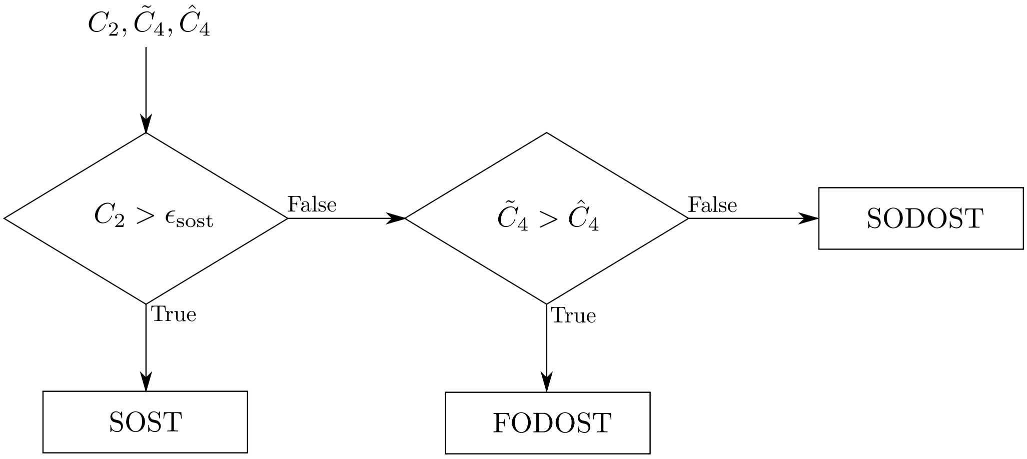

$C_2, \tilde{C}_4, \hat{C}_4$
$C_2 > \epsilon_{\text{sost}}$
False
True
SOST
$\tilde{C}_4 > \hat{C}_4$
False
True
FODOST
SODOST

Figure 5.5: Decision tree for model selection prior to the projection step. Note that $\tilde{C}_4$ denotes the FODOST confidence and $\hat{C}_4$ the SODOST confidence.

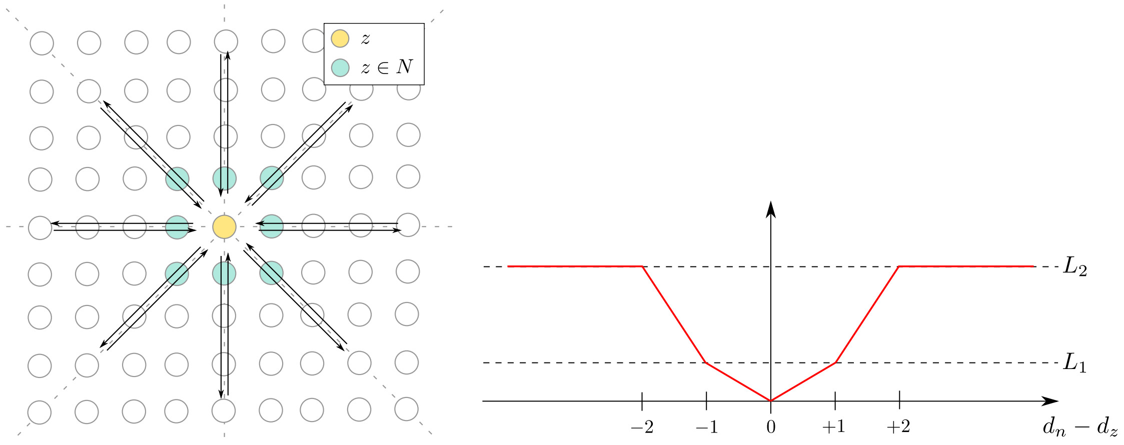
5.1.2 Voxel Volume Regularization

Before we move on with the presentation of the aforementioned strategies for the double layer extraction, we will discuss volume based regularization methods. Since we want to extract smooth results, we need to employ a method to regularize our voxel volume. Inspired by semi-global cost volume regularization in the context of stereo matching as in [22] and [13] we decided to regularize our voxel volume through a similar message parsing approach.
Since x and y are the directions we regularize in and d resembles the direction of our cost measure we denote $z = [x \; y]^{\mathrm{T}}$. First we transform our voting volume into a unary cost volume by negating it.

$$C_u(z, d) = -V(z, d)$$

Let $N = \left\{ z_n = [x_n \; y_n]^{\mathrm{T}} \;\mid\; 1 \leq |x_n - x|;\; 1 \leq |y_n - y|;\; z \neq z_n \right\}$ denote the intermediate neighbourhood around z. We implemented our method such that we compute the optimal path by summing up the min-marginals in different directions given by the 8 adjacent neighbours. This leads to a star-shaped propagation pattern as in fig. 5.6.

Along this 8 directions we define pairwise costs through two penalizations terms L_1 and L_2. Assume d_z denotes a depth bin at a location z. We penalize all voxel changes d_n where $|d_n - d_z| = 1$ for all $z \in N$ with L_1. Likewise if the voxel index changes by more than one depth bin, i.e. $|d_n - d_z| > 1$ for all $z \in N$ then we set a penalty of L_2. This leads



(a) Propagation directions **(b)** Penalty function

Figure 5.6: a) depicts the local neighbourhood around some z and also shows the 8 propagation directions. b) shows the penalty function based on the parameters L_1 and L_2.

to the definition of the following energy term

$$R(d) = \sum_{z} C_u(z,d) + \sum_{z \in N} L_1 \mathbf{1}_{|d_n - d_z| = 1} + \sum_{z \in N} L_2 \mathbf{1}_{|d_n - d_z| > 1} \tag{5.4}$$

where $\mathbf{1}_S$ denotes the 1-0 indicator function

$$\mathbf{1}_S = \begin{cases} 1 & if\ S \\ 0 & else \end{cases} \tag{5.5}$$

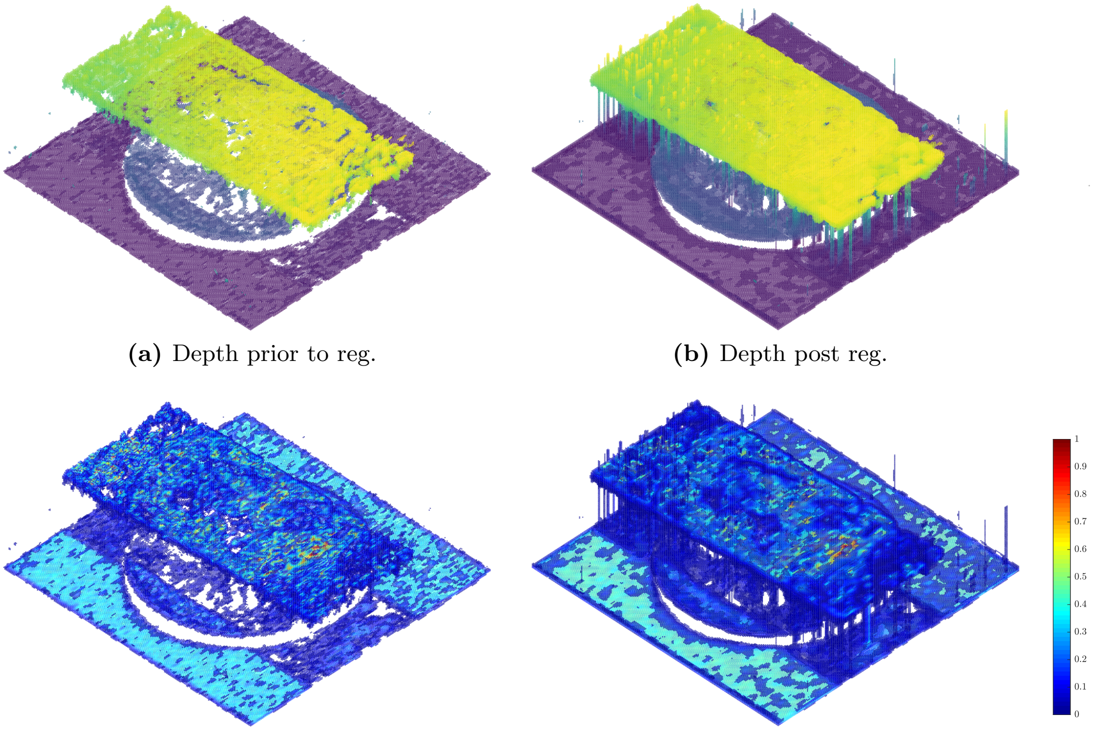
Since finding the global solution in 2D is a NP-hard problem [49] and thus not feasible, we instead can compute an optimal solution by computing the minimal cost path along each of the neighbouring pixel directions, which is similar to the principle of dynamic programming [9]. After the results from each direction are computed, the (min-marginals) are summed up. Let $C_{reg}(x, y, d)$ denote our regularized cost volume. We transform this cost volume back into a confidence volume through

$$V(x, y, d) = -C_{reg}(x, y, d) - M_d(C_{reg}(x, y, d)) \tag{5.6}$$

where $M_d(C_{reg}(x, y, d)) \in \mathrm{I\!R}^{\mathrm{M} \times \mathrm{N}}$ denotes the maximum along the direction of projection d for each x and y. A voxel volume before and after this regularization step is depicted in fig. 5.7.

5.2 Separate Voxel-Volume Approach (Variant 1)

In the introduction part of this chapter we announced that we are going to present two variants based on the voxel volume projection method introduced in section 5.1. This



(a) Depth prior to reg.

