Thesis

Multi-View 3D Reconstruction of Highly-Specular Objects

Aljoša Ošep

Bonn, March 2013

Universität Bonn
Institute of Computer Science II
Friedrich-Ebert-Allee 144, 53113 Bonn

Submitted to the Institute of Computer Science, Computer Graphics department,
University of Bonn

Reviewers: Prof. Dr. Reinhard Klein, Prof. Dr. Andreas Weber
Advisor: Dipl.-Ing. Michael Weinmann

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Bonn, 25.3.2013
Aljoša Ošep

CONTENTS

ABSTRACT

In this thesis, we address the problem of image-based 3D reconstruction of objects exhibiting complex reflectance behaviour using surface gradient information techniques. In this context, we are addressing two open questions. The first one focuses on the aspect, if it is possible to design a robust multi-view normal field integration algorithm, which can integrate noisy, imprecise and only partially captured real-world data. Secondly, the question is if it is possible to recover a precise geometry of the challenging highly-specular objects by multi-view normal estimation and integration using such an algorithm.

The main result of this work is the first multi-view normal field integration algorithm that reliably reconstructs a surface of object from normal fields captured in the real-world setup. The surface of the unknown object is reconstructed by fitting a surface to the vector field reconstructed from observed normal samples. The vector field and the surface consistency information are computed based on a feature space analysis of back-projections of the normals using robust, non-parametric probability density estimation methods. This normal field integration technique is not only suitable for reconstructing lambertian objects, but, in the scope of this work, it is also used for the reconstruction of highly-specular objects via multi-view shape-from-specularity techniques.

We performed an evaluation on synthetic normal fields, photometric stereo based normal estimates of a real lambertian object and, most importantly, demonstrated state-of-the art results in the domain of 3D reconstruction of highly-specular objects based on the measured data and integrated by the proposed algorithm. Our method presents a significant advancement in the area of gradient information based 3D reconstruction techniques with a potential to address 3D reconstruction of a large class of objects exhibiting complex reflectance behaviour. Furthermore, using this method, a wide range of proposed normal estimation techniques can now be used for the recovery of full 3D shapes.

CHAPTER 1

INTRODUCTION

3D reconstruction of real-world objects is a well-studied problem with a long history of research and development in the area of computer graphics and vision. Although over the years, many techniques have been developed and successfully applied in industry and entertainment, the area of 3D sensing and reconstruction still remains an open problem and provides several interesting challenges, one of them being the topic of this master thesis.

There are numerous applications of 3D reconstruction of real models. For the photo-realistic visual reproduction of cultural heritage and artistic works, acquisition of accurate geometry is the first and a very important step. Knowing the geometry of the objects may help mobile robots to recognize objects and plan and execute actions. As modelling of 3D objects is time-consuming and it is non-trivial to reproduce microscopic details, scanning and digitization of real objects may play important role for convincing appearance in 3D video games and movies in the future. Scanning of mechanical parts enables us to capture, analyse, visualize and even virtually modify them, do reverse engineering and physical simulations. It has also applications in medical imaging (visualization and segmentation of bones, tissues etc.).

While today most of the image sensors still acquire and produce 2D images, it is becoming more and more clear that the future is three-dimensional. Lately, the 3D imaging and display technology is also knocking on the doors of our homes for entertainment purposes, e.g. Microsoft Kinect camera (Figure 1.1) and stereo TV displays. In recent years, we witnessed rapid development in 2D digital imaging technology, and its capabilities keep expanding. Today, a digital camera is being carried in every pocket and we digitize our world on daily basis and share our photos on the world web, from which 3D structures can be recovered (Figure 1.1). On the other hand, the 3D imaging technology has not yet reached its potential yet and there is still room for research and development in this area. Keeping that in mind, a very promising approach to 3D digitisation of our world are image-based approaches, not only due to the availability of the sensors, it is also biologically motivated. It is well known that we, humans, use two eyes for

the depth-perception of the world. Even using a single eye, we are able to perceive the shape of real-world object, based on their texture and shading cues, motivating the development of computer vision methods.

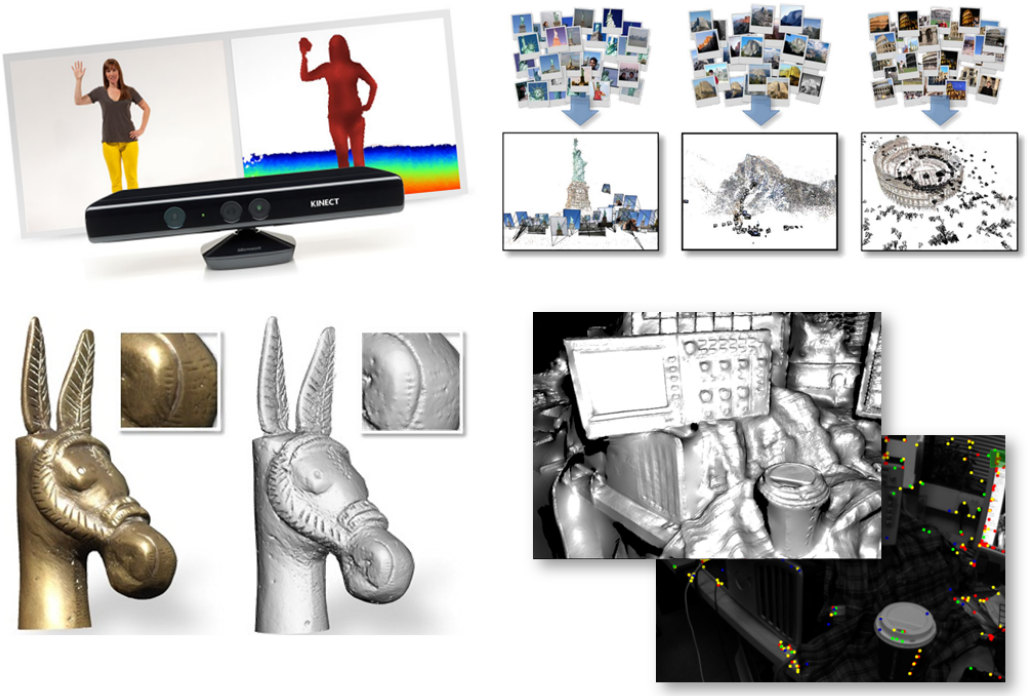

Figure 1.1: Microsoft Kinect, a consumer 3D camera *(top-left)*, 3D scene reconstruction from community photo collections [Sna09] *(top-right)*, high-precision image-based 3D reconstruction of a metal figurine [WRO+12] *(lower-left)* and DTAM: Dense Tracking and Mapping, an real-time single-camera based indoor environment 3D reconstruction and tracking method [NLD11] *(lower-right)*.

In spite of the successful development in the area of the 3D reconstruction research in the past years, researchers are still facing many challenges. While methods addressing lambertian and nearly lambertain objects can nowadays produce very faithful and accurate 3D reconstructions in controlled environments and there exist methods that help to robots infer the geometry of the real-world scene using simple RGB-cameras (Figure 1.1), most of the methods must still admit a trade-off between speed of acquisition and accuracy (e.g. accurate laser scanners versus fast time-of-flight cameras). Furthermore, highly precise 3D reconstruction usually requires controlled setups, at least in the case of image-based methods. The methods based on triangulation of features are fast, but cannot achieve a high precision and may completely fail in some areas. These disadvantages can be resolved with the help of controlled illumination - either helping to find im-

age correspondences (structured light) or relying on using (controlled) shading information in the problematic surface areas.

The objects exhibiting complex reflectance behaviour still present many challenges. While the lambertian assumption was in recent years successfully relaxed to wider classes of objects, e.g. glossy materials, there are still many materials, on which traditional methods will completely fail. Especially difficult are highly-specular (mirroring) objects, transparent (refractive) objects, translucent objects and heterogeneous objects, consisting of mixed materials. An overview of state-of-art developments, addressing this kind of materials, is given in [IKL+08].

For highly-specular and refractive objects, it is well known that they reflect most of the light into one specific direction, hence they do not have their own appearance - it differs fundamentally based on the viewpoint. For that reason, traditional methods, such as laser scanners, time-of-flight cameras, multi-view stereo and structured light systems, will fail to reconstruct such objects. One possible way to address the problematic objects is to explore the use of other visual cues, for example, silhouette information. However, these are usually hard to obtain in real-world scenarios. Additionally, using only this cue, concavities cannot be recovered. More promising visual cues are the shading information. Shading is a very important visual cue for the recognition of the objects. Looking at the Figure 1.2 it becomes clear that it fundamentally helps us to perceive the shape and depth of an object. Although it may not be possible for an observer to make an accurate

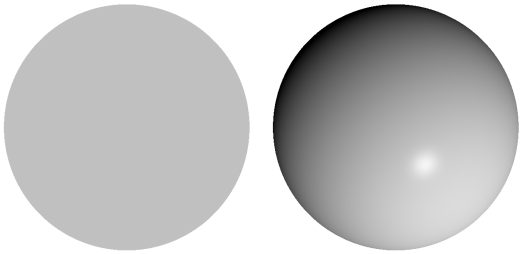

Figure 1.2: Shading is a very powerful cue for recognition of objects shape.

prediction on the depth of the object (Figure 1.2), we are able to infer at least partial information about the surface shape with the help of shading information. In literature, the use of shading cues to estimate surface orientation (normals), and consequently, the geometry, has already been demonstrated (shape-from-shading [Hor70], photometric stereo [Woo89], shape-from-specularity [CGS06]), but with certain limitations. These approaches usually help perceiving the shape from a certain perspective, but the question, how to infer the full geometry of an object based only on surface orientation information remains open. The surface orientation information (following from the shading information) has been successfully used for full 3D reconstruction in combination with other cues (multi-view stereo

[WWMT11], structured light [WRO+12]), but there are cases, where we do not have a commodity to rely on such prior data, for the case of mirror-like objects, for example. Methods for multi-view integration of normal fields have been reported [CLL07, Dai09], but with limited success on the real-world data and still relying on silhouette information, that would again be hard to automatically and reliably detect in case of mirroring objects.

The surface gradients (normals) can actually be treated as an intermediate representation and they can be obtained from several normal-estimation based techniques for many different materials. Having a robust multi-view normal integration algorithm, it would be possible to model 3D shapes of objects, based only on shading information. Hence, this carries a potential not only for 3D reconstruction of complex cases of objects, like highly-specular (mirroring) objects, but also for the reconstruction of inhomogeneous objects. An attractive option for addressing such objects would be the use of different normal estimation techniques for different parts, consisting of different materials and to use the multi-view normal integration algorithm to integrate all those information together.

In the scope of this thesis, we are addressing the precise 3D reconstruction of geometry in the controlled environment (dome-setup). The specific goal was first to develop a novel, outlier-robust multi-view normal field integration algorithm and first demonstrate its performance on classic photometric stereo data, captured from nearly lambertian objects. We show, that such an algorithm can be used for precise 3D reconstruction of arbitrarily-shaped, highly-specular objects, even in the presence of inter-reflections. In addition, our result on highly-specular data can be considered as state-of-the art in the area of precise 3D reconstruction of such specular objects and we believe, that it could be successfully applied on a much wider range of normal estimation based techniques.

1.1 Contribution

The work on this thesis resulted in two new contributions in the area of precise, controlled environment based 3D reconstruction. The main contribution is a new algorithm for multi-view normal field integration, which is to the best of our knowledge the first one, successfully applied on the data, captured in the real-world setup. In relation to that, a novel approach, based on feature space analysis is proposed for computation of surface consistency and vector field, that provides the surface in-out constraints. The final surface reconstruction problem is solved using convex relaxation based variational technique, recovering surface that fits best to the reconstructed vector field.

The second contribution is the application of the algorithm in the field of 3D reconstruction of highly-specular, mirroring objects. Surfaces of specular objects

are recovered by integration of hypothesised normals by the proposed multi-view normal field integration algorithm. For capturing the normals, we used a method for normal estimation, based on a calibrated environment. In relation to that, we designed a turn-table based setup in a way that display screens, used for illumination of the object, are partially visible to the cameras. This method is very precise and does not require capturing additional data or placing mirrors in the setup for the calibration. For establishing the correspondences between illuminated objects and scene points illuminating them, we used structured encoding of the light sources based on Grey codes. In the area of 3D reconstruction of highly-specular object, we believe, that this result is state-of-the art.

1.2 Thesis Outline

Chapter 2 gives a brief overview of the existing 3D reconstruction methods with focus on normal estimation and integration based techniques and prior work on 3D reconstruction of highly-specular objects.

Chapter 3 presents a new, outlier-robust method for the multi-view normal field integration and discusses the details of the proposed technique. Additionally, implementation of the algorithm is discussed.

Chapter 4 explains a new method for multi-view reconstruction of highly-specular objects, based on multi-view normal estimation and integration. Explained is the setup for capturing the data, the light map acquisition and, finally, it is explained how hypothesised normals are integrated.

Chapter 5 demonstrates the reconstruction results. First, the results obtained on perfect synthetic (OpenGL normal renderings) data, are discussed. Then, performance on the real-world normal fields, computed via photometric stereo technique is evaluated. Finally, results, based on shape-from-specularity data are presented.

Chapter 6 concludes the thesis and discusses future work and applications.

CHAPTER 2

RELATED WORK

In this chapter, we give a brief overview of 3D reconstruction techniques and related work. First, general 3D reconstruction techniques are reviewed. Both **passive** and **active** methods are explained with the focus being on the latter where our approach belongs. In addition to reviewing both, **triangulation** and **normal estimation** based methods, we shortly discuss the **hybrid methods** and **normal integration** techniques for classic, single-view and multi-view scenarios. This chapter is concluded with an overview of methods, dealing with reconstruction of highly-specular objects.

2.1 Passive Methods

Passive methods are dealing with geometry reconstruction, based solely on measuring reflected radiance from the object, without influencing its appearance (by changing light conditions, for example). The inputs in this case are usually images, captured by camera sensors. The goal is to infer the geometry of the object or scene, based on a pure vision-based analysis of the images. The core of these methods is the determination of the correspondences between individual pixels across the views, although incorporation of additional cues is possible, e.g. silhouettes [Lau94] or focus/defocus [NN94, FS02].