(b) Depth post reg.

(d) Confidence prior to reg.

(e) Confidence post reg.

Figure 5.7: Voxel volumes of the 'Coin & Tape' dataset before and after regularization. The voxel colors of the first row represent depth, the voxel colors in the second row depict the normalized confidence. The left column depicts the volume before regularization, the right column after regularization. The volumes has been clipped, such that only voxels with a voting value > 3 are shown.

section will be discussing the first of said variants. Recall our goal of refining our depth estimates such that we obtain a double layer result. One possibility of obtaining 2 layers from the presented projection approach is by maintaining separate voting volumes for the foreground and the background respectively, see fig. 5.8. Note that with this approach we refrain from dropping the structure tensor model hypotheses in the sense that we will combine estimates from both double orientation models according to their foreground/background affiliation. A first step is to create two empty voxel volumes with D depth bins

$$V_f(x,y,d) \in \mathrm{I\!R}^{\mathrm{M}\times\mathrm{N}\times\mathrm{D}}$$

$$V_b(x,y,d) \in \mathrm{I\!R}^{\mathrm{M}\times\mathrm{N}\times\mathrm{D}}$$

Next according to model selection procedure from section 5.1.1, the corresponding rays are projected into the voxel volumes. All estimates that belong to a front layer hypothesis, i.e. $\Phi_{\hat{\theta}}(x,y,s)$ or $\Psi_{\hat{\theta}}(x,y,s)$, are projected into the foreground volume $V_f(x,y,d)$. Likewise the background estimates $\Phi_{\hat{\gamma}}(x,y,s)$ and $\Psi_{\hat{\gamma}}(x,y,s)$ are projected into the background volume $V_b(x,y,d)$. After regularization of both of these volumes according to the semi-global regularization method in section 5.1.2, we obtain our foreground and background estimates by computing the maximum along direction d for both volumes

$$G_f(x,y) = \max_d V_f(x,y,d)$$

$$G_b(x,y) = \max_d V_b(x,y,d)$$

As an optional final step, TV-L1 [35] regularization can be used to get rid of residual outliers. The whole procedure based on this variant can be found in algorithmic form in algorithm 5. The results are discussed in chapter 6.

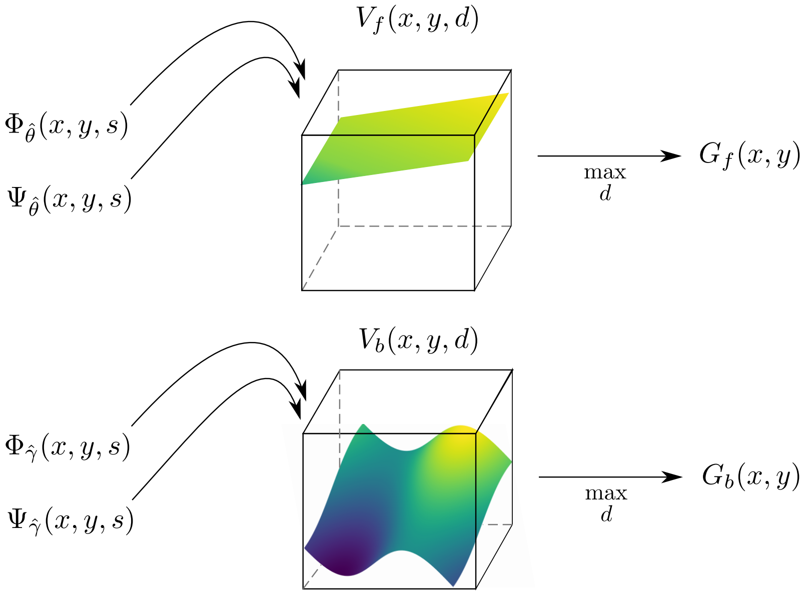

$V_f(x,y,d)$
$\Phi_{\hat\theta}(x,y,s)$
$\Psi_{\hat\theta}(x,y,s)$
max
d
$G_f(x,y)$
$V_b(x,y,d)$
$\Phi_{\hat\gamma}(x,y,s)$
$\Psi_{\hat\gamma}(x,y,s)$
max
d
$G_b(x,y)$

Figure 5.8: Depiction of the refinement procedure for variant 1.

5.3 Combined Voxel-Volume Approach (Variant 2)

The previous section discussed our proposed double layer refinement based on separate voxel volumes. In this section we will propose a second variant based on one combined voxel volume. The basic principle is to project all available structure tensor estimates into one volume $V(x,y,d) \in \mathbb{R}^{\mathrm{M}\times\mathrm{N}\times\mathrm{D}}$, see fig. 5.9. Due to this, all prior knowledge about affiliation to foreground or background from the structure tensor models is neglected. This in theory allows us to extract an arbitrary amount of layers from the volume, however we will hold on to our double layer assumption. As a remark, any further layer wouldn't have a valid support since estimates projected into the volume originate from models which have at most a double orientation hypothesis.

Similar to section 5.2 we first construct an empty volume and check the consistency between estimates from both illuminations. Afterwards the model selection according to fig. 5.5 is performed, followed by the projection of all available estimates, i.e. $\Lambda_{\hat\xi}(x,y,s)$, $\Phi_{\hat\theta}(x,y,s)$, $\Phi_{\hat\gamma}(x,y,s)$, $\Psi_{\hat\theta}(x,y,s)$ and $\Psi_{\hat\gamma}(x,y,s)$ into $V(x,y,d)$. Directly after the volume is regularized as described in section 5.1.2, we apply the non-maximum suppression and the joint TV-L1 method which will be described in the following sections. From this we obtain our final two layers $H_f(x, y)$ and $H_b(x, y)$.

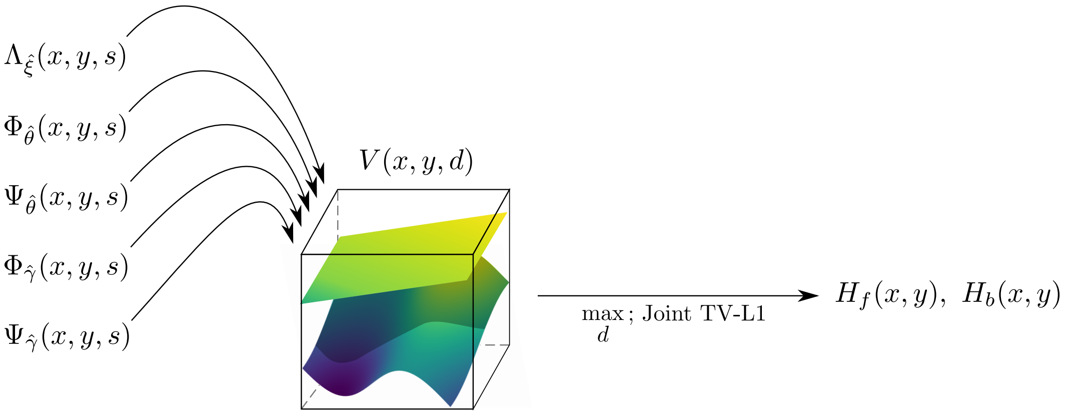

$\Lambda_{\hat{\xi}}(x, y, s)$
$\Phi_{\hat{\theta}}(x, y, s)$
$\Psi_{\hat{\theta}}(x, y, s)$
$\Phi_{\hat{\gamma}}(x, y, s)$
$\Psi_{\hat{\gamma}}(x, y, s)$
$V(x, y, d)$
$\max_{d}$; Joint TV-L1
$H_f(x, y), \; H_b(x, y)$

Figure 5.9: Depiction of the refinement procedure for variant 2.

5.3.1 Non-Maximum Suppression

Since we want to extract a foreground and a background layer from our regularized voxel volume $V(x, y, d)$, the idea comes natural to extract the two most correlated surfaces. This corresponds to extracting the two modes with the highest voting value along d for every x and y. The voxel with the highest vote can be obtained by computing the maximum along d. However, since the variance for the most confident mode is not point-wise, extracting the second highest depth bin d would in most cases return a value close to the most confident one. We can avoid this problem by applying a non maximum suppression (NMS) for the whole volume first. Algorithmic details regarding this are denoted in algorithm 2.

Algorithm 2 Implementation of the non-maximum suppression.

```
procedure NON-MAXIMUM SUPPRESSION($V(x, y, d)$)
    $V_d(x, y, d) \leftarrow \text{dilate}(V(x, y, d),\ d)$        ▷ Dilate along $d$ with width=3
    $V_{\text{NMS}}(x, y, d) \leftarrow V(x, y, d) \circ \mathbf{1}_{V(x,y,d)=V_d(x,y,d)}$
    return $V_{\text{NMS}}(x, y, d)$
end procedure
```

Here $\mathbf{1}_{V(x,y,d)=V_d(x,y,d)}$ denotes the element-wise 1-0 indicator function similar to eq. (5.5). The $\circ$ operator denotes the Hadamard product. From these results we extract the two depth estimates with the two highest voting values. For given coordinates x and y we thus extract the two most confident modes where the distribution along d is bimodal. In case of a unimodal distribution we extract only the most confident depth estimate and assign it to both result layers. This gives us our highest voting estimate $g_1(x, y)$ and second most voting estimate $g_2(x, y)$. Exemplary results are depicted in fig. 5.10. Notice how these results have no clear affiliation to the front or back. Instead both images seem to be pieced together by segments belonging to the front and the back respectively. To obtain a front- and back-layer from these images, we propose a joint TV-L1 method in the upcoming section.

5.3.2 Joint TV-L1

As mentioned in the beginning of section 5.3 the proposed second variant drops all prior knowledge on the affiliation of depth estimates to the front- or the back-layer. From the two most confident estimates which we obtained in section 5.3.1, we seek to find a smooth result for the final front- and back-layer depths. Therefore we formulate a joint TV-L1 problem based on the following convex energy

$$\min_{u,v} \|\nabla u\|_{2,1} + \|\nabla v\|_{2,1} + \lambda_u \|u - g_1\|_1 + \lambda_v \|v - g_2\|_1 \quad s.t. \quad u \geq v \tag{5.7}$$

Let $g_1, g_2 \in \mathbb{R}^{\mathrm{M}\times\mathrm{N}}$ denote the aforementioned images with the two most confident depth estimates. Furthermore $u, v \in \mathbb{R}^{\mathrm{M}\times\mathrm{N}}$ denote the primal variables of the given formulation and $\nabla : \mathbb{R}^{\mathrm{M}\times\mathrm{N}} \to \mathbb{R}^{\mathrm{M}\times\mathrm{N}\times 2}$ the finite difference operator. The variables $\lambda_u, \lambda_v > 0$ are used to adjust the weighting of the terms.

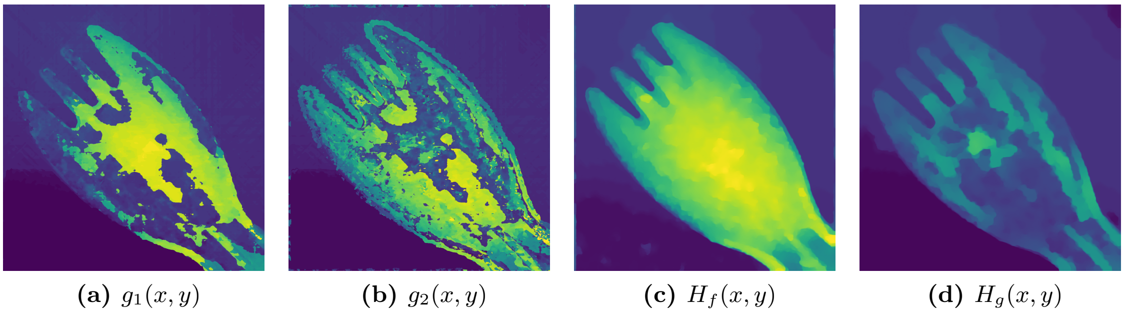

(a) $g_1(x, y)$ **(b)** $g_2(x, y)$ **(c)** $H_f(x, y)$ **(d)** $H_g(x, y)$

Figure 5.10: Input and result from the joint TV-L1 method for the 'Fork' dataset. a) and b) depict the labeling with the two highest confidences. c) and d) show the output for the front- and back-layer.

Since we want a smooth result we try to find a solution with sparse edges, thus we added a total variation term for u and v respectively. Furthermore our final result shall be close to the estimates from the original images g_1 and g_2, thus we add two data fidelity terms to our primal energy. The crucial point however is, that we want to sort our solution such that $u \geq v$, i.e. we obtain a solution for the foreground and background simultaneously. Solving this problem directly is hard due to the discontinuity of the TV semi-norm $\|\cdot\|_{2,1}$. However note that our primal problem has the general form

$$\min_x f(Kx) + h(x) \tag{5.8}$$

which can be reformulated in its saddle-point notation

$$\min_{x} \max_{y} \ (Kx)^{\mathrm{T}} y + h(x) - f^{*}(y)$$

where $f^{*}(y)$ denotes the convex conjugate of $f(x)$. By splitting up our energy w.r.t u and v we can denote the corresponding saddle-point formulation.

$$\min_{u} \left\{ \|\nabla u\|_{2,1} + \lambda_u \|u - g_1\|_1 \right\} + \min_{v} \left\{ \|\nabla v\|_{2,1} + \lambda_v \|v - g_2\|_1 \right\} \quad s.t. \quad u \geq v \tag{5.9}$$

Let $K = \nabla$ and denote that the convex conjugate of $f : \|\cdot\|_{2,1}$ is given by the indicator function of the $2, \infty$ norm-ball

$$f^{*}(y) = \delta_{\|\cdot\|_{2,\infty} \leq 1}(y) = \begin{cases} 0 & \|y\|_{2,\infty} \leq 1 \\ +\infty & else \end{cases} \tag{5.10}$$

The saddle-point formulation thus can be denoted by the following

$$\begin{aligned} &\min_{u} \max_{p} \ \left\{ (\nabla u)^{\mathrm{T}} p + \lambda_u \|u - g_1\|_1 - \delta_{\|\cdot\|_{2,\infty} \leq 1}(p) \right\} + \\ &\min_{v} \max_{q} \left\{ (\nabla v)^{\mathrm{T}} q + \lambda_v \|v - g_2\|_1 - \delta_{\|\cdot\|_{2,\infty} \leq 1}(q) \right\} = \\ &\min_{u,v} \max_{p,q} \ \left\{ (\nabla u)^{\mathrm{T}} p + (\nabla v)^{\mathrm{T}} q + \lambda_u \|u - g_1\|_1 + \lambda_v \|v - g_2\|_1 - \delta_{\|\cdot\|_{2,\infty} \leq 1}(p) - \delta_{\|\cdot\|_{2,\infty} \leq 1}(q) \right\} \\ &s.t. \quad u \geq v \end{aligned}$$

In this formulation $p, q \in \mathbb{R}^{\mathrm{M} \times \mathrm{N} \times 2}$ denote our dual variables. A saddle-point problem of this form can be adequately solved by the PDHG algorithm proposed by Chambolle and Pock [10]. The algorithm solves the problem by making alternating gradient descent steps in the primal variables and gradient ascent steps in the dual variables. The general form of PDHG is given in 3.

Algorithm 3 General PDHG algorithm from [10]

```
procedure PRIMAL-DUAL HYBRID GRADIENT(PDHG)
    Choose $x^0 \in \mathbb{E}$; $y^0 \in \mathbb{E}^*$; $\tau, \sigma > 0$
    for $k \geq 0$ do
        $x^{k+1} \leftarrow \mathrm{prox}_{\tau h}(x^k - \tau K^{\mathrm{T}} y^k)$
        $\tilde{x}^{k+1} \leftarrow 2x^{k+1} - x^k$                    ▷ Over-relaxation
        $y^{k+1} \leftarrow \mathrm{prox}_{\sigma f^*}(y^k + \sigma K \tilde{x}^{k+1})$
    end for
end procedure
```

According to the reference, the algorithm converges if

$$\tau\sigma L^2 \leq 1$$

where τ, σ denote the step sizes for descent and ascent. The parameter L bounds the operator norm from above $\|K\|_{op} = \|\nabla\|_{op} \leq L$ and is estimated to be $L = \sqrt{8}$ since we operate in 2D.

To apply the algorithm to our problem we must determine the operators $\text{prox}_{\tau h}(x)$ and $\text{prox}_{\sigma f^*}(y)$. In the dual space we need to determine the proximal map of the convex conjugate of $f(x)$. Fortunately this can easily be determined and is given by projection onto the $2, \infty$ norm ball

$$\text{prox}_{\sigma f^*}(\hat{y}) = \arg\min_{y} f^*(y) + \frac{1}{2\sigma}\|y - \hat{y}\|_2^2$$

$$\text{prox}_{\sigma f^*}(\hat{y})_i = \frac{\hat{y}_i}{\max(1, \|\hat{y}_i\|_2)}$$

This is equally applicable for p and q. The proximal map $\text{prox}_{\tau h}(x)$ in primal space due to the constraint unfortunately is not so straight forward. In the unconstrained case the sought operator for $h(u)$ is given by the well known shrinkage operator for both variables u and v.

$$\text{prox}_{\tau h}(\hat{x}) = \arg\min_{x} \lambda_x \|x - g\|_1 + \frac{1}{2\tau}\|x - \hat{x}\|_2^2$$

$$\text{prox}_{\tau h}(x)_i = (g_j)_i + \max(0, |\hat{x}_i - (g_j)_i| - \lambda_x \tau)\,\text{sign}(\hat{x}_i - (g_j)_i)$$

Note that in this formulation $(g_j)_i$ denotes the index of g_j. However these separate proximal steps are only valid if the constraint $u \geq v$ is fulfilled. In the event that $u < v$ we need to find an alternative operator. In this case the closest valid solution is subject to $u = v$. The proximal map for both primal variables thus becomes

$$\text{prox}_{\tau h}(\bar{u}, \bar{v}) = \arg\min_{u} \lambda_u \|u - g_1\|_1 + \lambda_u \|u - g_2\|_1 + \frac{1}{2\tau}\|u - \bar{u}\|_2^2 + \frac{1}{2\tau}\|u - \bar{v}\|_2^2$$

This minimization is problematic due to multiple L1-norms, however it can be solved by the median formula proposed by Li and Osher [31]. Their proposal denotes that given a minimization problem of the form

$$\arg\min_{u \in \mathbb{E}} E(u) = \sum_{i=1}^{K} w_i |u - g_i| - F(u) \qquad (5.11)$$

the optimal solution can be computed through

$$u_{opt} = v_{opt} = \text{median}\{p_0, ..., p_K, g_1, ..., g_K\}. \tag{5.12}$$

Applied to our formulation we have $K = 2$ and $w_1 = \lambda_u$, $w_2 = \lambda_v$. Furthermore note that $F(u) = \frac{1}{2\tau} \|u - \bar{u}\|_2^2 + \frac{1}{2\tau} \|u - \bar{v}\|_2^2$. The optimal solution thus is given by

$$u_{opt} = v_{opt} = \text{median}\{p_0, p_1, p_2, g_1, g_2\} \tag{5.13}$$

with $p_i = (F')^{-1}(W_i)$. The full derivation of this optimal solution can be found in section A.3. Our PHDG method can thus be formulated as denoted in algorithm 4.