2.1.1 Triangulation-based Methods

The triangulation techniques methods are based on the simple, biologically inspired idea, that when observing a feature in at least two views, it is possible to determine its depth. Observing a feature in a single view defines a set of possible solution along the viewing ray (i.e. a ray from the optical centre of the sensor through the projection of the feature on the image plane). While a single view is not sufficient for the determination of the exact 3D pose of the feature, a second (calibrated) view introduces necessary constraints to resolve the ambiguities and

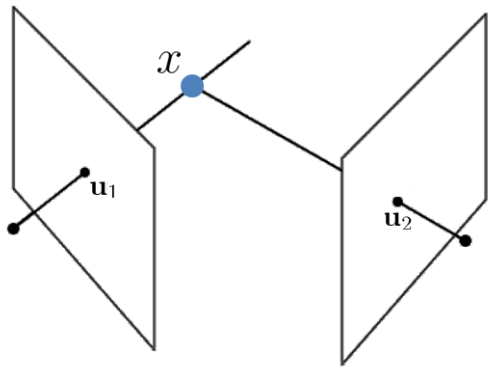

Figure 2.1: A feature at spatial point **x** is observed at image location $\mathbf{u}_1$ in the first camera and $\mathbf{u}_2$ in the second camera. From two calibrated cameras, it is possible to determine its 3D position.

the 3D location of the feature corresponds to the intersection of the viewing rays from both views. This idea is illustrated in Figure 2.1. In principle, introducing additional views decreases the ambiguity in the 3D location of the feature.

In reality, the difficult problem in context of passive methods is finding the corresponding features across the views. The reasons for this arise from the fact that in reality, many areas are not well-textured (i.e. walls, tables, etc.) and hence, do not have easily distinguishable appearance. In addition, the visual impression of the observed features may change rapidly due to changing illumination. In particular, objects exhibiting complex reflectance behaviour are problematic, since they are more prone to changes in appearance when being observed from different view points, highly-specular objects can be considered as the extreme case. For current methods, a general assumption is that objects exhibit lambertian reflectance behaviour. Most of the triangulation-based methods are hence addressing the problem of matching the observed surface points between different views in order to be able to reliably triangulate surface points.

A family of popular methods are the **multi-baseline** methods, where depth maps with respect to a subset of views are computed. Using the epipolar constraint, the best match between intensity values on the epipolar line between two images is computed, using e.g. the sum of squared distances (SSD) or normalized cross-correlation (NCC). Then, from the identified correspondences, depth is computed with respect to the reference image based on the triangulation. The depth maps have to be consequently merged to recover the full 3D geometry, using e.g. [CL96].

In case of **volumetric methods**, the object geometry is represented implicitly in a discretised volume. The task of 3D reconstruction is then addressed as

a volumetric segmentation problem. The early approaches are based on computation and segmentation of a photo-consistency function $c(\mathbf{x}) : \mathbb{R}^3 \mapsto \mathbb{R}$, denoting how consistent the image back-projections to the spatial point $\mathbf{x} \in \mathbb{R}^3$ are, e.g. [SD97, KS00, YPW03]. In general, regularized methods based on energy minimization of an appropriate energy functional consisting of photo-consistency function as a data term and regularization (usually minimal-surface based) term are able to deal better with raising difficulties, such as untextured areas, outliers due to violations of lambertian surface, sensor noise, etc. Several authors proposed many different energy models and optimisation techniques. For example, there are volumetric approaches based on active-contour methods [ES04], level-set methods [FK98, KBC06, CLL07, LQ05], graph-cut based global optimisation [KZ02, VETC07, YAC06] and convex relaxation based optimisation methods [KBC06, KKB$^+$07].

2.1.2 Shape from Silhouettes

In case of shape-from-silhouettes methods [Lau94], the object of interest is segmented from the background on the 2D image domain and represented as binary image. Then, the silhouettes are intersected in the volume via the voxel carving method. In this case, a bounding volume corresponding to the object is discretised as voxel-grid or octree. Then, each voxel is back-projected to each silhouette. Once a voxel is back-projected to the area of the image marked as background, it is inconsistent with the silhouettes and rejected from the volume. Although this method is simple and fast, silhouettes are difficult to compute in reality. Furthermore, concave regions cannot be recovered and high-quality results are in general not possible to obtain.

2.2 Active Methods

The base idea in the active methods is to measure the objects radiance after emitting a light towards it, either visual spectrum of the light (e.g. structured light systems, laser-stripping methods) or to human observers invisible spectra (e.g. time-of-flight cameras and Microsoft's Kinect use infra-red light). Most of these methods compute either depth maps with respect to image-sensor location or point clouds. For the recovery of the full geometry of 3D objects, measurements require post processing. In case of range scans, individual depth maps can be combined using e.g. approach presented in [CL96]. In case of point clouds, usually it is necessary first to align clouds from individual scans, using e.g. ICP algorithm [BM92]. To recover surface representation of the shape, one of the methods for shape fitting to point clouds should be applied to the merged point cloud

[HDD+92, CT11, LB07, KBH06, MPS08, OBA+03].

2.2.1 Time-of-Flight Camera Based Methods

The base components of time-of-flight cameras are illumination unit, that emits an infra-red light signal towards the object, and an image sensor, that measures the radiance emitted back toward the sensor. Since the speed-of-light is known precisely, it is possible to compute depth for each individual pixel based on the time difference between the time signal was emitted towards the object and the time at which it was reflected back to the image sensor (or alternatively, the distance can be also computed from a phase-shift between emitted and measured signal). While time-of-flight cameras are very fast, capable of perceiving depth of the environment with respect to the camera in real-time, their distance and lateral resolution is rather limited, presenting certain limitations for precise 3D reconstruction of objects. Furthermore, measurement errors and noise due to light interference further reduce the quality of the scans. Approaches for 3D reconstruction of objects, that attempt to overcome limitations in depth-image quality however exist [CSC+10].

2.2.2 Triangulation with Structured Light Patterns

The idea behind such methods is to simplify correspondence detection between individual pixels from different views by projecting structured patterns to the object before capturing the images. The base component of structured patterns based setups are projectors, playing the role of an emitter for displaying structured patterns (in general, in the visible part of EM spectra), and calibrated cameras (at least two). By observing structured patterns, projected on the object, correspondence detection is greatly simplified. Correspondences can then be solved by decoding the patterns, projected on the object. Matching the decoded codes relates pixels between the images. After that, the 3D surface positions can be triangulated. These methods can overcome issues of not-well textured areas and changes in illumination, but are still sensitive to highly reflective/refractive and translucent objects. Furthermore, solving the correspondences is usually limited to the projector resolution which is in general much lower than the camera resolution.

The structured pattern emitting approaches mostly differ by the type of patterns displayed (an excellent overview of coding strategies is given in [SPB04]). In general, the pattern types can be divided into **single-shot** projections and **multi-shot** projections. The single-shot based techniques (e.g. rainbow pattern [TI90], De Brujin sequences [SBM98], M-Arrays [MOC+98]) emit only one pattern per captured view. The advantage of these approaches is a shorter acquisition time, enabling to measure even active/moving scenes. However, in general, the decoding stage is more complex and error-prone.

The temporal sequences on the other hand, such as binary codes [PA82], Gray codes [ISM84] or phase-shift methods can provide a reliable decoding, and consequently, high-precision reconstructions. Since at the encoding stage, a series of those patterns has to be displayed (and captured) per-view, these encoding strategies are not appropriate for dynamic scenes, but are a favourable choice in the static setups. With binary codes and Gray codes, a series of vertical and horizontal image-encoded bit sequences is displayed, where white regions of the image pattern correspond to the value of 1 and the black region to 0. For the image of size width $\times$ height, $\lceil \log_2(\text{width}) \rceil + \lceil \log_2(\text{height}) \rceil$ images have to be captured. Then, for each view, for each pattern and for each pixel it must be identified, whether the white or black portion of the projected pattern illuminated it. From the sequence of bits, each pixel is assigned a unique id, that matches the id in images from other views. The advantage of using Gray codes instead of binary for encoding is that adjacent codewords differ only in one bit, making the decoding more robust.

2.2.3 Overview of Gradient Estimation based Techniques

Methods based on **surface gradient estimation** are, in contrast to triangulation based methods, usually able to preserve high-frequency details of the surface. Another advantage of these methods is that some of them can exploit prior knowledge about the materials, for example, there are techniques for normal estimation of lambertian surfaces, e.g. photometric stereo [Woo89]. There exist approaches, that work successfully on highly-specular materials, e.g. shape-from-specularity approaches [CGS06, FCMB09]. There are also normal estimation techniques that can address normal estimation of very large range of materials [ZBK02, GCHS05]. Furthermore, these normal estimation techniques usually work in a complementary manner. Techniques, designed for lambertian surfaces sustain a quality loss as the material is exhibiting more specular reflectance behaviour. The situation for shape-from-specularity approaches is just reverse, hence the combination of different normal estimation approaches and consecutive normal integration seems to be very prominent for handling materials with arbitrary reflectance behaviour and still preserve fine surface details.

However, these methods require a normal integration step in order to recover the surface, which is in practice problematic. Especially challenging is the capturing of full 3D surface geometry based solely on normal estimation techniques. Approaches for multi-view integration of normal fields do exist [CLL07, Dai09], but these approaches were successfully applied only on synthetic data. It appears, that to this date, a robust algorithm for multi-view integration of measured normal fields from real-world measurements do not exist and the central task of this thesis is to close this gap.

Shape-from-Shading

The shape-from-shading approaches attempt to recover the surface information based purely on shading information from a single view. The first hand-computation based methods for surface depth estimation from shading date back to 1951 [Dig51] and the first computer vision approaches for recovery of the surface from shading information were due to Horn [Hor70]. The most basic assumptions in most of shape-from-shading methods are known light source positions, lambertian reflectance behaviour of the surface and constant albedo. The goal of these methods is then to obtain depth information from single image based on the observed intensities by the image sensor. The shape orientation information can be obtained from single image by considering knowledge about the image formation (forward model). Obtaining the surface orientation information is then the inverse problem of the forward model (lambertian assumption)

$$\mathcal{I}(\mathbf{u_x}) = \mathrm{k_x} \cdot (\mathbf{n_x}^\mathsf{T} \cdot \mathbf{l}), \tag{2.1}$$

where $\mathcal{I}(\mathbf{u_x})$ is the observed intensity of the image $\mathcal{I}$ at the pixel $\mathbf{u_x} \in \Omega$, $\mathbf{n_x}$ is the normal of surface point $\mathbf{x}$, projected to image location $\mathbf{u_x}$, $\mathrm{k_x}$ is the surface albedo at surface point $\mathbf{x}$ and $\mathbf{l}$ is the light direction vector. The first method [Hor70] attempt to solve the problem by deriving a PDE from the image-formation model (2.1), however, the original problem is severely ill-posed [DP00, Koz97] and the methods do not produce high-quality results in practice. As often in case of ill-posed problems, variational approaches were proposed [HB86]. The main idea of these methods is to optimise an energy functional, minimizing the squared error between observed intensity and intensity produced by estimated normal estimates. By introducing additional assumptions, e.g. the smoothness constraint, there is a unique solution to the minimization problem, but as pointed out in [FC88], the resulting normal field might not be integrable, posing challenges to normal integration algorithms.

Classic Photometric Stereo

The classic photometric stereo is an approach for estimating the gradient (normals) of a surface exhibiting lambertian reflectance behaviour. However, in contrast to shape-from-shading techniques, photometric stereo is a well-posed problem, i.e. the exact orientation of the surface can be computed by considering more than one image at the input. As pointed out in [Woo89], to uniquely determine the normal seen at pixel $\mathbf{u_x}$, at least three equations are needed for three unknowns. Hence, three images taken under three different (known) light source positions suffice for the computation of the normals. In reality, due to numerous non-lambertian effects (specular reflections, shadows, noise, etc.), more images

produce more accurate normal estimates and normals can be computed by linear least-squares.

Prior to the computation of the normal field from a single view, geometric and radiometric calibration must be performed, i.e. light source positions must be known, images should be linearised and corrected for possibly varying light source intensities and light fall-off. Assuming n light source positions in the scene, n images are captured, one per active light source, see Figure 2.2. The

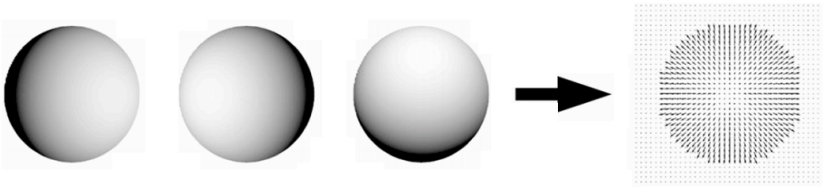

Figure 2.2: Three images of a sphere, taken by a single camera, under different illumination positions uniquely determine normal of the surface (*left*). The normal field, visualised as *needle map* (*right*) (image source: [FY07]).

normal at the pixel $\mathbf{u_x} \in \Omega$ can then be computed from the observed intensities by linear-least squares fitting. The $1 \times n$ image intensities vector for pixel $\mathbf{u_x}$ is defined as $\mathbf{I}_{\mathbf{u_x}} = [\mathcal{I}_{1,\mathbf{u_x}} \dots \mathcal{I}_{n,\mathbf{u_x}}]^\top$, the $n \times 3$ light direction matrix is

$$\mathbf{L} = \begin{bmatrix} - & \mathbf{l}_1 & - \\ - & \mathbf{l}_2 & - \\ . & . & . \\ . & . & . \\ . & . & . \\ - & \mathbf{l}_n & - \end{bmatrix}, \tag{2.2}$$

and the 1×3 vector we are solving for is $\mathbf{g}_{\mathbf{u_x}} = [g_{1,\mathbf{u_x}}\ g_{2,\mathbf{u_x}}\ g_{3,\mathbf{u_x}}]$. Using multiple light sources and images, equation (2.1) may be rewritten as

$$\mathbf{I}_{\mathbf{u_x}} = \mathbf{L}\mathbf{g}. \tag{2.3}$$

For the pixel $\mathbf{u_x}$, the vector $\mathbf{g} = k_{\mathbf{x}} \cdot \mathbf{n_x}$ can be computed as:

$$\mathbf{g} = \mathbf{L}^{+}\mathbf{I}, \tag{2.4}$$

where $\mathbf{L}^{+}$ is the Moore-Penrose pseudo-inverse of $\mathbf{L}$. The surface normal estimate is then $\mathbf{n} = \frac{\mathbf{g}}{\|\mathbf{g}\|}$ and the corresponding albedo is $\|\mathbf{g}\|$.

The classic method [Woo89] was also generalized to unknown lighting conditions [BJK07]. In addition to linear-least squares based method, there also exist more sophisticated robust photometric stereo approaches, e.g. [WGS+11, Aiz12].