Algorithm 4 PDHG algorithm as used in our method.

```
 1: procedure PDHG for Double Layer Refinement Variant 2
 2:     Choose $u^0, v^0 \in \mathbb{R}^{\mathrm{M \times N}}$; $p^0, q^0 \in \mathbb{R}^{\mathrm{M \times N \times 2}}$; $\tau, \sigma > 0$; $\lambda_u, \lambda_v > 0$
 3:     for $k \geq 0$ do
 4:         $\tilde{u}^k = u^k - \tau \nabla^{\mathrm{T}} p^k + \tau\alpha$
 5:         $\tilde{v}^k = v^k - \tau \nabla^{\mathrm{T}} q^k - \tau\alpha$
 6:         $u^{k+1} \leftarrow \text{prox}_{\tau h}(\tilde{u}^k)$
 7:         $v^{k+1} \leftarrow \text{prox}_{\tau h}(\tilde{v}^k)$
 8:         if $u^{k+1} < v^{k+1}$ then
 9:             $W_0 \leftarrow \lambda_u + \lambda_v$
10:             $W_1 \leftarrow -\lambda_u + \lambda_v$
11:             $W_2 \leftarrow -\lambda_u - \lambda_v$
12:             $p_i \leftarrow \frac{(\tilde{u}^k + \tilde{v}^k + \tau W_i)}{2}$    $i = 0, 1, 2$
13:             $u^{k+1} \leftarrow \text{median}\{p_0, p_1, p_2, g_1, g_2\}$
14:             $v^{k+1} \leftarrow u^{k+1}$
15:         end if
16:         $\tilde{u}^{k+1} \leftarrow 2u^{k+1} - u^k$
17:         $\tilde{v}^{k+1} \leftarrow 2v^{k+1} - v^k$
18:         $p^{k+1} \leftarrow \text{prox}_{\sigma f^*}(p^k + \sigma \nabla \tilde{p}^{k+1})$
19:         $q^{k+1} \leftarrow \text{prox}_{\sigma f^*}(q^k + \sigma \nabla \tilde{q}^{k+1})$
20:     end for
21: end procedure
```

Note that the additional terms $\tau\alpha$ in line 4 and line 5 help separate the solution u and v since we prefer two separate layers instead of $u = v$. This is equivalent to adding $\alpha \sum_p (v_p - u_p)$ to the primal energy in eq. (5.7), where p denotes the index of each pixel in the image. Without this precaution, solutions tend to average on a level in between the foreground and the background such that both data fidelity penalties are balanced.

This concludes our presentation of the second volume projection variant. In the upcoming section we will show and discuss the results from variant 1 & 2.

6. Results

In the course of the previous chapter we introduced two variants of our double layer refinement method. Based on different structure tensor models we showed how we combine the depth estimates from all these models such that we obtain a double layered final result. To support our newly introduced method, we acquired a variety of datasets and applied both variants to this data. These results are depicted in fig. 6.2 and fig. 6.3. Since it is in some cases difficult to comprehend a given dataset with transparent objects, appendix B gives brief descriptions for clarification.

A closer look at these figures shows that the depth estimation for the transparent objects is mostly consistent between both variants. It also can be observed that both variants behave differently when it comes to the depth estimation of opaque surfaces. As an example consider the background region surrounding the coin in the 'Coin & Tape' dataset. This can be explained by the conceptional difference in the number of projection volumes. Since the depth estimate from variant 2 is obtained from the uni-modal/bi-modal evaluation in the combined volume, the depth estimates in these regions are consistent between $H_f(x, y)$ and $H_b(x, y)$. Furthermore notice that the results for variant 1 show minor regularization artefacts due to the absence of subsequent TV-L1 regularization.

On a general note it can be observed that the applicability of both variants heavily depends on the nature of the transparent object itself. Naturally it follows that the quality of the results thus varies between datasets. Note that this statement is purely based on our subjective impressions. For an objective metric and comparison we would require ground truth data in addition to the input data. This ground truth data could either be provided by synthesizing data in the first place or by providing precise measurements through other approaches as for example through active stereo methods. Unfortunately due to the limited scope and time frame of this thesis we aren't able to provide ground truth data.

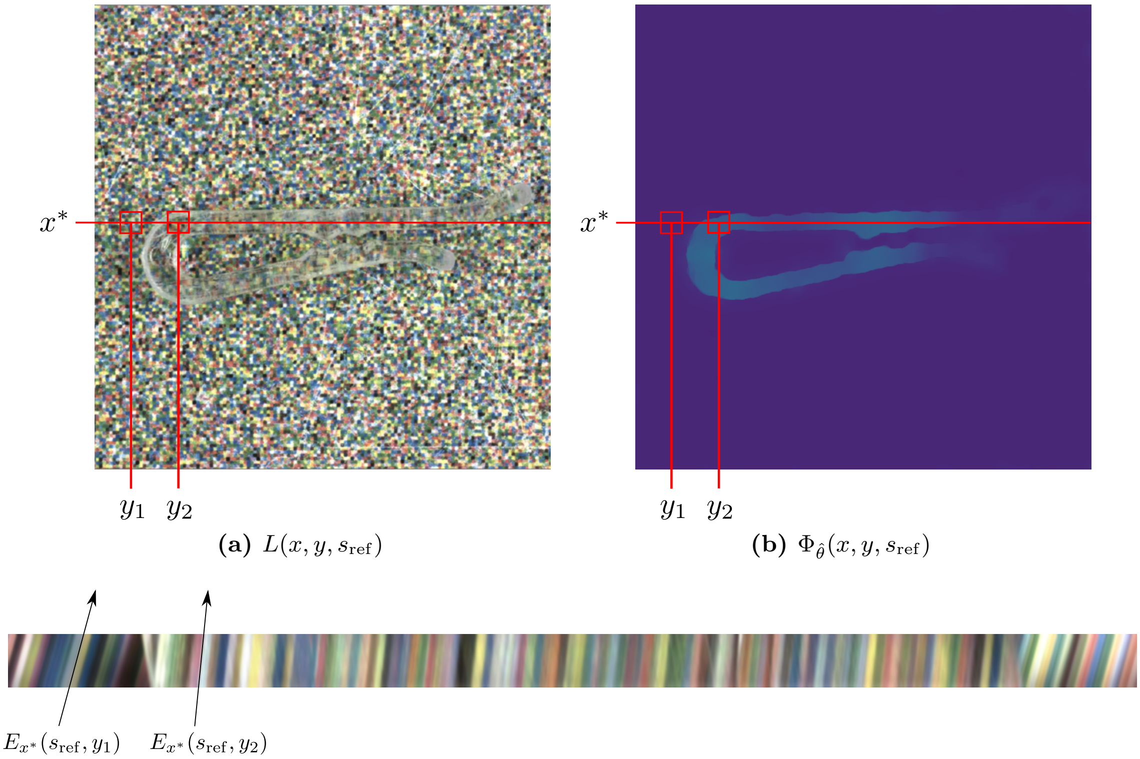
The only external transparency-dataset we were able to test was the aforementioned 'Maria' dataset, which shows a figurine behind a glass plane. This dataset is well conditioned since the transparent object is planar and oriented orthogonal to the camera axis. Unfortunately also this dataset didn't provide ground truth data. Transparent objects of this kind resemble the most basic case. A similar case can also be seen in the 'Phone Case' or 'Coin & Tape' datasets. Others as for example the 'Fork' or 'Eyedrops' datasets show more advanced shapes but are still estimated well. Notice that the results from some datasets show regions, where the back-layer estimate aligns with the front-layer, which implies that those regions are occluded. In other words, in regions where this is the case, the object isn't transparent enough for a double layer estimation.

It is interesting to observe how well our method works for highly reflective data such as the 'Wooden Balls' or 'Wallplugs' dataset. This is also the case for the planar regions of the Coin in the 'Coin & Tape' example. Not only is it possible to estimate the transparent surface well but also the object within the plastic bag can be identified and separated from the rest. For a comparison with the plain results from the structure tensor models, see fig. 4.11.
From the aforementioned figures it can be seen, that the principle approach with structure tensors is well suited for most of the acquired datasets, but for others the approach delivers unsatisfying results. Thus we will discuss the limits of the method in the upcoming section and describe our findings w.r.t. to the acquired data.