2.2.4 Hybrid Techniques

The hybrid techniques usually combine triangulation-based and normal estimation techniques and take advantage of both approaches. First, the rough (prior) geometry is constructed using the triangulation-based approaches. Using that geometry, normals are estimated, and based on that information, surface can be corrected and refined. However, in order to get a good result, a reasonable result from the first step is required, otherwise these methods in general will not converge to the correct solution.

The combination of multi-view stereo technique and shape-from-shading techniques was suggested in [BZK86]. A high-quality (comparable even to laser scans) approach, combining MVS techniques with shape-from-shading, was demonstrated in [WWMT11]. They use classic multi-view stereo for an initial geometry estimate and refine the surface based on shading information, using unknown (uncalibrated) lighting conditions. The correction based on shading information is especially beneficial in low-textured areas, where the MVS method is not able to reliably estimate depth information.

An attractive idea of combining MVS technique and photometric stereo was explored in [WLDW11], where an initial, rough shape is also recovered by a MVS method and used for normal estimation via photometric stereo. Then, the initial shape is refined based on normal information, again helping especially in low-textured areas. A similar combination of MVS and photometric stereo was used for 3D reconstruction using a hand-held camera in [HYJK09], where the point light source was attached to the camera for the illumination of the object.

Furthermore, an interesting method was proposed in [WRO+12], where the authors combine structured light with Helmholtz normals [ZBK02]. Both, structured light and Helmholtz normal information is used in a single optimisation step, based on minimizing a minimal-surface based energy functional with a convex relaxation based method [YBT10] in an octree-discretised volume. In addition to being very robust with respect to a large variety of materials (lambertian, glossy, etc.) a great level of surface details can be recovered due to the use of normal information.

2.3 Single-View Normal Field Integration

The task of single-view normal field integration algorithms is to reconstruct a partial surface or depth map (2.5D reconstruction) from a single normal field as illustrated in Figure 2.3. The problem of normal field integration is difficult due to the fact that in general the observed or computed normal fields are not integrable. Not every vector field is a gradient of a function (a necessary condition for a vector

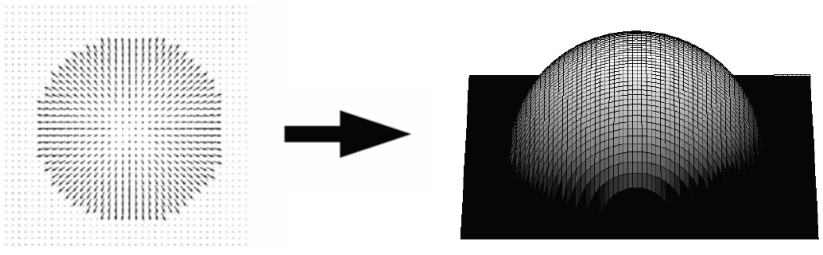

Figure 2.3: A normal field and respective partial reconstruction (image source: [FY07]).

field to be gradient of a function is that its curl equals zero). In general, due to sensor noise, outliers or systematic errors the measured gradient fields will not be integrable and simple path integration will accumulate errors at the integration stage and produce highly erroneous results.

To address this problem, there is a line of methods that enforce integrability. One of the most robust methods by [FC88] works by projecting normals into the integrable space (in this case, Fourier basis functions are used) and then integrable normals can be integrated by applying path integration on the image domain. In literature, many different basis functions were proposed, e.g. shapelets, [Kov05], wavelets [HLKF95].

There exist also methods, that do not require the normal field to be integrable but rather try to reconstruct the surface that fits normal field best. Probably the first of these methods is the variational approach presented in [HB86], which is based on minimizing an energy functional and penalizing squared differences between surface normals and observed normals. The Euler-Lagrange equation is derived from the energy functional and solved using Gauss-Seidel relaxation. The method proposed in [SCS90] is based on solving the Poisson equation and for the technique presented in [HO11] the direct linear-least squares method is applied instead of variational calculus methods.

2.4 Multi-View Normal Field Integration

Multi-view normal field integration is a relatively new topic and there are not many methods addressing this problem. In this case, many normal fields, taken from different views around an object are given and the goal is to reconstruct the full 3D, closed and differentiable surface of the object. In the literature, the first work addressing the problem was published in [CLL07]. In their work, the authors first compute the visual hull of the object based on the silhouettes of the normal fields using the shape-from-silhouette approach [Lau94]. They solve the problem

in a variational framework, formulating the energy functional consisting of regularization (minimal surface) term and data (flux) term, from which they derive a geometric PDE that describes the surface evolution. They solve the geometric PDE by iteratively evolving the surface with level-sets [OS88]. The geometry from the previous step is used for the computation of the visibility function and it is being updated in each iteration. In their evaluation, the authors show that their algorithm works on synthetic data and is resilient to additive Gaussian noise. They also demonstrate results on photometric stereo [Woo89] data, but it is generated synthetically (OpenGL renderings), so it is not clear how their algorithm performs in the presence of outliers and systematic errors.

In [Dai09] the multi-view normal field integration problem is formulated in terms of Markov random fields (MRF). In this work, surface is represented an binary indicator function in a grid and uses graph-cuts [BVZ01] to compute the discrete in-out voxel labelling that maximizes the joint probability of the MRF. The probability of the surface is formulated at each voxel using three energy terms, a surface prior (in practice, a visual hull is used), a normal disparity term that penalizes normal deviations and a novel surface orientation constraint. To overcome discretisation artefacts, additional step is applied, similar to the technique described in [CLL07]]. For the evaluation, the author presents excellent result on the synthetic data set, but on the real-world data, where normals are estimated by photometric stereo [Woo89], the final reconstruction does not converge towards the desirable solution and it seems to be quite close to the visual hull of the object.

2.5 Reconstruction of Highly-Specular Objects

While 3D reconstruction of nearly lambertian objects is a well-addressed problem and many successful methods for full 3D reconstruction of objects exist, reconstruction of highly-specular 3D objects is still a widely open topic. The main difficulty of measuring this kind of objects lies in a fact that they reflect nearly all incoming light in one direction. Thus, highly specular objects do not have their own appearance but rather only reflect the surrounding environment.

However, knowledge about image formation model for the specular surfaces can be used for 3D reconstruction purposes. It is well known that at surface point belonging to a specular object, an incoming light ray will be reflected in the near proximity of the direction of perfect reflection, depending on how smooth the surface locally is, see Figure 2.4. The methods for 3D reconstruction of specular objects can be divided into two groups. A family of methods, that attempts to reconstruct surface based on observations how virtual features of the environment move, are refereed to as **shape from specular flow** methods. For that, a dense collection of views of the object are needed, typically a video sequence. Another

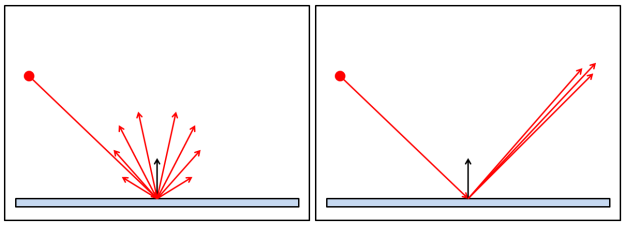

Figure 2.4: While lambertian surfaces *(left)* due to rough surface micro-structure reflect light uniformly and give view-independent appearance to the surface, locally smooth specular surfaces accumulate reflections around the direction of perfect reflection.

branch of methods makes use of the calibrated environment. By observing the reflections of the environment in the objects and relating virtual features to their corresponding 3D location, it is possible to compute the orientation of the surface, since the normal is a bisector of the view and light direction at surface points, assuming that the surface is specular. Then normal integration methods can be used to integrate the normals and reconstruct the surface.

2.5.1 Methods Based on Specular Flow

The methods, addressing 3D reconstruction of specular surfaces based on specular flows assume distant and unknown environment. The main idea is to capture video sequences of either a moving object or a moving environment and use computed optical flow (in this case, called specular flow) on the specular object between image sequences to reconstruct the surface. Then, the shape can be recovered by solving a system of partial differential equations, that are usually subject to initial conditions. One of the main difficulties of this approach is an accurate computation of optical flow, which is an ill-posed problem by itself.

The first approach to show how specular flow can be related to 3D geometry of objects [RB06] demonstrated results on a specular sphere, covered by some diffuse markers. In [AVBSZ07], a new theoretical framework for the reconstruction of the specular objects based on specular flows is introduced. The authors assume a static observer, an orthographic projection and a moving environment. They formulate the reconstruction of the specular shape based on solving a system of PDEs, that relates the observed specular flow and the shape of the specular object. In addition to theoretical contributions, they also demonstrate some practical aspects on a mirroring sphere. In successive work [ABS11], they formulate the reconstruction of 3D objects from specular flow in a variational framework and they estimate flow and recover shape simultaneously.

2.5.2 Methods Based on Calibrated Environment

In a nutshell, methods based on a calibrated environment take advantage of precise knowledge about the environment. By relating the images of the surface to the spatial locations which illuminated surface, it is possible to compute surface normals (orientations) based on the law of reflection and fit surface to the observed normal fields.

In the context of mesostructure reconstruction of nearly flat, specular objects, the authors of [CGS06] proposed a setup, where they take a sequence of images under different illumination conditions. In their approach, a hand-waving light source was used (distant illumination) and four calibration spheres were placed around the object in order to compute the light source position in every captured image frame. In every frame specular reflections are detected by thresholding the captured image. By identifying highlighted pixels, and having a calibrated camera, normal fields are computed based on the law of reflection - the normal is a bisector between view and light vector at the image locations, where highlights have been detected. To obtain partial surfaces of the objects, normal integration is applied to the resulting normal field. The approach published in [FCMB09] is based on a similar idea, only in this case, to compute normal fields, structured illumination is used. In particular, a sequence of Gray codes is displayed on a LCD screen, and by decoding them, the observed surface points can be related with respective 2D locations on the display screen. Having the screen calibrated, a 3D location for each observed surface point can be identified. The normal field and the resulting surface can be recovered then in same fashion as in [CGS06]. However, since a near illumination is used, their reconstruction model is ill-posed and it is unclear how does it effect the results.

For 3D surface reconstruction of objects with more general geometry, the authors of [SWN88] used an led-grid, illuminating a specular object. By successfully switching LEDs on and off while capturing the images, they recover a reflectance map (in the following chapters, we will call it light map), relating the 3D scene points (LEDs), causing highlights, on base of which normals can be recovered and integrated. Initially they use single view and distant light assumption (distance from the object to the LED array is large with respect to the objects size), but they also proposed using multi-view constraint on normal orientation in case the near-light is used. When distant light assumption is violated, original problem becomes ill-posed and light-view pair define a family of possible surfaces, that gave rise to the observations. However, the ambiguity can be resolved by adding another view and the surface normal is the one where both (or all) views match. They show profiles of reconstructions of a specular sphere and solder joint for evaluation of their algorithm.

A similar approach to resolve ambiguities raising from near-illumination was

employed in [BS03]. The authors formulate their approach in a volumetric, space carving framework and use a target, on which a structured pattern is printed to illuminate the object. Having the display target and the camera geometrically calibrated, they compute normal hypotheses across the volume. Each view-light direction provides a family of normal hypotheses. Hence, each view suggests a normal hypothesis at each voxel. Then, they define normal disparity measure and carve away all voxels, for which the angular distance between hypothesised normals is large. In the evaluation, they demonstrate that the reconstructed voxels fit the mirroring spoon they consider. A similar idea is applied in [NWR08], where the authors perform a triangulation based on specularity consistency across the views. Observed surface points are related to the scene points that illuminated them based on displaying Gray codes on the calibrated display (LCD) screen, illuminating the object. After triangulation, normals are again estimated at the computed depths and based on the new normal information, the surface is refined (a similar iterative scheme was used in [TLGS05]). The authors of [NWR08] demonstrate their reconstruction results of nearly flat objects (e.g. coins, curved mirrors) and the authors of [TLGS05] demonstrate their results on a curved mirror as in addition to a metallic, slightly curved plumbery object.

In [BW10], an excellent overview of principles of shape-from-specularity is provided and the author also discusses the nature of the ill-possessedness of the problem and suggests several regularization approaches. The author points out, that ambiguities can be resolved by incorporating additional information, and the multi-view normal constraint is discussed as only one of the possible solutions. Furthermore, many useful practical tips for setup design and the computation of the light maps are stated.

CHAPTER 3

MULTI-VIEW NORMAL FIELD INTEGRATION

3.1 Problem Statement

In this section, a new algorithm addressing the classic problem of multi-view normal field integration is explained. Just as in the original formulation of the problem [CLL07], we assume a setup, in which multiple cameras are placed around the object we would like to reconstruct and the cameras provide normal estimates of the observed object (Figure 3.1). More formally, it is assumed we have κ_c calibrated cameras, oriented towards the object of interest. Each camera C_i, $i = 1 \dots \kappa_c$ comes in a pair with an image $\mathcal{I}_i : \Omega \to \mathbb{R}^3$, where $\Omega \in \mathbb{R}^2$ is the image domain, and a perspective projection matrix $\mathbf{P}_i$. Each image $\mathcal{I}_i$ contains color-coded normals of all surface points $\mathbf{x} \in \partial S$ of a solid $S \subset \mathbb{R}^3$ (i.e. the object to be reconstructed), visible in camera C_i.

It is assumed that the normals were estimated by a normal estimation process, e.g. photometric stereo and its generalizations [Woo89, HS05, GCHS05], Helmholtz normal estimation technique [ZBK02] or shape-from-specularity based approaches [CGS06, FCMB09]. We treat the normal estimation process as a black box: the method used for estimation is independent of the integration process and we assume that normals have already been estimated. In a real-world scenario, accurate normal estimation of a surface is a very challenging task, that is why we assume that normal field estimates are corrupted by noise and outliers. Sources of the noise are usually due to the image acquisition process, e.g. noise originating from the sensor, circuitry of the digital camera or too short/too long exposure times. The most notable sources of systematic errors and outliers are:

- Systematic errors due to imprecise camera calibration
- Reflectance model violations
- Setup assumptions violations (e.g. distant light source)
- Shadows

Figure 3.1: Visualization of a hypothetical setup: an object, surrounded by several cameras. Each camera provides normal estimates for the surface points, visible to the camera.

- Inter-reflections and other global illumination effects.

For that reason it is not only very important for the normal integration process to be resistant to noise, robustness towards the outliers is essential as well.

After the normal estimation process, the value for the pixel $\mathbf{u}_i = (u_x, u_y)^T \in \Omega$ corresponds to $\mathcal{I}_i(\mathbf{u}) = f(\tilde{\mathbf{n}}_{i,\mathbf{x}})$, where $\tilde{\mathbf{n}}_{i,\mathbf{x}}$ is a normal estimate of the true normal $\mathbf{n}(\mathbf{x})$ at surface point $\mathbf{x} \in \partial S$ from camera C_i and $\mathbf{u}_i = \mathbf{P}_i\,\mathbf{x}$ is a projection of $\mathbf{x}$ to the image plane of camera C_i. The function $f : \mathbb{S}^2 \mapsto \mathbb{R}^3_+$ is a linear mapping from the normal space to the RGB space that encodes normal estimates to color channels of the image. In following chapters, images containing normal information will be also referred to as *normal fields*, $\mathcal{N}_i$.

Assumed is the following projection process: a perspective projection and a pinhole camera model. The matrix $\mathbf{P}_i$ denotes a projection matrix of camera C_i: $\mathbf{P}_i = \mathbf{K}_i\,[\mathbf{R}_i\,|\,\mathbf{t}_i]$. For the camera C_i, the matrix $\mathbf{K}_i$ is 3×3 matrix of intrinsic camera parameters that maps points from the camera coordinate frame to the image plane frame. The matrix $[\mathbf{R}_i\,|\,\mathbf{t}_i]$ is a 3×4 rigid-transformation matrix, that maps points from the world coordinate frame to the camera coordinate frame. It is described by a rotation matrix $\mathbf{R}_i$ and a translation vector $\mathbf{t}_i$.

The goal of the multi-view normal field integration problem is to recover a full, closed and differentiable surface ∂S of the solid S that led to normal observations, based solely on normal information, i.e. normal fields $\mathcal{N}_1 \dots \mathcal{N}_{K_c}$, and camera parameters $\mathbf{P}_1 \dots \mathbf{P}_{K_c}$. Additionally, for most applications, it is desirable to recover the surface in a form of a triangular mesh.

Most of the algorithms dealing with 3D reconstruction using normal information from several views tend to use other visual cues. There exist hybrid approaches (explained in Chp. 1) that use geometry from multi-view stereo (MVS) [WLDW11] or structured light [WRO+12] as a prior knowledge. These approaches usually recover the rough geometry by triangulation or correspondences-based approaches, estimate normals based on the initially recovered geometry and use normal information to refine surface afterwards. Even in the first work addressing the multi-view normal field integration problem [CLL07], silhouette cues are used to recover initial geometry, that is used as a initial guess to the level-set optimisation method. Their optimisation approach is, however, not a global optimisation method and is sensible to that initial guess. Also in [Dai09], visual hull computed from silhouettes is used as in-out constraint and as a initial guess to an Expectation-Maximization based normal disparity computation.

In the proposed approach, only normal information is used for the reconstruction. Silhouettes are in practice not trivial to recover automatically on the real-world data. Especially problematic are shadowed areas of the object which are highly susceptible to false segmentation and highly-specular areas, which are very similar to the background and hence, very hard to segment. Furthermore, we do not demand that the inputs to our algorithm are complete normal fields. There are cases, where it is not possible to estimate normals on complete projected area of the object to the camera. When segmenting the object based on silhouettes, parts where normals would not have been recovered would be carved away leading to the false reconstruction in the very first iteration.

The rest of this chapter is organised as follows: first the general idea based on variational formulation is highlighted. Then, the outlier-robust method for surface consistency and vector field computation, that provides in-out constraints, is explained. The discussion about the practical implementation of the algorithm concludes the chapter.

3.2 Approach

In a nutshell, this approach is based on the fitting of an implicit function to the surface-consistency scaled vector field, computed by a feature-space analysis of the back-projected normals. To compute the vector field, at each spatial point in the considered volume, back-projected normals from all normal fields are mapped

to the feature space. At the points near the surface, back-projected normals form a clear peak for the parameters, corresponding to the true surface normal, see Figures 3.3 and 3.4. We take the density estimate of the probability density function of the parameters, forming a peak, as a measure of surface consistency. In order to cope with outliers and noise, a very robust analysis of the feature space is essential.

Usually the observed normal fields themselves are non-integrable due to noise, outliers and even holes, where data could not have been recorded. Even if the normal fields would have been integrable, there is no guarantee that the vector field, computed from normal field back-projections is. Therefore, as very common in computer vision tasks, the surface reconstruction problem is addressed in a variational manner. For segmenting objects interior from the background, we formulate an energy functional, consisting of a minimal area term for the regularization and a flux term as a data term. To be able to cope with high-precision reconstruction demands, and on the other hand, with high memory requirements, we employ an octree data structure for the volume discretisation and perform a binary in-out labelling in a spatially continuous setting. To avoid discretisation artefacts in the final reconstruction, an additional smoothing optimisation step is performed.

3.2.1 Variational Approach and Energy Model

The early works addressing 3D reconstruction from multiple images are based on the simple voxel-carving technique, where individual voxels in the discretised volume are being rejected based on a thresholding of the photo-consistency function [SD97, KS00, YPW03]. One of the first works addressing surface reconstruction based on normal consistency was also formulated in the context of voxel-carving [BS03], taking normal disparity as a surface-consistency measure. However, the real-world measurements tend to be incomplete and suffering from noise and outliers, leading to reconstructions containing holes and suffering from over-fitting. In light of these difficulties, variational approaches have been successfully applied in several fundamental computer vision problems, e.g. image segmentation [CV99, CKS97, OS88, MS89] and de-noising [ROF92]. The area of multi-view stereo 3D reconstruction is not an exception. The state-of-the art methods are formulated in a variational framework [FK98, LQ05, KBC06, KKB$^+$07, KC08]. For an excellent overview of variational methods for the purpose of multi-view 3D reconstruction we refer to [Kol12]. The main advantage of these approaches is the simple incorporation of prior knowledge in terms of regularization. By imposing regularity on the preferred solution, over-fitting can be effectively prevented. Furthermore, the holes in the measurements (e.g. unobserved areas or areas where the model fails) can be, depending on the regularization scheme, filled to some degree.

Considering the challenges that the multi-view normal field integration problem is posing (see Sec. 3.1), the reconstruction task is put into optimisation context: we are looking for the implicit function through which the flux of the surface-consistency scaled vector field, that is reconstructed from its projections to the image plane (the observed normal fields $\mathcal{N}_i$) is maximal. The method is based on the minimization of an energy functional similar to the one derived in [CLL07]. It consists of a data term, maximizing the flux of the vector field through the surface, and a regularization term, enforcing minimal surface area. One reason for this particular choice of energy functional is that it naturally extends classic single-view normal integration functionals [Hor90, SCS90] to the multi-view setting, as shown in [CLL07]. However, in addition to the different scheme for the minimization of respective energy functional, our method for computation of the vector field differs from the one employed in [CLL07].

Implicit Function Fitting Approach

The basic idea of the implicit function fitting approach is to fit an implicit function, representing the object, to the data. After the reconstruction of an implicit function, the surface of the object can be recovered by extracting its appropriate level set using e.g. Marching Cubes [LC87] or Dual-Contouring [SW04] and stored as a polygonal mesh. The implicit representation of the object for the purpose of 3D reconstruction has several advantages over explicit representations. It can handle arbitrary topology without the need for parametrization. Furthermore, the implicit function reconstruction based approaches are usually very robust to noise and can deal well with the filling of the holes and produce water-tight surfaces. The cost of this kind of representation are higher memory demands for the digital representation of the object.

For the reconstruction of the surface, we consider a volume $V \subseteq \mathbb{R}^3$ tightly bounding the object. In the volume, the surface is represented implicitly by a binary indicator function, indicating the interior of the object S in the volume V by $\gamma : \mathbf{x} \to \{0,1\}$, $\mathbf{x} \in V$.

Energy Model

The exact energy model we consider is the following:

$$E(\partial S) = \lambda_1 \underbrace{\int_{\partial S} dA}_{E_1} - \lambda_2 \underbrace{\int_{\partial S} \mathbf{n} \cdot (c\mathbf{N})\, dA}_{E_2}, \tag{3.1}$$

where λ_1 and λ_2 are the weights of individual terms, $\mathbf{n}(\mathbf{x})$ denotes the outward unit normal of ∂S at the spatial point $\mathbf{x}$, $c(\mathbf{x})$ is the surface consistency function based

on normal disparity and $\mathbf{N}(\mathbf{x})$ is the vector field, reconstructed from observed normal fields. The first term, E_1, is the regularization term, penalizing high oscillations of the surface which effectively prevents the over-fitting and propagates a smooth, minimal-area surface over the holes and parts where no data has been recorded. Smoother reconstructions are achieved by increasing the value of the weight λ_1. By maximizing the data term E_2 we are looking for the surface ∂S that aligns best to the vector field $\mathbf{N}(\mathbf{x})$ i.e. the surface through flux is maximal. By scaling the vector field with the surface-consistency function $c(\mathbf{x})$, solutions in the region where surface consistency is high are preferred.

Our energy model is actually a specialization of the more general family of minimal surface problems given in the form

$$E(\partial S) = \lambda_1 \underbrace{\|\partial S\|}_{\text{reg. term}} + \lambda_2 \underbrace{\int_S f \, dV}_{\text{data term}}, \tag{3.2}$$

where $f(\mathbf{x})$ represents the cost of assigning $\mathbf{x}$ to the object interior S and $\|\partial S\|$ denotes the area of the surface with respect to some norm [Kol12]. A common way to solve this kind of energy functionals for purposes of 3D reconstruction or volumetric segmentation is to treat it as a binary segmentation problem. Then, the goal is to divide the volume into two distinct regions: the region belonging to the object, S, and the background, $V \setminus S$. Binary segmentation can be posed as a recovery of a binary indicator function $\gamma : \mathbf{x} \mapsto \{0, 1\}$, that assigns every spatial point $\mathbf{x}$ either to interior of the object or to the background. Following [Kol12], equation (3.2) can be restated as a volume integral

$$E(\gamma) = \lambda_1 \int_V \|\nabla \gamma\| \, dV + \lambda_2 \int_V \gamma f \, dV. \tag{3.3}$$

In order to represent the energy model in equation (3.1) as a volume integral, the surface integral of the flux of the vector field is replaced by the volume integral of the divergence of the surface-consistency scaled vector field (since by the divergence theorem the maximization of the flux-surface integral is equivalent to the maximization of the volume integral of the divergence of the vector field). Hence, the regional term $f(\mathbf{x})$ is assigned to be the negative of the divergence $-\nabla \cdot (c\,\mathbf{N})$ of the vector field similar as done in [LB07] in the context of shape fitting to oriented point clouds.

$$E(\gamma) = \lambda_1 \int_V \|\nabla \gamma\| \, dV - \lambda_2 \int_V \gamma \left(\nabla \cdot (c\,\mathbf{N})\right) \, dV. \tag{3.4}$$

Energy Minimization

In the area of computer vision, the development of efficient and reliable methods for minimizing functionals of the form given in equation (3.2) has a long history.

Several numerical techniques have been developed for the purposes of 2D segmentation on the image plane, characterizing regions of interest by an 2D contour such as the active contour model [CV99] and level-sets [OS88]. In the past years, for a specific class of minimal length/area-based energy functionals in a discrete setting, graph-cut based techniques [GPS89, BK03, KB05] turned out to be especially successful. In contrast to the active contour model [CV99, CKS97] and level-sets [OS88], that usually converges towards local minima, graph-cuts can guarantee the computation of an optimal solution to the discretised version of the energy functional. In this case, the segmentation is performed by computing a minimal cut on the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of graph nodes which correspond to image pixels or voxels of the discretised volume and $\mathcal{E}$ is the set of edges. The edges between spatial nodes (also referred to as *n-links*) encode the desired local metric and edges connecting data nodes to two additional terminal nodes (*t-links*) encode regional costs of the energy functional. The segmentation task is then reduced to the partitioning of the data nodes to two disjoint sets, one containing the source node and the second containing the sink node. The cut between two terminal (*source* and *sink*) nodes with lowest cost exactly corresponds to the minimal cut on the graph $\mathcal{G}$. Due to the duality between *min-cut* and *max-flow* problems, efficient, polynomial-time algorithms for segmentation via graph-cuts exist, e.g. [FF62, GT86]. A detailed review of max-flow algorithms for computer vision applications is given in [BK04].

All these methods have been successfully applied in the area of 3D segmentation in the volumetric data for the purpose of 3D reconstruction: active contour [ES04], level-sets [FK98, KBC06, CLL07, LQ05] and graph-cuts [KZ02, VETC07, YAC06]. While the idea of using graph-cuts for 3D reconstruction is very attractive due to its global optimisation guarantees, the shortcomings of this approach become very problematic in this particular field. The main problem of this method is that the regularized solution typically suffers from the grid bias. This metrification problem can be reduced by considering a higher number of nodes for the approximation of the metric, but that even increases the already high memory demands in the case of volumetric data, prohibiting a fine discretisation of the volume. The *touch-expand* algorithm proposed in [LB07] for computation of the max-flow in the volume can reduce memory demands, but it takes advantage of the sparsity of the volume, which is in our case not guaranteed.

In recent developments in the area of multi-view 3D reconstruction [KKB$^+$07, KC08], which are based on the work of [NEC06] in area of image segmentation, analogue convex relaxation methods have been proposed to solve the minimal length/surface based energy functional (3.2). The basic idea of convex relaxation approaches is to relax the restriction of a reconstruction of the binary-valued indicator function $\gamma : \mathbf{x} \mapsto \{0, 1\}$ to the real domain: $\gamma : \mathbf{x} \mapsto [0, 1]$. With the relaxation of the (non-convex) domain of binary functions to the domain of real func-

tions, functional (3.3) can be globally optimised by means of convex optimisation, since it is a convex energy functional, defined on a convex domain (for proof, see [KC08]). Due to the *thresholding theorem* the optimal solution to the original binary problem can be obtained by simply thresholding the solution γ^* of the relaxed problem for any threshold $T \in (0,1)$ (for details, see [NEC06, KKB$^+$07]). The segmentation problem can thus be reduced to solving a constrained convex optimisation problem of minimizing functional 3.3 w.r.t. the relaxed indicator function $\gamma : \mathbf{x} \mapsto [0,1]$. In [KSK$^+$08], the authors compare discrete optimisation methods (graph-cuts) and convex-relaxation based methods for multi-view 3D reconstruction applications. They conclude, that the use of convex-relaxation based methods presents great benefits over the discrete, classic graph-cut approaches. In a nutshell, these approaches have much lower memory demands and do not suffer from metrification artefacts, but usually at the cost of higher computation time.