6.1 Limitations to the Method

From the results which have been presented in the previous section, it can clearly be seen that some datasets work better than others. The applicability of the presented method is determined by many factors and conditions imposed by the given data. First and foremost, the most trivial observation is that the approach based on structure tensor computation requires transparent surfaces with at least a minimal amount of surface structure. This may be given by surface roughness of the object itself or other causes such as markings, scratches or dust. Without this surface conditioning, little to no structure will be imposed in the EPI domain thus rendering the structure tensor approach useless. Note that because we are dealing with passive methods, the presence of this surface conditioning is a necessary condition for this approach to work. As an example, consider the 'Ruler' dataset from fig. 6.3. Due to the plain and smooth surface, no depth clues can be picked up in the data. This can also be observed from the EPIs of this dataset. One possibility to handle the depth estimation of such hard cases could be the incorporation of prior knowledge through shape constraints. As opposed to the presented approach which is a straight forward estimation based on the given data, an approach of this kind would allow for a more global solution. However this would still require the presence of at least a minimal amount of surface structure and thus remains unsuitable for completely transparent objects. An approach from a completely different perspective could be developed by using the refractive properties of the transparent object to our advantage. Observe that the depth level of the transparent object in the aforementioned 'Ruler' dataset is different relative to the background. In fact a vast variety of the back-layer results from the presented datasets show a different depth level than the background underneath. This is due to refractive lifting effects of the transparent materials. It can be seen that different materials with different object thicknesses impose this effect in different magnitudes on the results. Fig. 6.1 depicts this effect in the EPI domain. Inspired by the approach in [11], one method could utilize the refractive properties of light w.r.t different wave lengths. Obtaining clues from the depth estimation for single channel data separately could yield a viable solution for transparent objects with no surface structure.

Another fundamental limitation to the structure tensor based approach is the apparent limitation to two orientations. Adding another orientation and introducing a 3^{rd} order


x^*
y_1 y_2
x^*
y_1 y_2
$E_{x^*}(s_{\text{ref}}, y_1)$
$E_{x^*}(s_{\text{ref}}, y_2)$

(a) $L(x, y, s_{\text{ref}})$

(b) $\Phi_{\hat{\theta}}(x, y, s_{\text{ref}})$

(c) EPI $E_{x^*}(s, y)$

Figure 6.1: Demonstration of the refractive lifting effect on the 'Shirt Clips' dataset. a) depicts the central view of the input LF. b) depicts the background estimate from variant 2. c) shows the epipolar image $E_{x^*}(s, y)$ at x^*. Observe how the refractive effects of the transparent material lift the background to a closer depth estimate. Comparing $E_{x^*}(s_{\text{ref}}, y_1) = -15.34°$ with $E_{x^*}(s_{\text{ref}}, y_2) = -6.67°$ reveals that this effect can directly be observed in the given EPI.

structure tensor model would yield a 4×4 structure tensor which would be constructed from 3^{rd} order derivatives. Since we already showed how noise sensitive the second order structure tensor is, we would expect a triple orientation model to be impractical. Also the computational effort would skyrocket due to the larger scale of the tensor.

6.2 Implementation

We have implemented and tested our algorithms in MATLAB. Further we rely on a high degree of parallelisation since a lot of operations are point-wise. This especially applies to acceleration on a GPU. Depending on the size of the given light fields, our implementation usually requires a high amount of memory. This is mostly due to the parallel eigenvalue decomposition for the double orientation models and the voxel volume projection. As an example, the results for both variants of the 'Maria' [53] dataset with $M = 926,\ N = 926,\ P = 9$ can be computed in a few seconds. The largest acquisition that we were able to compute without running out of memory was $M = 2344,\ N = 2304,\ P = 33$. For such a LF the runtime is close to a minute.

We tested our implementation on a system based on an AMD Ryzen™ 3700X CPU,

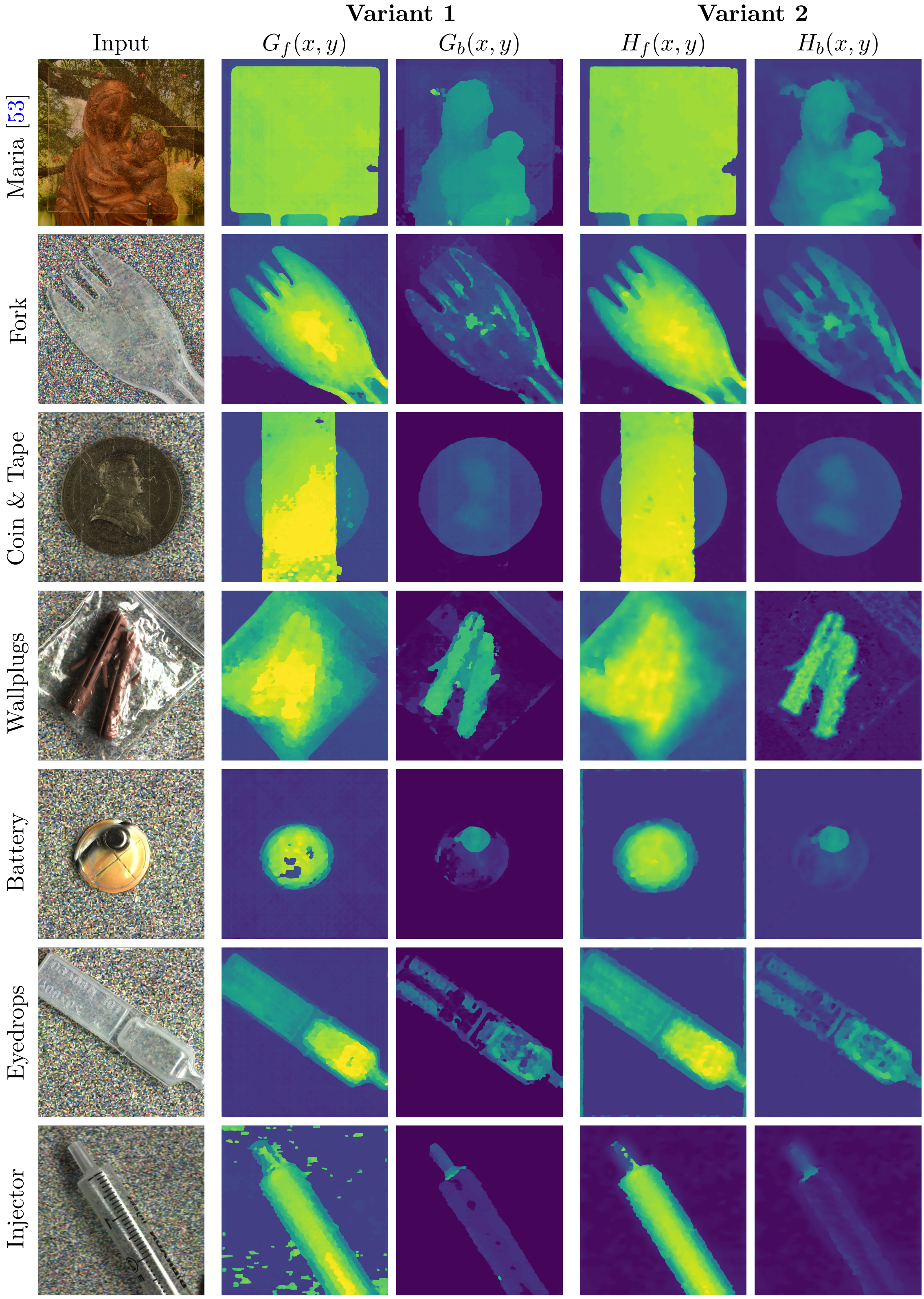


Figure 6.2: Final results for the introduced variants from section 5.2 and section 5.3, each depicting front- and back-layer. Part 2.