In order to be able to use fine discretisations of the volume and take advantage of fine-precision information provided by the normal information, a convex relaxation based method is used for the minimization of proposed energy functional (3.4). The optimisation is done using the continuous max-flow based method, proposed in [YBT10], since it is particularly simple to implement. This method can be considered as the dual-model of the convex relaxation method proposed in [NEC06], which can also be regarded as a continuous version of the min-cut problem. Using the notation of [YBT10], we set the continuous max-flow capacity functions as:

$$C(\mathbf{x}) = \lambda_1,\ C_s(\mathbf{x}) = \lambda_2 \max(0, \nabla \cdot (c\,\mathbf{N})(\mathbf{x})),\ C_t(\mathbf{x}) = \lambda_2 \max(0, -\nabla \cdot (c\,\mathbf{N})(\mathbf{x})),$$

and solve

$$\min_{\gamma \in [0,1]} \int_V (1-\gamma) C_s + \gamma\, C_t + C\, \|\nabla \gamma\|\ \mathrm{d}V \tag{3.5}$$

using the multiplier-based max-flow algorithm proposed in [YBT10]. The desired binary segmentation is obtained by thresholding the minimizer γ^* of (3.5).

3.2.2 Vector Field Computation and Surface Consistency Measure

In this section, we explain how the vector field $\mathbf{N}(\mathbf{x})$ and the surface-consistency function $c(\mathbf{x})$ are computed, which are used for the data term in the optimisation process. Naturally, this requires determining for every spatial point $\mathbf{x} \in V$ how likely it belongs to the surface, and what the most likely normal at that point is.

We treat the observed normal fields $\mathcal{N}_i$, $i = 1 \ldots \kappa_c$ as discrete samples of a continuous vector field, which are projected to the image planes of the cameras C_i and one of the main tasks is to recover the most probable vector field that lead to

the observations. Let $\mathrm{D}_{\mathbf{x}} = \{\tilde{\mathbf{n}}_{1,\mathbf{x}} \dots \tilde{\mathbf{n}}_{\kappa_c,\mathbf{x}}\}$ denote a set of normals back-projected from the input normal fields $\mathcal{N}_i$ to the spatial point $\mathbf{x}$. The vector field $\mathbf{N}(\mathbf{x})$ at $\mathbf{x}$ is defined as the most probable normal from the set of back-projected normal samples $\mathrm{D}_{\mathbf{x}}$.

Before we explain our method in detail, we would like to establish some observations of the nature of the data we are dealing with. Naturally, at the points belonging to the surface, normal estimates from different views should exhibit a small angular error w.r.t. the true surface normal. In reality, when observing these sets of normals being back-projected to the surface points, a perfect matching is never achieved due to noise, outliers, systematic errors and discretisations of the volume V. This concept is illustrated in Figure 3.2. Even in case of perfect data,

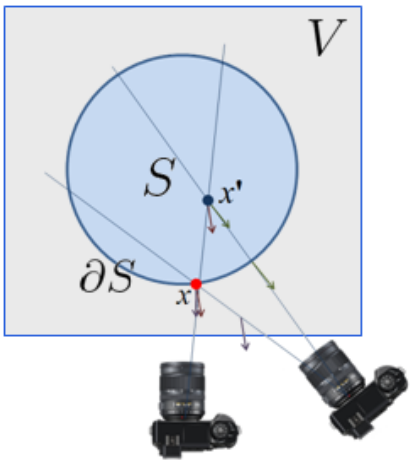

Figure 3.2: The two normal samples back-projected to $\mathbf{x}$ exhibit a small angular error making $\mathbf{x}$ a good candidate for being a surface point. On the other hand, normals back-projected to $\mathbf{x}'$ will exhibit a large angular error, since $\mathbf{x}'$ does not belong to the surface.

outliers due to back-projected normals from the occluded views will be present. However, when observing the set $\mathrm{D}_{\mathbf{x}}$ for a spatial point $\mathbf{x}$ in the close proximity of the surface, it is clear that the back-projections always form a dense cluster, oriented in the direction of the true surface normal $\mathbf{n}(\mathbf{x}) \in \mathbb{S}^2$, see Figure 3.3. Due to the constantly present noise and outliers, the datasets obtained from back-projected normals $\mathrm{D}_{\mathbf{x}}$ are complex-structured and highly-cluttered, see Figure 3.3. For this reason, the detection of the true normal direction $\mathbf{n}(\mathbf{x})$ at surface points is non-trivial. The natural choice for the surface consistency measure would be the disparity between the normals in $\mathrm{D}_{\mathbf{x}}$, but simply computing the angular error between the normals in the set would not lead to a faithful normal consistency measure due to constantly present outliers. The effects of outliers, originating

from occluded views, could be compensated by the computation of a visibility approximation, which is often done by using silhouette information. Then one possible way for computation of a surface consistency measure would be to fit a probabilistic model to the observed data.

Based on similar observations, the author of [Dai09] explored an approach, where he developed a statistical model for the likelihood of the surface passing through a 3D point $\mathbf{x}$ based on back-projected normals. The utilized model assumes that for the visible back-projected samples the normals are distributed according to a Gaussian around the true surface normal $\mathbf{n}(\mathbf{x})$ and the normal estimates from the occluded views are treated as outliers and modelled by a uniform probability distribution function. The visibility is treated as a hidden variable. For the maximization of the likelihood, the Generalized Expectation-Maximization (GEM) algorithm is used, which requires an initial estimate of the geometry as a visibility approximation. As pointed out in [HZ03], the assumption of a Gaussian for modelling the error is not justified at all, but it might be a reasonable approximation, since it is the tendency of complex processes to converge towards a Gaussian. An issue of fitting a parametric model to the observed variables arises due to the fact, that occluded views are not the only source of outliers, at least not in real-world scenarios. In addition, as the underlying pdf is multi-modal, the GEM approach, based on estimation of the visibility, might not be able to find the maximal mode. As a result, while their approach produces excellent results on the synthetic normals, it does not perform very well on the real-world normal datasets computed via photometric stereo.

Looking at the visualization of the estimated probability density function from the observed discrete samples using an Gaussian kernel (Figure 3.4), it becomes obvious, that the underlying probability density function is in fact multi-modal and far more complicated than a mixture of Gaussian and uniform distribution. The choice of a poor model usually leads to a bad performance and modelling all possible sources of errors and outliers is infeasible, making non-parametric methods favourable. Based on these observations, we propose to treat all outliers equally and perform instead a robust feature-space analysis on a set of back-projected normals $\mathsf{D}_{\mathbf{x}}$ at each spatial point. The feature space can be regarded as the probability density function of the parameters, describing the observed data and the significant features correspond to the modes of this unknown underlying probability density function [CM02]. Consequently the identification of the highest mode and the corresponding density of the pdf at the center of this mode would provide both, parameters for the normal estimate and a surface consistency measure.

The vector field $\mathbf{N}(\mathbf{x})$ is then computed by mapping all normal samples in $\mathsf{D}_{\mathbf{x}}$ to the feature-space and analysing the probability density function $\rho(\varphi)$ of the parameters of that feature space. The mapping $\Phi : \mathbb{S}^2 \mapsto \mathsf{F}$ maps normals from the surface of the sphere to the feature space forming a set $\mathsf{F}_{\mathbf{x}} = \{\varphi_{1,\mathbf{x}} \dots \varphi_{\kappa_c,\mathbf{x}}\}$.

The normal direction which corresponds to the highest mode of the probability density function of the parameters is assigned to be a normal at $\mathbf{x}$ and the value of the scalar-valued surface consistency function $c : \mathbb{R}^3 \mapsto \mathbb{R}_+$ is the density of $\rho(\varphi)$ at that point. The vector field $\mathbf{N}(\mathbf{x})$ is defined as

$$\mathbf{N}(\mathbf{x}) = \Phi^{-1}(\varphi^*), \tag{3.6}$$

where φ^* is the maximal mode of the pdf $\rho(\varphi)$ according to

$$\varphi^* = \arg\max_{\varphi} \ \rho(\varphi|F_{\mathbf{x}}), \tag{3.7}$$

and the corresponding scalar-valued surface consistency function is given by

$$c(\mathbf{x}) = \rho(\varphi^*). \tag{3.8}$$

For finding the modes and density estimates we implemented two approaches for non-parametric density estimation, a histogram method and a mean-shift clustering.

Histogram Method

The idea of the histogram method is to approximate the unknown pdf with the discrete bins. This idea is illustrated in Figure 3.5. For that, the feature space is first discretised into bins of width Δ_w and height Δ_h. Then, for each bin the number p of observations of $\rho(\varphi)$ falling into a specific bin is counted [Bis06]. In this case, we parametrise the feature space via spherical coordinates, the azimuthal angle θ and the elevation ϕ angle. The number of actual bins depends on the chosen angular resolution Γ, resulting in a 2D accumulator parametrized by $\alpha = 1 \ldots \frac{360}{\Gamma}$ and $\beta = 1 \ldots \frac{180}{\Gamma}$. The normalized probability density function (constant over the bin) is then given by

$$\rho(\theta, \phi) = \frac{p_{\alpha\beta}}{\kappa_c \cdot \Delta_{\alpha\beta}}, \tag{3.9}$$

where $\Delta_{\alpha\beta}$ is the area of the bin and $p_{\alpha\beta}$ is the number of observations, falling to the bin. The parameter κ_c denotes the total number of views (normals). The maximal mode of the discretised pdf $\rho(\theta, \phi)$ can be found by a simple exhaustive search for maxima over the accumulator. Obviously, this method depends on one parameter, the size of the bin, determined by the angular resolution Γ. This parameter has to be chosen carefully, and it depends on both the quality of the normals and the discretisation of the volume V. In practice, we used an angular resolution between $5^\circ - 10^\circ$ for the feature space, on synthetic datasets even a

precision of 1° is possible for finer discretisations. The advantages of this method are a high robustness towards the outliers and a low computational complexity.

Even though this simple approach can be successfully used for the computation of the vector field and can produce reasonable reconstruction results, it suffers from the typical drawbacks of the histogram method, the discretisation artefacts. After finding the best parameters in the feature space and computing back the normal value from the angular parameters, the normal orientation can only be obtained up to an angular resolution of the feature space, due to the constant value of the underlying pdf in the bins.

A second problem of the histogram method is the curse of dimensionality, since the number of bins is raising exponentially with the dimension. For that reason, we parametrized the feature space by spherical coordinates when estimating density with this method. However, this parametrization does not come without cost: the transformation from Cartesian to spherical coordinate system introduces singular points at the poles, where $\phi = 0$ or $\frac{\pi}{2}$, and normals, pointing in that direction (or near proximity) will not form a dense cluster.

Kernel Density Estimation and Mean-Shift Clustering

The idea behind the kernel density estimation methods is to overcome the limitations of the histogram method by estimating a smooth pdf from the discrete (observed) samples by a superposition of smooth kernels K at discrete samples, see Figure 3.5. More precisely, it is assumed, we have N observations $\mathbf{y}_i \in \mathbb{R}^d$, $i = 1, \ldots, N$, drawn from a continuous pdf $\rho(\mathbf{y})$. Then, the probability density estimate using the kernel K of the observed data at the point $\mathbf{y} \in \mathbb{R}^d$ can be estimated by:

$$\rho(\mathbf{y}) = \frac{1}{Nh^d} \sum_{i=1}^{N} K\left(\frac{\mathbf{y} - \mathbf{y}_i}{h}\right), \tag{3.10}$$

where $K(\mathbf{y})$ is a *kernel function* and h^d is a d-dim. hyper cube with edge length h [Bis06]. Additionally, we refer to h as the *window parameter* (also refereed to as *smoothing parameter*, which plays a similar role as the bin size in the histogram approach). Intuitively, the kernel density at a certain spatial point is the sum of the values of the kernel functions, that are positioned in the center of the discrete samples. To ensure that the resulting pdf is valid, it is required the that kernel function is symmetric and it integrates to one. Amongst the most common kernel functions are the Gaussian kernel, the Epanechnikov kernel and the uniform kernel, visualized in Figure 3.6.

The mean-shift clustering algorithm [FH06, Che95] is an algorithm for finding the modes of the probability density function of the feature space without actually estimating the density. The whole mean-shift clustering procedure can be seen as

performing an adaptive step size gradient ascent on the pdf, estimated from the discrete samples using a kernel K [FH06]. Considering the family of radially-symmetric kernels defined as:

$$K(\mathbf{y}) = c_{k,d}\, k\left(\|\mathbf{y}\|^2\right), \tag{3.11}$$

where $c_{k,d}$ is a normalization constant which ensures $K(\mathbf{y})$ integrates to one and $k(\mathbf{y})$ is a profile of the kernel [Che95, CM02] and by denoting the derivative of the profile with $g(\mathbf{y}) = -k(\mathbf{y})$, the gradient $\nabla\rho$ (please refer to [CM02] for derivation) is:

$$\nabla\rho(\mathbf{y}) = \frac{2c_{k,d}}{Nh^{d+2}} \underbrace{\left[\sum_{i=1}^{N} g\left(\left\|\frac{\mathbf{y}-\mathbf{y}_i}{h}\right\|^2\right)\right]}_{(1)} \underbrace{\left[\frac{\sum_{i=1}^{N} \mathbf{y}_i g\left(\left\|\frac{\mathbf{y}-\mathbf{y}_i}{h}\right\|^2\right)}{\sum_{i=1}^{N} g\left(\left\|\frac{\mathbf{y}-\mathbf{y}_i}{h}\right\|^2\right)} - \mathbf{y}\right]}_{(2)}. \tag{3.12}$$

The first term is a scalar proportional to the density estimate at $\mathbf{y}$ with the shadow kernel $G(\mathbf{y}) = c_{g,d}\, g\left(\|\mathbf{y}\|^2\right)$ and the second term is a mean-shift vector, which points towards the maximum increase of the density. The nearest mode to the discrete sample $\mathbf{y}_i$ can thus be computed by centring the window at $\mathbf{y}_i$, computing the mean-shift vector and successively translating the window in the direction of the mean-shift vector (pointing in the direction of the gradient of the kernel density estimate with kernel K), as long as its magnitude does not converge to zero. In order to locate all the modes, the mean-shift procedure is initiated from all discrete samples $\mathbf{y}_i$.

In the mean-shift case, we have chosen Cartesian coordinates for the parametrization of the feature space (hence, the mapping function Φ is an identity). The same parametrization, as used for histogram approach would be possible, but as already discussed, the transformation from the Cartesian to the spherical coordinate system introduces singular points. In practice, for a spatial point $\mathbf{x}$, the mean-shift procedure is performed directly on $D_{\mathbf{x}}$. After running the mean-shift procedure, all modes of ρ are identified. At every mode, the density of the pdf can be estimated based on the kernel K, and the parameters corresponding to the highest mode correspond to the normal assigned to the spatial point $\mathbf{x}$ and the consistency at that point is directly the kernel density estimate at that mode.

We performed experiments using Gaussian and Epanechnikov kernels. Due to the faster convergence of the mean-shift procedure using the Epanechnikov kernel, the execution time of the algorithm in this case was notably faster, but running the mean-shift algorithm using a Gaussian kernel produced higher quality (smoother) results.

The draw-back of computing of the vector field using the mean-shift algorithm is a higher computational complexity and it presents the bottle-neck of our method. The mean-shift procedure time complexity is quadratic in the number of samples, which in our case approx. represents the number of cameras, and it must be performed for each discrete sample of the volume. Consequently, increasing the number of views fundamentally slows down the running time. However, in cases where lower computational time would be desired, either Epanechnikov kernel or histogram method can be used.

3.3 Implementation

When using a volumetric representation of an object, a memory efficient discretisation of the volume and a careful choice of numerical methods for the optimisation procedure is essential. Use of a regular (voxel) grid for the discretisation is elegant and simple, but, in general, it does not allow fine-detailed reconstructions, since the number of voxels in the grid is raising cubically with the cross-sectional resolution, effectively preventing fine discretisations due to practical limitations of current computer systems. On the other hand, the use of normal information in theory allows for fine-precision reconstructions. In order to be able to achieve high-quality reconstructions, we consider a discretisation of the volume, which is successively adapting to the surface consistency measure [WRO+12].

In particular, we use an octree data structure for the dicretisation. Naturally, the volume should be fine discretised in the area where it is likely that the surface is passing through and roughly elsewhere. To achieve that, the octree is coarsely subdivided first up to pre-defined level by the employed initial subdivision procedure. Second, the octree continues adapting to the surface by performing successive reconstructions, each time refining the octree only in the narrow band near the last reconstruction [WRO+12].

After the computation of the surface-consistency scaled vector field in the subdivided octree volume, the divergence in each octree cell is computed. The next step is the computation of the the binary indicator function using the continuous max-flow solver [YBT10]. To overcome discretisation limitations an additional smoothing optimisation step based on normal information is performed in the narrow band near the binary solution, similar to the approach used in [CT11, WRO+12].

3.3.1 Octree-based Discretisation and Initial Subdivision

The bounding volume V is discretised using an octree $\mathcal{O}$. For every octree node $o \in \mathcal{O}$, the following information are stored: a value indicating a labelling of the

node, a value representing the value of the divergence in that node and additionally, corner data for eight corners, where the neighbouring nodes share the corner data information. The corners of the nodes store the information about the surface-consistency scaled vector field $c\mathbf{N}(\mathbf{x})$ i.e. each cell corner $q \in \mathcal{O}$ stores a consistency-scaled normal.

The initial, coarse subdivision of the volume is implemented as follows. Each node $o \in \mathcal{O}$ is tested whether the surface could pass through it or not, based on back-projections of the normals from the input normal fields. In case the number of matchings exceeds a predefined threshold, the node is refined. In order to test whether the surface might be passing through o, footprints of the node are computed and compared with respect to each camera.

The node footprint, computed with respect to camera C_i, is a simple histogram, $\mathbf{H}_i$, computed as follows. The set of normals, covering the area of $\mathcal{N}_i$, that intersects with the back-projection of the node o to the image planes of the cameras, is transformed from Euclidean coordinates to spherical coordinates by computing azimuthal and elevation angle. We consider an angular discretisation of 10° and compute binary-valued histograms $\mathbf{H}_i$, $i = 1 \dots \kappa_c$ of size 18×36 for each node-projection to the camera image plane. For the node in consideration, an union of binary histograms $\mathbf{U}_o = \sum_{i=1}^{\kappa_c} \mathbf{H}_i$ is computed and thresholded. When any of the bins of thresholded $\mathbf{U}_o$ is non-zero, the surface might be still passing through the node, since many histograms (cameras) are suggesting that they have observed a normal, corresponding to non-zero entry of $\mathbf{U}_o$. Such node will get recursively refined further.

Note, that main idea behind the initial subdivision strategy is similar to the core idea of the reconstruction approach: in the area, where the surface is passing, normal samples should agree, and by this approach, we are just checking if number of matching normals within the projected areas are higher than a user-defined threshold. This threshold should depend on the number of cameras in the scene and it should not be set too high, in order to make sure that all areas, where the surface might be passing, are sufficiently refined, even at the cost of refining more nodes than necessary. In practice, we used an initial subdivision strategy up to the octree level 7. The slice of the initial subdivision of the octree for the synthetic sphere example is visualized in Figure 3.7.

3.3.2 Vector Field and Divergence Computation

Per-Corner Vector Field and Surface Consistency Computation

The value of the surface-consistency scaled vector field $c\mathbf{N}(\mathbf{x})$ is computed for each corner q of each node o in the octree as follows. The corner q with the world-coordinates $\mathbf{x}$, which can be considered as a discrete sample of the volume V at

which the value of $c\mathbf{N}$ is evaluated, is first back-projected to each camera C_i. The set of normal samples $D_{\mathbf{x}} = \{\tilde{\mathbf{n}}_{1,\mathbf{x}} \dots \tilde{\mathbf{n}}_{\kappa_c,\mathbf{x}}\}$ being back-projected to the spatial point $\mathbf{x}$, which corresponds to the corner, is then mapped to the feature space $F_{\mathbf{x}}$, just as discussed in Sec. 3.2.2. Then, one of the probability density estimation methods discussed in 3.2.2 is applied on the set of discrete points $F_{\mathbf{x}} = \{\varphi_{1,\mathbf{x}} \dots \varphi_{\kappa_c,\mathbf{x}}\}$ in feature space in order to find the parameters φ^*, for which the probability density function of the parameters is highest. The normal corresponding to φ^* is then scaled by the corresponding density estimate at that point and assigned to the corner q. A slice of computed surface consistency function is visualized in Figure 3.7.

Per-Cell Flux and Divergence Computation

After the computation of the vector field $c\mathbf{N}$, the divergence $\nabla \cdot c\mathbf{N}$ is computed for each node in the octree, see Figure 3.7. Due to the divergence theorem, the volume integral of the divergence in the node is equivalent to the flux through all six faces of the cube, representing the octree node. Hence, the divergence for the node o is computed by summing the flux through the faces, divided by the edge length of the node:

$$\nabla \cdot c\mathbf{N}(o) = \frac{\sum_{i=1}^{6} \text{flux}(f_i)}{o.\text{length}}, \tag{3.13}$$

where the flux through the face f_i face is computed as a dot product between the face normal and the sum of the vector field values at the corners of the face.

3.3.3 Iterative Surface Reconstruction and Post Processing

In order to maintain memory and computational efficiency an unnecessary subdivision of the octree at higher octree depths should be avoided. In order to do so, after a coarse initial subdivision of the octree, a coarse reconstruction of the surface is computed. For the binary labelling of the nodes, the energy functional (3.5) is minimized using the multiplier-based max-flow algorithm proposed in [YBT10, WRO+12] and the result is thresholded for each cell to obtain a binary labelling, see Figure 3.7.

After the labelling, the nodes of the octree that lie in the close proximity of the cutting-plane between the background and the segmented region are computed and refined to the next octree level. The whole reconstruction procedure is then repeated using the new, finer subdivision of the volume. In practice, we used an initial subdivision of the volume up to the octree level 7 and performed two additional labelling-refinement steps.

After recovering the binary indicator function $\gamma(\mathbf{x})$ at the desired octree resolution, the resulting reconstruction is not smooth. In order to obtain a smooth

polygonal mesh, an additional smoothing step is performed before extracting the appropriate iso-surface. To obtain a smooth surface, a smooth signed distance function $s(\mathbf{x})$ is computed in near proximity of the binary result (inspired by [CT11, WRO+12]). The smooth signed distance function is hard-constrained to lie within a band of one octree cell of the binary result and is computed by minimizing an energy functional, which enforces $s(\mathbf{x})$ to adapt to the vector field $N(\mathbf{x})$ and penalizes high-curvatures of the surface in order to avoid over-fitting. As a final step, the resulting implicit smooth signed distance function $s(\mathbf{x})$ is converted into a polygonal mesh using octree-based iso-surface extraction method, proposed in [KKDH07].

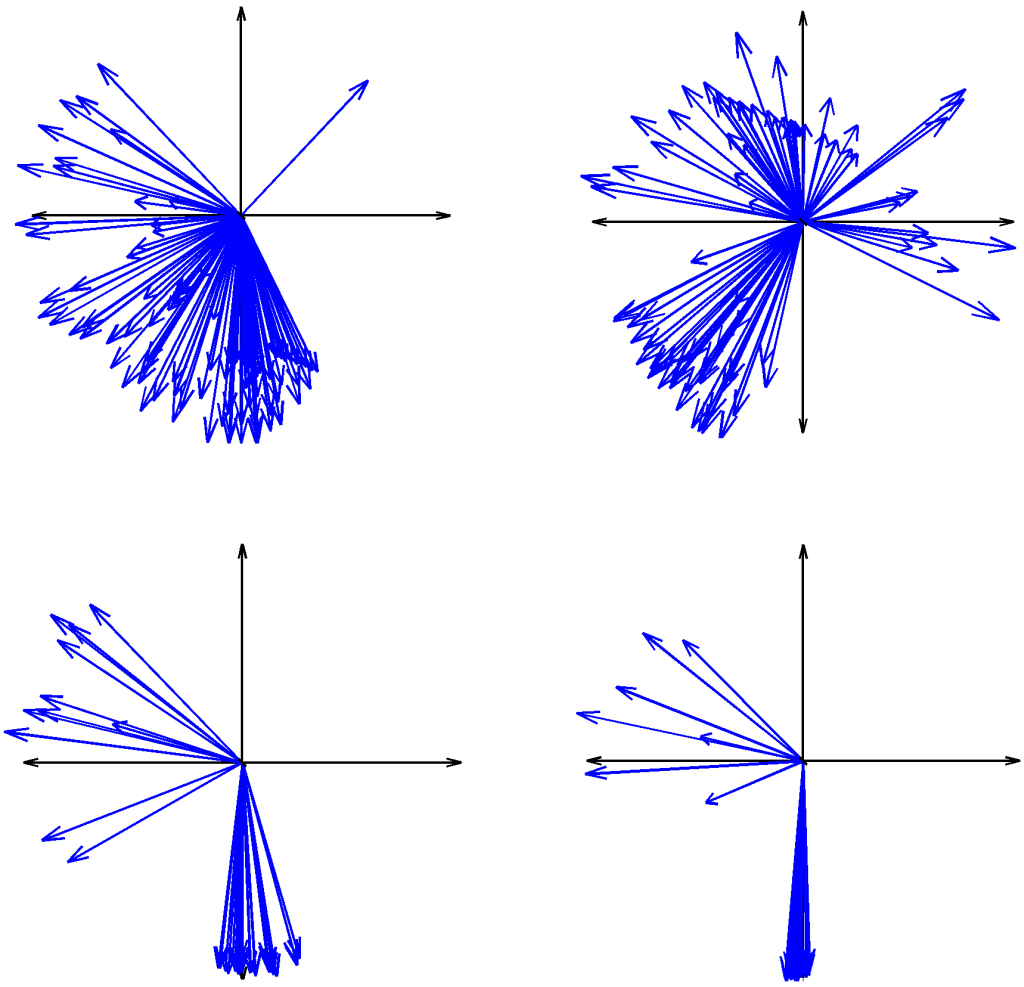

Figure 3.3: Visualization of back-projected normals to four spatial points with varying distances from the surface. The distance is decreasing from top-left visualization to lower-right.

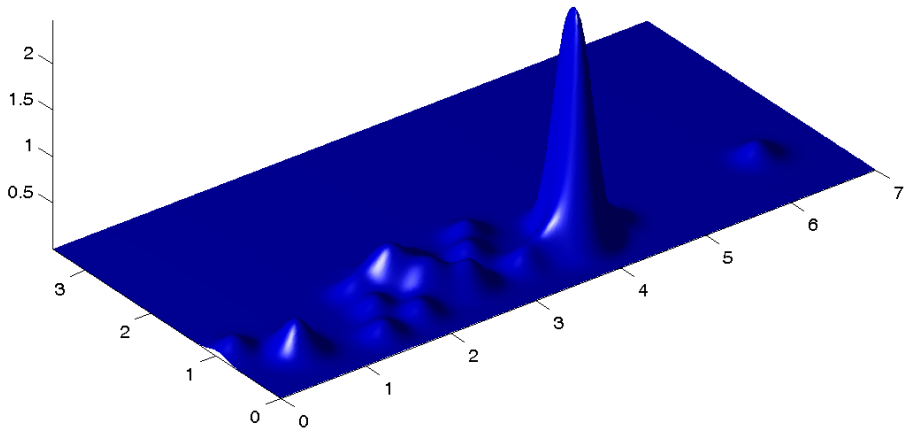

Figure 3.4: Visualization of the probability density function of the observed data $D_{\mathbf{x}}$, where $\mathbf{x}$ is a spatial point at the near proximity of the surface. The probability density function was reconstructed from the discrete samples using a Gaussian kernel. The parameter space is parametrized via spherical coordinates, and the two values for which the pdf is highest correspond to the normal direction at $\mathbf{x}$. The density of the pdf for these parameters gives a measure of surface consistency.

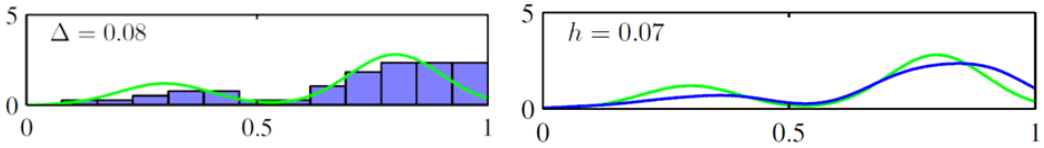

Figure 3.5: Illustration of two methods for estimation of unknown pdf (green) from the discrete samples: histogram method *(left)* and kernel density estimation method *(right)* (image source: [Bis06]).