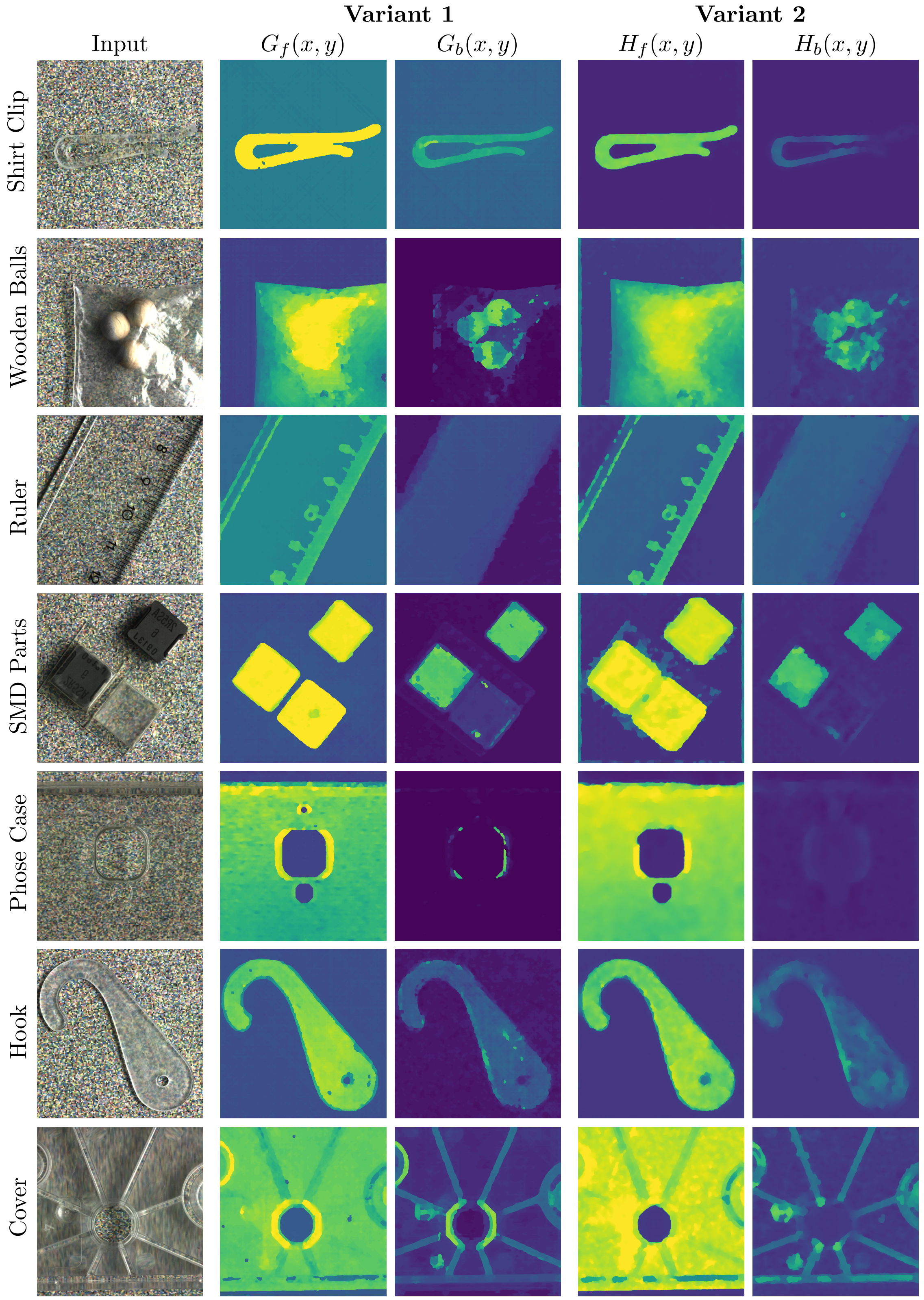


Figure 6.3: Final results for the introduced variants from section 5.2 and section 5.3, each depicting front- and back-layer. Part 2.

supported by 64 GB of RAM. For the GPU acceleration an NVIDIA$^{®}$ GeForce RTX 2080 Ti was used.

6.2.1 Implementation of Variant 1

Algorithm 5 Double layer depth computation for a scene with transparent materials based on the voxel-volume projection method (Variant 1)

```
procedure DOUBLE LAYER DEPTH ON TRANSPARENT (VARIANT 1)($L(x, y, s)$)
    $L(x, y, s) \leftarrow$ Load light field data
    Perform contrast normalization                                  ▷ see algorithm 1
    for each epipolar plane image $E_x(s, y)$ do
        $\hat{d}_y, \hat{d}_s \leftarrow \sigma_i * \nabla E_x(s, y)$                 ▷ Compute gradients, see section 3.2.3
        $\hat{d}_{yy}, \hat{d}_{sy} \leftarrow \sigma_i * \nabla \hat{d}_y$                       ▷ Compute $2^{nd}$ order gradients
        $\hat{d}_{ys}, \hat{d}_{ss} \leftarrow \sigma_i * \nabla \hat{d}_s$
    end for
    Aggregation over $T$, $T_f$                                     ▷ see section 3.2.5
    $T_f \leftarrow \hat{d}_y, \hat{d}_s$                              ▷ Construct double orientation ST eq. (4.10)
    $T \leftarrow \hat{d}_{yy}, \hat{d}_{sy}, \hat{d}_{ss}$                         ▷ Construct double orientation ST eq. (4.7)
    $\Phi_{\hat{\theta}}(x, y, s), \Phi_{\hat{\gamma}}(x, y, s) \leftarrow T$          ▷ Compute angular estimates section 4.2.3
    $\Psi_{\hat{\theta}}(x, y, s), \Psi_{\hat{\gamma}}(x, y, s) \leftarrow T_f$
    $C_4(x, y, s) \leftarrow T, T_f$                                 ▷ Compute confidence section 4.2.4
    Validate $\Phi_{\hat{\theta}}$,$\Psi_{\hat{\theta}}$,$\Phi_{\hat{\gamma}}$,$\Psi_{\hat{\gamma}}$                                 ▷ see section 5.1.1
    $V_f(x, y, d) \leftarrow \text{proj}(\Phi_{\hat{\theta}}, \Psi_{\hat{\theta}}, C_4)$             ▷ Proj. estimates, see section 5.1
    $V_b(x, y, d) \leftarrow \text{proj}(\Phi_{\hat{\gamma}}, \Psi_{\hat{\gamma}}, C_4)$
    $V_f(x, y, d) \leftarrow$ Volume Regularization                  ▷ see section 5.1.2
    $V_b(x, y, d) \leftarrow$ Volume Regularization
    $G_f(x, y) \leftarrow V_f(x, y, d)$                               ▷ see section 5.2
    $G_b(x, y) \leftarrow V_b(x, y, d)$
    return $G_f(x, y), G_b(x, y)$
end procedure
```

6.2.2 Implementation of Variant 2

Algorithm 6 Double layer depth computation for a scene with transparent materials based on the voxel-volume projection method (Variant 2)

```
procedure Double Layer depth on Transparent (Variant 2)($L(x, y, s)$)
    $L(x, y, s) \leftarrow$ Load light field data
    Perform contrast normalization                                   ▷ see algorithm 1
    for each epipolar plane image $E_x(s, y)$ do
        $\hat{d}_y, \hat{d}_s \leftarrow \sigma_i * \nabla E_x(s, y)$          ▷ Compute gradients, see section 3.2.3
        $\hat{d}_{yy}, \hat{d}_{sy} \leftarrow \sigma_i * \nabla \hat{d}_y$          ▷ Compute $2^{nd}$ order gradients
        $\hat{d}_{ys}, \hat{d}_{ss} \leftarrow \sigma_i * \nabla \hat{d}_s$
    end for
    Aggregation over $S$, $T$, $T_f$                                 ▷ see section 3.2.5
    $S \leftarrow \hat{d}_y, \hat{d}_s$                              ▷ Construct single orientation ST eq. (3.5)
    $T_f \leftarrow \hat{d}_y, \hat{d}_s$                            ▷ Construct double orientation ST eq. (4.10)
    $T \leftarrow \hat{d}_{yy}, \hat{d}_{sy}, \hat{d}_{ss}$          ▷ Construct double orientation ST eq. (4.7)
    $\Lambda_{\hat{\xi}}(x, y, s) \leftarrow S$                      ▷ Compute angular estimates section 3.2.1
    $\Phi_{\hat{\theta}}(x, y, s), \Phi_{\hat{\gamma}}(x, y, s) \leftarrow T$          ▷ Compute angular estimates section 4.2.3
    $\Psi_{\hat{\theta}}(x, y, s), \Psi_{\hat{\gamma}}(x, y, s) \leftarrow T_f$
    $C_2(x, y, s) \leftarrow S$                                      ▷ Compute confidence section 3.2.4
    $C_4(x, y, s) \leftarrow T, T_f$                                 ▷ Compute confidence section 4.2.4
    Validate $\Lambda_{\hat{\xi}}$,$\Phi_{\hat{\theta}}$,$\Psi_{\hat{\theta}}$,$\Phi_{\hat{\gamma}}$,$\Psi_{\hat{\gamma}}$          ▷ see section 5.1.1
    $V(x, y, d) \leftarrow \text{proj}(\Lambda_{\hat{\xi}}, C_2)$    ▷ Proj. estimates, see section 5.1
    $V(x, y, d) \leftarrow \text{proj}(\Phi_{\hat{\theta}}, \Psi_{\hat{\theta}}, C_4)$
    $V(x, y, d) \leftarrow \text{proj}(\Phi_{\hat{\gamma}}, \Psi_{\hat{\gamma}}, C_4)$
    $V(x, y, d) \leftarrow$ Volume Regularization                    ▷ see section 5.1.2
    $V_{\text{NMS}}(x, y, d) \leftarrow$ Non-Maximum Suppression      ▷ see section 5.3.1
    $g_1(x, y) \leftarrow V_{\text{NMS}}(x, y, d)$                   ▷ see section 5.2
    $g_2(x, y) \leftarrow V_{\text{NMS}}(x, y, d)$
    $H_f(x, y), H_b(x, y) \leftarrow$ joint TV-L1                    ▷ see section 5.3.2
    return $H_f(x, y), H_b(x, y)$
end procedure
```

7. Conclusion

In the course of this master's thesis we discussed the theory and principles of 3D depth computation for scenes with transparent materials. We gave an introduction to the basic problem and discussed how structure tensor methods in conjunction with light field data can be used to tackle this problem. After highlighting relevant preliminaries, we showed how single orientation structure tensor models can be applied to compute depth estimates for Lambertian surfaces. We subsequently showed the importance of double orientation models for the extension to non-Lambertian surfaces. Not only did we discuss the mathematical principles of the presented models, but also highlighted findings and improvements to the method. Furthermore we showed the results of the structure tensor models and discussed the behaviour of each model in regions where the corresponding model hypothesis is wrong. Moreover we presented our solution to combine model based estimates and showed how to improve upon existing methods. Our procedure based on voxel-volume projection enabled us to effectively utilize estimates from different models such that we obtained smooth double layer results. Subsequently we presented the results on acquired data for two distinct variants of this volume projection approach. During the presentation of these results we showed the limitations regarding the properties of the input data. We showed that the presented approach relies on structure in the EPI domain which subsequently leads to problems for completely transparent objects. We finalized this thesis by pointing out further findings and presented detailed information regarding our implementation.