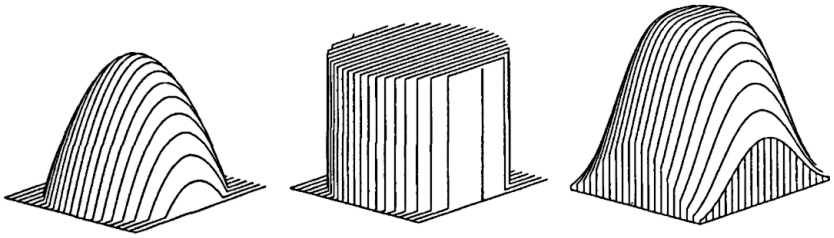

Figure 3.6: Visualizations of Gaussian, uniform and Epanechnikov kernel (image source: [Che95]).

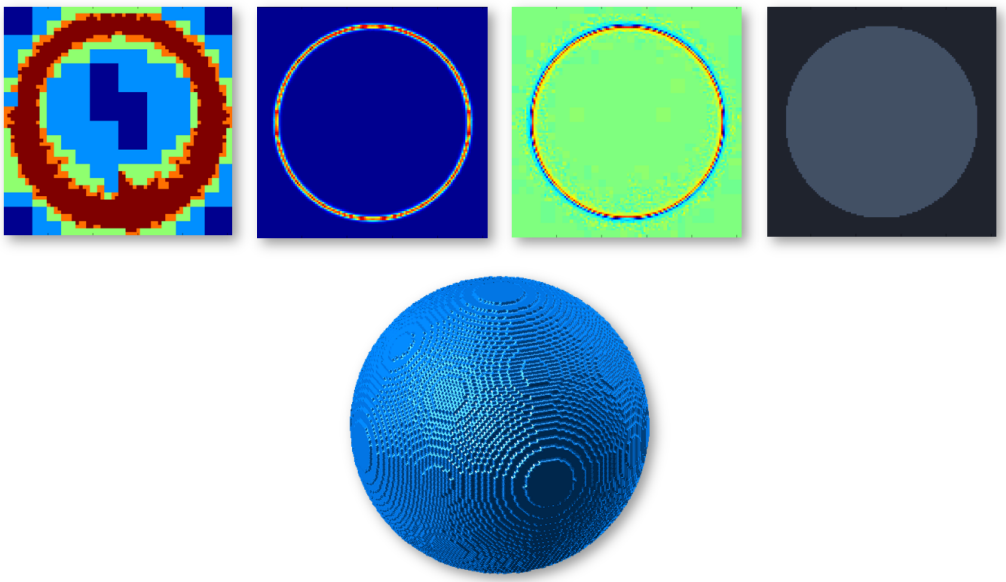

Figure 3.7: Slices (from left to right) of the initial subdivision of the volume, surface consistency function, divergence and corresponding binary labelling for the synthetic sphere dataset. At the bottom, mesh obtained after first iteration of the segmentation is visualized.

CHAPTER 4

MULTI-VIEW SHAPE FROM SPECULARITY

In this chapter, the ideas behind classic shape-from-specularity approaches are extended to the multi-view setting in order to capture the full geometry of objects with highly-specular surfaces (e.g. polished metal, mirror, etc.). The proposed method does not rely on any prior knowledge about the geometry, such as the assumption of rather flat objects [CGS06, FCMB09], but it is assumed that the BRDF of the surface of the unknown object has a strong specular component. Assuming ∂S is a specular surface of a solid S, a large portion of the incoming

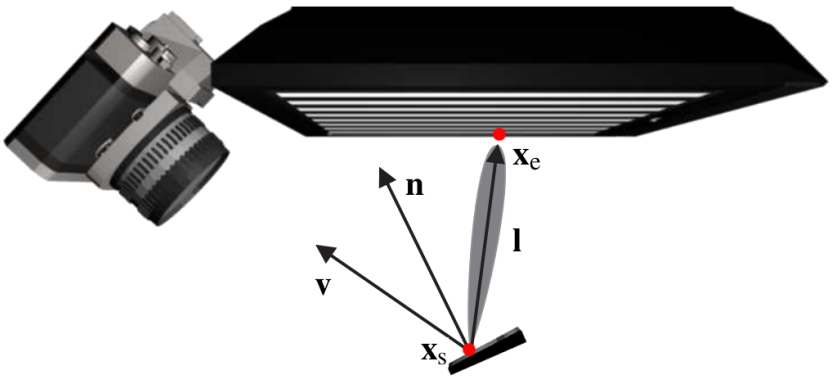

Figure 4.1: The 3D surface point $\mathbf{x}_s$, observed by the camera, is illuminated by the scene point $\mathbf{x}_e$. The normal $\mathbf{n}(\mathbf{x}_s)$ is a bisector of unit vectors $\mathbf{v}$ and $\mathbf{l}$ (image source: [FCMB09]).

light to the surface point $\mathbf{x}_s \in \partial S$ is reflected toward the direction of perfect reflection $\mathbf{r}$. When the view direction $\mathbf{v}$ coincides with $\mathbf{r}$, the image sensor observes at $\mathbf{x}_s$ a reflection of scene point $\mathbf{x}_e$. In this case, the vectors $\mathbf{v}$ and $\mathbf{l}$ are coplanar and the normal at $\mathbf{x}_s$ is the bisector of them, see Figure 4.1. Using knowledge of this image formation model for specular surfaces, normal fields can be computed. By capturing images of the object with a calibrated camera, we have at each pixel information about the view direction $\mathbf{v}$ and the light direction vector $\mathbf{l}$ can be com-

puted by using a calibrated and structured environment. Using this information and assuming far-field illumination, it is possible to compute corresponding normal estimates of surface points by computing the bisector $\mathbf{n} = \frac{\mathbf{v}+\mathbf{l}}{\|\mathbf{v}+\mathbf{l}\|}$ between the vectors $\mathbf{v}$ and $\mathbf{l}$. This approach has already been successfully used in context of the reconstruction of specular object [CGS06, NWR08, FCMB09, BHB11, BS03], addressing simple, nearly flat objects.

The main idea of our approach is the following. From multiple views of the object, we compute **light maps** [BW10], which relate projections of surface points $\mathbf{x}_s$ into the cameras with the scene points which caused the illumination at $\mathbf{x}_s$. Since in our setup, the light sources used for the computation of the light maps are close to the object and distant lighting assumption is violated, normals can not be computed directly. Instead, based on the law of reflection, normal hypotheses are computed at each voxel and the multiple views provide necessary constraints for resolving the normal ambiguities along the viewing rays. Hypothesised normals are integrated using the algorithm we proposed in Chp. 3.

In continuation of this chapter, the nature of the problem is discussed first and a notation is established. Then, the measurement process for recording the reflectance data of the object from multiple views, necessary for relating the surface points of the unknown object with the scene points that illuminated them, is explained. Finally, generation of the normal hypotheses and their integration using the multi-view normal field integration algorithm is presented.

4.1 Problem Statement

The problem formulation for reconstructing specular objects by means of multi-view normal field integration is very similar to the problem formulation described in Chp. 3. We assume that the object of interest exhibits a specular reflectance behaviour and that it is surrounded by κ_c calibrated cameras C_i, $i = 1 \dots \kappa_c$, which are oriented towards the object, see Figure 3.1, and κ_d display screens D_j, $j = 1 \dots \kappa_d$. Each camera comes in a pair with κ_d light maps $\mathcal{L}_j$, $j = 1 \dots \kappa_d$. We assume a perspective projection model, and for each camera, the perspective projection matrix $\mathbf{P}_i$, formed by the intrinsic camera parameters and the extrinsic camera parameters $[\mathbf{R}_i \,|\, \mathbf{t}_i]$.

Light-maps $\mathcal{L}_{i,j} : \mathbb{S}^2 \mapsto \mathbb{R}^3$ [BW10] assign to each view-direction (i.e. pixels on the image plane of C_i) a 3D scene point $\mathbf{x}_e \in \mathbb{R}^3$, which illuminated the surface point $\mathbf{x}_s$, which has been projected to the image plane of the camera. It is important to note, that light maps will in general provide only partial information about the illumination for the area of the surface visible to the cameras. Furthermore, light maps will in general contain noise and outliers due to inter-reflections. It can also easily occur, that light maps will contain large areas, falsely assumed to be part

of the surface. In addition, in general it will not be possible to relate image points to exact scene points, illuminating the surface, but rather with a larger area. The ambiguity in this source of illumination area will vary from pixel to pixel.

The ultimate goal here is, given the non-perfect light maps $\mathcal{L}_{i,j}$ and the corresponding camera parameters $\mathbf{P}_i$, to recover the full surface ∂S of a specular object S. In order to do so, data necessary for the computation of these light maps has to be captured and the light maps must be computed as robustly as possible. To capture the data, it is necessary to build a setup, that enables performing such measurements and perform all required calibration.

4.2 Approach

In order to reconstruct the surface ∂S of a specular object S, two main steps have to be taken. **First**, it is necessary to compute the light maps $\mathcal{L}_{i,j}$ for each camera-screen pair. Before the light maps can be computed, the necessary measurements of the specular object under the calibrated environment need to be taken. For that, we extended the setup, described in Chp. 5 with display screens, that display structured patterns. From the series of those patterns, which are illuminating the surface points, it is possible to relate each projected surface point to the small portion of the display screen, which illuminated that point. To relate the screen portion acting as a source of illumination to the 3D scene position, the display screens need to be geometrically calibrated. **Second**, it is necessary to hypothesise normals of the specular surface, based on the input light maps, and to compute the vector field from these hypotheses. Then this vector field must be integrated using the algorithm explained in Chp. 3 to recover the surface.

4.2.1 The Light-Map Acquisition

There are many possible ways to capture the data, necessary for the computation of light maps. For example, [SWN88] used a LED-grid, successfully switching LEDs on/off while capturing the images. The authors of [CGS06] used a hand-waved light source and four spheres placed around the object. The light source position can be computed for every captured frame from the highlights on the spheres. Specular reflections on the object are detected by simple thresholding. The drawback of those approaches is that they require capturing N images for N single light source positions.

An alternative is to use a coded environment. This way, for a captured image, several sources of light illuminate the object in parallel. In this case, the relation between these sources in the scene and their reflections towards the image sensor must be established. With a single-shot based encoding (e.g. De Brujin sequences

[SPB04], M-Arrays [MOC+98]) it is necessary to capture only one image per view. For example, [BS03] used such a printed color-coded target. In general, for the static setups, a favourable option is the use of sequential (temporal) codes, displayed by the computer screen. In case of binary or Gray codes, capturing only $\lceil \log_2(\text{width}) \rceil + \lceil \log_2(\text{height}) \rceil$ images for a display screen of resolution width $\times$ height is required [LT09]. In the context of shape-from-specularity methods, this type of encoding was used in [FCMB09, NWR08, BHB11]. Furthermore, using display screens, a high resolution can be achieved and the light source positions can be decoded robustly. This approach has widely been used in structured light systems (discussed in Chp. 2) for establishing correspondences between surface points across the images.

For the acquisition of light maps, we decided to use for the encoding Gray codes due to the low Hamming distance (adjacent codes differ for only one bit) and, consequently, high robustness towards falsely decoded labels. A prominent alternative would be the use of phase-shifting. Having a set of κ_d display screens D_j, $j = 1 \dots \kappa_d$, each camera C_i captures κ_p pattern images for each screen, where one image is taken per pattern image displayed on the screen. Such an encoding allows from a sequence of observed intensities for each pixel to identify a region on a display screen, which illuminated the surface point projected to that pixel.

The display screens must be placed close to the object in order to illuminate a large area of the object. In our setup they are placed in a way, that the area, illuminated by the screens, is at least partially visible to the cameras. Furthermore, one display screen per camera is in practice insufficient for a full illumination. We observed, that in practice, it is necessary to relate at least two screens with a view, one at a side and one below the object. Additional screens might be beneficial for recording the data in the deep concave areas.

Encoding

The encoding procedure goes as follows. For each screen D_j, a series of vertical and horizontal Gray code patterns is displayed and illuminating the object. Each camera C_i then captures an image for each displayed pattern, see Figures 4.2, 4.3 (left side). In addition, in order to make the decoding process robust, for each pattern its complement is displayed as well as two additional images, one entirely white and second entirely black. Hence, the total number κ_p of displayed patterns (and per camera-display pair captured images) is $2\left\lceil \log_2(D_j.\text{width}) \right\rceil + 2\left\lceil \log_2(D_j.\text{height}) \right\rceil + 2$.

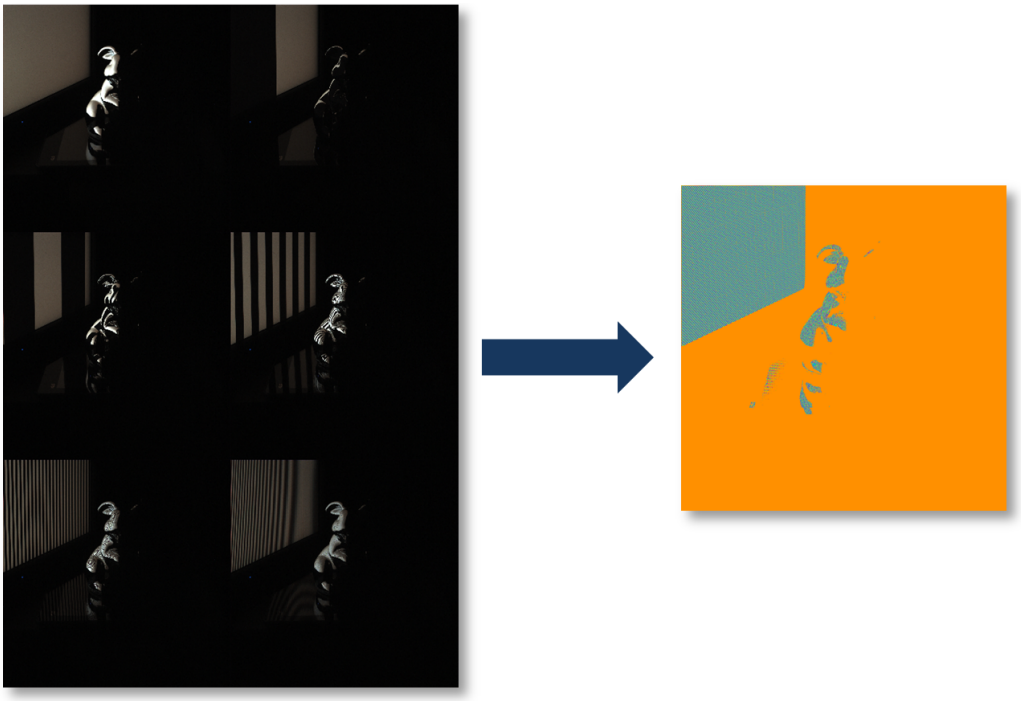

Figure 4.2: A set of six images, where the mirror bunny is illuminated with a series of patterns, and corresponding decoded labels.