Appendices

A. Proofs and Derivations

A.1 Proof of Transparency Constraint

Let $x \in \mathbb{R}^2$ denote bivariate image coordinates and let $f(x)$ denote a double-orientation image patch in a region Ω. Further let $f(x)$ be an additive composition of two single-orientation images according to eq. (4.1). Through the linearity and commutativity of the operators $\left(\frac{\partial}{\partial u(\theta)}\right)$ and $\left(\frac{\partial}{\partial v(\gamma)}\right)$ we can prove the transparency constraint [42] for the overlaid patch description $f(x)$.

$$\frac{\partial^2 f(x)}{\partial u(\theta) \partial v(\gamma)} = 0 \qquad \forall x \in \Omega \tag{A.1}$$

Poof:

$$\begin{aligned}
\left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right) f(x) &= \left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right)(f_1(x) + f_2(x)) \\
&= \left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right) f_1(x) + \left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right) f_2(x) \\
&= \left(\frac{\partial}{\partial v(\gamma)}\right)\left(\frac{\partial}{\partial u(\theta)}\right) f_1(x) + \left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial}{\partial v(\gamma)}\right) f_2(x) \\
&= \left(\frac{\partial}{\partial v(\gamma)}\right)\left(\left(\frac{\partial}{\partial u(\theta)}\right) f_1(x)\right) + \left(\frac{\partial}{\partial u(\theta)}\right)\left(\left(\frac{\partial}{\partial v(\gamma)}\right) f_2(x)\right) \\
&= \left(\frac{\partial}{\partial v(\gamma)}\right)\left(\frac{\partial f_1(x)}{\partial u(\theta)}\right) + \left(\frac{\partial}{\partial u(\theta)}\right)\left(\frac{\partial f_2(x)}{\partial v(\gamma)}\right) \\
&= \left(\frac{\partial}{\partial v(\gamma)}\right) 0 + \left(\frac{\partial}{\partial u(\theta)}\right) 0 = 0 \qquad \square
\end{aligned}$$

A.2 Double-Orientation from MOP Vector Proof

Let $m_2(\theta, \gamma)^{\mathrm{T}} = \Big[\cos(\theta)\cos(\gamma), \quad \sin(\theta)\cos(\gamma) + \sin(\gamma)\cos(\theta), \quad \sin(\theta)\sin(\gamma)\Big]^{\mathrm{T}}$ denote the MOP vector for the double orientation structure tensor, then the angles θ and γ are obtained from the roots of

$$z^2 - m_2^{(2)} z + m_2^{(1)} m_2^{(3)} = 0. \tag{A.2}$$

Proof:

$$z_{1,2} = \frac{\sin(\theta+\gamma)}{2} \pm \sqrt{\frac{\sin(\theta+\gamma)^2}{4} - \cos(\theta)\sin(\gamma)\sin(\theta)\cos(\gamma)}$$

$$2z_{1,2} = \sin(\theta+\gamma) \pm \sqrt{\sin(\theta+\gamma)^2 - \underbrace{4\cos(\theta)\sin(\gamma)\sin(\theta)\cos(\gamma)}_{\cos(\theta-\gamma)^2 - \cos(\theta+\gamma)^2}}$$

$$2z_{1,2} = \sin(\theta+\gamma) \pm \sqrt{\underbrace{\sin(\theta+\gamma)^2 + \cos(\theta+\gamma)^2 - \cos(\theta-\gamma)^2}_{\sin(\theta-\gamma)^2}}$$

$$2z_{1,2} = \sin(\theta+\gamma) \pm \sin(\theta-\gamma)$$

$$z_1 = \sin(\theta)\cos(\gamma), \qquad z_2 = \cos(\theta)\sin(\gamma)$$

$$\frac{z_1}{m_2^{(1)}} = \frac{\sin(\theta)\cos(\gamma)}{\cos(\theta)\cos(\gamma)} = \tan(\theta), \qquad \frac{z_2}{m_2^{(1)}} = \frac{\cos(\theta)\sin(\gamma)}{\cos(\theta)\cos(\gamma)} = \tan(\gamma)$$

$$\theta = tan^{-1}\left(\frac{z_1}{m_2^{(1)}}\right), \qquad \gamma = tan^{-1}\left(\frac{z_2}{m_2^{(1)}}\right) \qquad \square$$

A.3 Derivation of Proximal Map

In section 5.3.2 we denoted the following minimization problem for the prox operator $\text{prox}_{\tau h}$

$$\arg\min_u \lambda_u \|u - g_1\|_1 + \lambda_u \|u - g_2\|_1 + \underbrace{\frac{1}{2\tau}\|u - \bar{u}\|_2^2 + \frac{1}{2\tau}\|u - \bar{v}\|_2^2}_{F(u)}$$

This problem is of the form [31]

$$\arg\min_{u \in \mathbb{E}} E(u) = \sum_{i=1}^{K} w_i |u - g_i| - F(u)$$

where $p_i = (F)^{-1}(W_i)$, $K = 2$, $w_1 = \lambda_u$ and $w_2 = \lambda_v$. The parameters W_i can be computed from

$$W_i = -\sum_{j=1}^{i} w_j + \sum_{j=i+1}^{n} w_j \qquad i = 0, 1, 2$$

$$W_0 = 0 + \lambda_u + \lambda_v$$

$$W_1 = -\lambda_u + \lambda_v$$

$$W_2 = -\lambda_u - \lambda_v$$

With these parameters the only step left is to compute $p_i = (F)^{-1}(W_i)$

$$\begin{aligned} F(p_i) &= \frac{1}{2\tau} \|p_i - \bar{u}\|_2^2 + \frac{1}{2\tau} \|p_i - \bar{v}\|_2^2 \\ F'(p_i) &= \frac{1}{\tau}(p_i - \bar{u}) + \frac{1}{\tau}(p_i - \bar{v}) \\ &= \frac{1}{\tau}(2p_i - \bar{u} - \bar{v}) = W_i \\ \implies p_i &= \frac{\bar{u} + \bar{v} + \tau W_i}{2} = (F')^{-1}(W_i) \qquad i = 0, 1, 2 \end{aligned}$$

This leads to the optimal solution

$$u_{opt} = v_{opt} = \text{median}\{p_0, p_1, p_2, g_1, g_2\}$$

B. Datasets

Fork

Plastic fork taken from a MRE (Meal Ready to Eat), intended for one time use.

Coin & Tape

Transparent scotch tape spanned over a coin. An illustration of this scene has been given in fig. 4.1.

Wallplugs

Ordinary wall plugs in a clear utility bag.

Battery

Cutout of a coin cell packaging including the coin cell itself. The battery is enclosed in a clear plastic dome. The remainder of the cut out has been removed to avoid brand promotion.

Eyedrops

Plastic one-way-use eyedrop dispenser from the drug store.

Injector

Classic clear plastic injector body without inner piston.

Shirt Clips

Clear clips commonly found with packaging of shirts.

Wooden Balls

Fairly similar to the 'Wallplug' dataset. Wooden balls ≈ 8 mm in a clear utility bag.

Ruler

Clear ruler with the markings on the top, i.e. facing the camera.

SMD Parts

SMD capacitors in a clear packaging. The right capacitor has been taken out of the packaging. The clear packaging is placed face done with the opening.

Phone Case

Section of clear phone case. The depictions shows the camera cut-out of the case.

Hook

Semi-transparent hook commonly found in packaging.

Cover

Clear cover from an industrial railing part.

C. Acronyms and Abbreviations

AIT	Austrian Institute of Technology
EPI	Epipolar Plane Image
FODOST	First Order Double Orientation Structure Tensor
ICI	Inline Computational Imaging
LF	Light Field
MOP	Mixed Orientation Parameter
NMS	Non-Maximum Suppression
SGM	Semi Global Matching
SODOST	Second Order Double Orientation Structure Tensor
SOST	Single Orientation Structure Tensor
ST	Structure Tensor
s.t.	subject to
TV	Total Variation
w.r.t.	with respect to

Bibliography

[1] Inline computational imaging (ici), https://www.ait.ac.at/themen/high-performance-vision/inline-computational-imaging/. AIT Auastrian Institute of Technology. 16, 17, 22, 24

[2] Gaussian derivatives. In *Front-End Vision and Multi-Scale Image Analysis: Multi-Scale Computer Vision Theory and Applications, written in Mathematics*, pages 53–69. Springer Netherlands, Dordrecht, 2003. 34

[3] T. Aach, C. Mota, I. Stuke, M. Muhlich, and E. Barth. Analysis of superimposed oriented patterns. *IEEE Transactions on Image Processing*, 15(12):3690–3700, Dec 2006. 17, 29, 31, 32, 45, 46, 47, 48, 51, 52

[4] E. H. Adelson and J. R. Bergen. The plenoptic function and the elements of early vision. In *M. S. Landy and J. A. Movshon (Eds.), Computational models of visual processing*, pages 3–20, Cambridge, MA, US, 1991. The MIT Press. 20

[5] Vamsi Kiran Adhikarla, Marek Vinkler, Denis Sumin, Rafał Mantiuk, Karol Myszkowski, Hans-Peter Seidel, and Piotr Didyk. Towards a quality metric for dense light fields. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 25

[6] Doris Antensteiner, Svorad Štolc, Kristián Valentín, Bernhard Blaschitz, Reinhold Huber-Mörk, and Thomas Pock. High-precision 3d sensing with hybrid light field & photometric stereo approach in multi-line scan framework. volume 2017, pages 52–60, 01 2017. 24

[7] Heinz H. Bauschke and Jonathan M. Borwein. On projection algorithms for solving convex feasibility problems, 1996. 35

[8] J. Bigün and G. Granlund. Optimal orientation detection of linear symmetry. In *Proc. ICCV*, pages 433–438. 29, 36

[9] S. Birchfield and C. Tomasi. Depth discontinuities by pixel-to-pixel stereo. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 1073–1080, Jan 1998. 70

[10] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. working paper or preprint, June 2010. 75

[11] Zhihu Chen, Kwan-Yee K. Wong, Yasuyuki Matsushita, and Xiaolong Zhu. Depth from refraction using a transparent medium with unknown pose and refractive index. *Int. J. Comput. Vision*, 102(1–3):3–17, 2013. 80

[12] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 303–312, New York, NY, USA, 1996. Association for Computing Machinery. 17

[13] Gabriele Facciolo, Carlo de Franchis, and Enric Meinhardt. Mgm: A significantly more global matching for stereovision. In Mark W. Jones Xianghua Xie and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 90.1–90.12. BMVA Press, September 2015. 69

[14] W. Forstner and Alfred Pertl. Photogrammetric standard methods and digital image matching techniques for high precision surface measurements. pages 57–72, 12 1986. 19

[15] G. Graber, T. Pock, and H. Bischof. Online 3d reconstruction using convex optimization. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 708–711, Nov 2011. 17

[16] M. S. K. Gul and B. K. Gunturk. Spatial and angular resolution enhancement of light fields using convolutional neural networks. *IEEE Transactions on Image Processing*, 27(5):2146–2159, May 2018. 24

[17] Haitao Wang, S. Z. Li, and Yangsheng Wang. Face recognition under varying lighting conditions using self quotient image. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, pages 819–824, May 2004. 40

[18] Marsha Jo Hannah. A system for digital stereo image matching. 1989. 19

[19] Richard Hartley and Andrew Zisserman. Epipolar geometry and the fundamental matrix. In *Multiple View Geometry in Computer Vision*, page 239–261. Cambridge University Press, 2 edition, 2004. 19

[20] S. Heber and T. Pock. Convolutional networks for shape from light field. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3746–3754, June 2016. 21, 22, 23, 25, 28, 43

[21] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Reliable surface reconstruction from multiple range images. In Bernard Buxton and Roberto Cipolla, editors, *Computer Vision — ECCV '96*, pages 117–126, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg. 17

[22] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, Feb 2008. 69

[23] Katrin Honauer, Ole Johannsen, Daniel Kondermann, and Bastian Goldluecke. A dataset and evaluation methodology for depth estimation on 4d light fields. pages 19–34, 03 2017. 25, 43

[24] Yijun Ji, Qing Xia, and Zhijiang Zhang. Fusing Depth and Silhouette for Scanning Transparent Object with RGB-D Sensor. *International Journal of Optics*, 2017:1–11, 2017. 16

[25] O. Johannsen, A. Sulc, and B. Goldluecke. Occlusion-aware depth estimation using sparse light field coding. In *German Conference on Pattern Recognition (Proc. GCPR)*, 2016. 17

[26] David G. Jones and Jitendra Malik. A computational framework for determining stereo correspondence from a set of linear spatial filters. In G. Sandini, editor, *Computer Vision — ECCV'92*, pages 395–410, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg. 19

[27] Michael Kass and Andrew Witkin. Analyzing oriented patterns. *Comput. Vision Graph. Image Process.*, 37(3):362–385, March 1987. 29

[28] Sanjeev J. Koppal. Lambertian reflectance. In Katsushi Ikeuchi, editor, *Computer Vision: A Reference Guide*, pages 441–443. Springer US, Boston, MA, 2014. 27

[29] Ullrich Köthe. Edge and junction detection with an improved structure tensor. pages 25–32, 09 2003. 35

[30] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 31–42, New York, NY, USA, 1996. ACM. 20

[31] Yingying Li and Stanley Osher. A new median formula with applications to pde based denoising. *Commun. Math. Sci.*, 7(3):741–753, 09 2009. 76, 92

[32] D. Marr and T. Poggio. A computational theory of human stereo vision. pages 301–328, C204Proc. R. Soc. Lond. B, 1979. 15

[33] Larry Matthies, Takeo Kanade, and Richard Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3:209–238, 09 1989. 19

[34] D. Miyazaki and K. Ikeuchi. Inverse polarization raytracing: estimating surface shapes of transparent objects. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 910–917 vol. 2, June 2005. 16

[35] Mila Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1):99–120, Jan 2004. 72

[36] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, April 1993. 20

[37] D. J. Panton. A flexible approach to digital stereo mapping. *Photogram. Eng. Remote Sensing vol. 44, no. 12*, pages 1499–1512, 1978. 19

[38] C. Perwass and L. Wietzke. The next generation of photography: An introduction to light field photography, http://www.raytrix.de. 2010. 21

[39] C. Perwass and L. Wietzke. Single lens 3d-camera with extended depth-of-field. volume 8291, 2012. 21

[40] M. Rossi and P. Frossard. Graph-based light field super-resolution. In *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6, Oct 2017. 24

[41] Mattia Rossi and Pascal Frossard. Geometrically consistent light field super-resolution via graph-based regularization. *IEEE Transactions on Image Processing*, PP, 01 2017. 24

[42] M. Shizawa and T. Iso. Direct representation and detecting of multi-scale, multi-orientation fields using local differentiation filters. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 508–514, June 1993. 48, 91

[43] O. Smith. Eigenvalues of a symmetric 3x3 matrix. In *Comm. ACM 4, 168*, 1961. 51

[44] Daniel Soukup, Reinhold Huber-Mörk, S Stolc, and B Holländer. Depth estimation within a multi-line-scan light-field framework. 12 2014. 23, 28

[45] Jian Sun, Yin Li, and Sing Bing Kang. Symmetric stereo matching for occlusion handling. volume 2, pages 399–406, 01 2005. 19

[46] R. Szeliski and P. Golland. Stereo matching with transparency and matting. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 517–524, Jan 1998. 16

[47] Borislav Trifonov, Derek Bradley, and Wolfgang Heidrich. Tomographic reconstruction of transparent objects. pages 51–60, 01 2006. 16

[48] Roger Y. Tsai. Multiframe image point matching and 3-d surface reconstruction. In *ICASSP*, 1983. 20

[49] Olga Veksler. *Efficient Graph-Based Energy Minimization Methods in Computer Vision*. PhD thesis, USA, 1999. AAI9939932. 70

[50] Svorad Štolc, Reinhold Huber-Mörk, Branislav Holländer, and Daniel Soukup. Depth and all-in-focus images obtained by multi-line-scan light-field approach. In *Electronic Imaging*, 2014. 17, 21, 22, 23, 24, 28

[51] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4D lightfields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 17, 21, 23, 34

[52] S. Wanner and B. Goldluecke. Reconstructing reflective and transparent surfaces from epipolar plane images. In *German Conference on Pattern Recognition (Proc. GCPR, oral presentation)*, 2013. 17, 25, 45, 51, 68

[53] S. Wanner, S. Meister, and B. Goldluecke. Datasets and benchmarks for densely sampled 4d light fields. In *Vision, Modelling and Visualization (VMV)*, 2013. 25, 43, 54, 62, 81, 82

[54] Sven Wanner. *Orientation Analysis in 4D Light Fields.* PhD thesis, Universität Heidelberg - Heidelberg Collaboratory for Image Processing (HCI), 2014. 24, 32, 35

[55] M. D. Wheeler, Y. Sato, and K. Ikeuchi. Consensus surfaces for modeling 3d objects from multiple range images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 917–924, Jan 1998. 17

[56] John W. Woods. Chapter 7 - image enhancement and analysis. In John W. Woods, editor, *Multidimensional Signal, Image, and Video Processing and Coding (Second Edition)*, pages 223 – 256. Academic Press, Boston, second edition edition, 2012. 35

[57] Bojian Wu, Yang Zhou, Yiming Qian, Minglun Cong, and Hui Huang. Full 3d reconstruction of transparent objects. *ACM Transactions on Graphics*, 37:1–11, 07 2018. 16

[58] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):926–954, Oct 2017. 20

[59] Yanghai Tsin, Sing Bing Kang, and R. Szeliski. Stereo matching with reflections and translucency. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I, June 2003. 16

[60] Kuk-Jin Yoon and Soo-Ok Kweon. Stereo matching with symmetric cost functions. volume 2, pages 2371 – 2377, 02 2006. 19

[61] Y. Yoon, H. Jeon, D. Yoo, J. Lee, and I. S. Kweon. Light-field image super-resolution using convolutional neural network. *IEEE Signal Processing Letters*, 24(6):848–852, June 2017. 24