Decoding

The task of the decoding procedure is to compute a light map $\mathcal{L}_{i,j}$ from a series of images captured by the camera C_i using screen D_j as illuminant. For each camera pixel $\mathbf{u}$, a codeword of $\frac{\kappa_p}{2}+1$ bits, which assigns an unique label to $\mathbf{u}$, has to be recovered from the captured images. This is done robustly by comparing pairs of images $(\mathcal{I}_{l,P},\mathcal{I}_{l,I})$, $l = 1 \dots \frac{\kappa_p}{2}+1$, taken while illuminating the object with the primary patterns and their inverses. For the image pair $(\mathcal{I}_{l,P},\mathcal{I}_{l,I})$ and the pixel $\mathbf{u}$ it must be determined, whether the white or the black portion of the pattern image illuminated the surface point projected to $\mathbf{u}$. This is done by comparing $\mathcal{I}_{l,P}$ and $\mathcal{I}_{l,I}$:

$$\text{codeword}_{\mathbf{u}}(l) = \begin{cases} 1, & \text{if } \mathcal{I}_{l,P}(\mathbf{u}) > \mathcal{I}_{l,I}(\mathbf{u}) \\ 0, & \text{if } \mathcal{I}_{l,P}(\mathbf{u}) < \mathcal{I}_{l,I}(\mathbf{u}) \\ \text{unreliable} & \text{if } |\mathcal{I}_{l,P}(\mathbf{u}) - \mathcal{I}_{l,I}(\mathbf{u})| < T \end{cases} \tag{4.1}$$

The screen position corresponding to the decoded label can be identified by comparing the decoded label value to labels computed from original patterns. The offset $\mathbf{q} = (q_x \; q_y)^T$ in the table of labels matching to the decoded label is the position on the screen that illuminated the projected point.

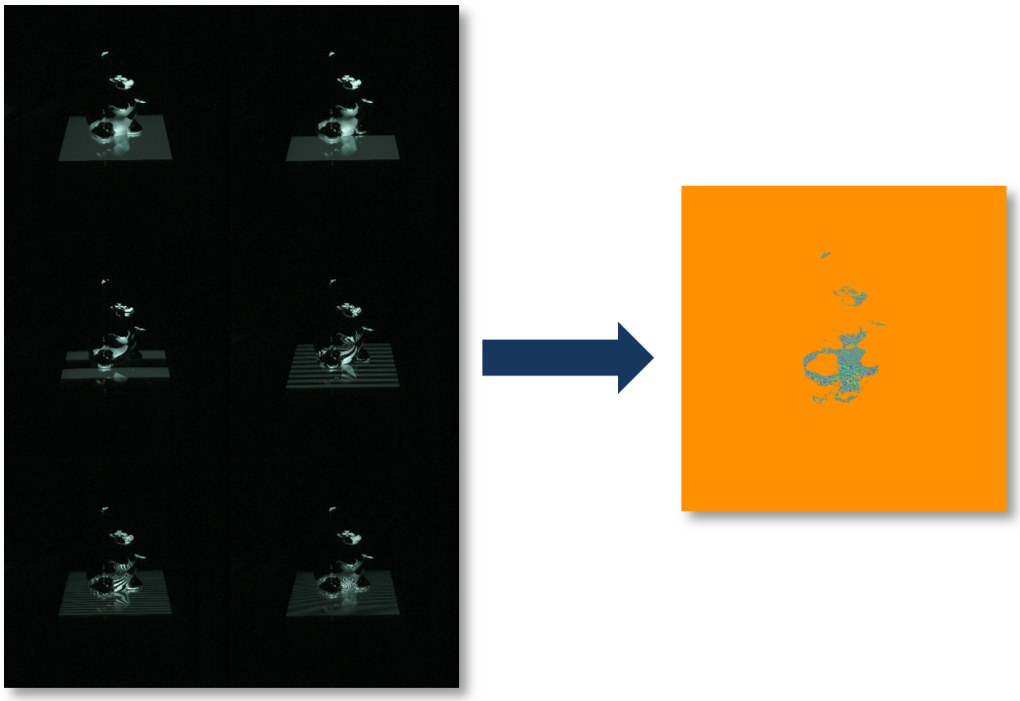

Figure 4.3: A set of six images, where the mirror bunny is illuminated with the tablet from the bottom, and corresponding decoded labels. Note, that on the surface on the tablet, there is also observable pattern, reflected from the bunny. This area is also decoded and it can not be distinguished from the labels, belonging to the bunny.

In reality, due to non-perfect mirroring objects, it is usually not possible to reliably decode all the bits - at some point, the high-frequency patterns get blurred and it is no longer possible to distinguish between the pattern and its inverse. In [FCMB09], this fact was exploited for determining the level of specularity of the surface. Furthermore, due to the curvature of the object and, consequently, slightly different distances of the surface points from the screen and possibly even a varying level of reflectance across the object, it is not possible to use the same number of bits for the decoding of the image points, see Figure 4.4. For that reason, we implemented a *fuzzy decoding* procedure. In theory, a unique label is decoded from a sequence of $\frac{\kappa_p}{2}+1$ bits, relating the projected point to a unique pixel on the display screen. In practice, only the first K-bits with $K < \frac{\kappa_p}{2}+1$ can be reliably decoded. For this reliably decoded portion, it is not possible to assign a unique label, since more than one label share exactly the same sequence of the first K bits. The labels sharing the first K-bits with the decoded codeword correspond to a certain region on the display screen (Figure 4.4). The less bits

can be used for decoding, the larger is the ambiguity in the portion of the screen that illuminated the projected point. For computing the normal, we use the center of the screen area, that could have been identified from the decoded codeword.

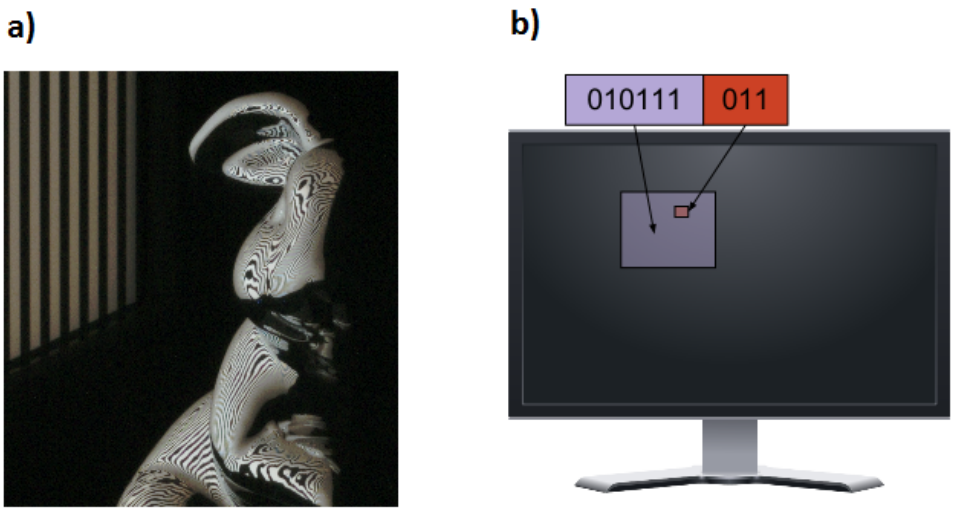

Figure 4.4: a) A pattern, illuminating a region of the bunny. While in many areas it is still possible to observe the pattern, in some regions (eye, ear), the pattern is strongly blurred and a reliable decoding is impossible. b) The first K-bits of the codeword can only relate to a certain region on the screen (violet), while the whole codeword can relate to a certain pixel (red).

After decoding the patterns, light maps $\mathcal{L}_{i,j}$ are stored as images, where each pixel stores the value of a label, that relates the projected point to the area of the screen which illuminated the point, see Figures 4.2 and 4.3. Note, that the pixel values do not store the 3D location of the source illuminant directly, but with the geometric calibration data of the screen, this information is trivially computable.

Cleaning the Labels

Looking at the first image in Figure 4.5, it can be observed, that the direct result after the pattern decoding contains several noisy decodings, which might be additional sources of errors in the normal computation and integration process. As a post-processing step after decoding, a cleaning of the labels is performed in two steps. First, decoded labels are thresholded based on the number of bits used for decoding - in the case that a very low number of bits was used, the ambiguity in the area of the screen which illuminated the projected point is rather large. Simply taking a center of this area as source of illumination might lead to large errors in

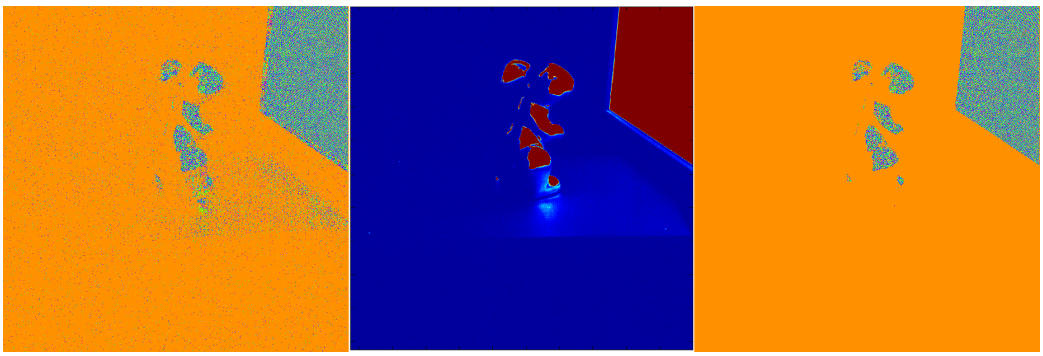

Figure 4.5: From left to right: the noisy decoding from the mirror bunny dataset, the sum of contrasts between pattern and its inverse for each pattern-inverse pair and a thresholded result, based on sum of intensities.

the normal estimates. A second criterion for cleaning is based on contrasts in the captured pattern-inverse image pairs $(\mathcal{I}_{l,P}, \mathcal{I}_{l,I})$, $l = 1 \ldots \frac{\kappa_p}{2} + 1$, used for decoding. For each pixel, the average of contrasts between the decoded patterns and their inverses is computed and thresholded. From the second image in Figure 4.5 it can be observed, that these contrast average values form a clear border between the reliably decoded and noisy areas.

4.2.2 Setup and Calibration

We perform the necessary measurements of the specular object using the turntable-based setup, described in Chp. 5, and extend it using display screens. To illuminate the object fully with the structured illumination, we place one screen at the side of the turntable and one directly on top of the turntable, below the object. In practice, we used LCD screens at the side and a tablet computer on top of the turntable, see Figure 4.6. For the geometric calibration purposes, we placed the LCD screen in a way that some cameras can see at least a small portion of it (the tablet computer is always seen by all cameras, except for the area occluded by the object). By applying rotations to the turntable, at which the object and the tablet are placed, we are able to record the object from multiple sides by illuminating it with a sequence of patterns and capturing the images for each rotation, see Figure 4.6.

For establishing correspondences between illuminated points and their illuminators, it is essential that the transformation between the coordinate frames of the screens and the world coordinate frame is known precisely. In most of the shape-from-specularity based approaches, the utilized screen is not directly visible to the cameras. That necessitates the use of planar or spherical mirrors in order

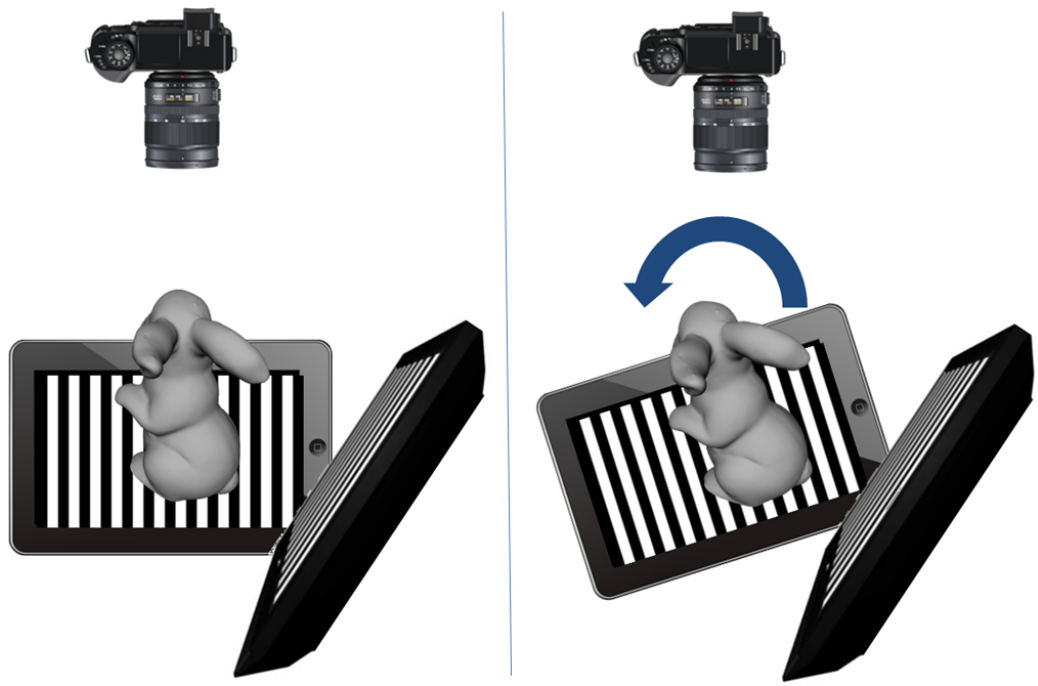

Figure 4.6: The setup for the measurement of light maps: the cameras are fixed on an arc and a display screen is put on the side. The tablet computer used illuminating from the bottom and the object of interest are placed on top of the rotation table.

to determine the position of the screen displaying the patterns. For example, in [BW10, Cla10], a planar mirror is used, placed in a way that it reflects the screen patterns towards the camera. In this case, the position of the mirror must be determined first (using e.g. markers at the corners or by attaching a calibration target to the mirror). Then, by observing the patterns being reflected from the screen to the mirror, it is possible to determine its pose with respect to the mirror (i.e. the mirror plays the role of a virtual camera). Knowing both transformations, the relation between the display target and the camera can be obtained as a composition of both. Alternatively, the authors of [FHB09] propose to use two spherical mirrors (of known radius and pose) and display a series of Gray codes at the target screen. Observing the reflected Gray codes in both spheres it is possible to identify for each reflected point to which pixel of the screen it corresponds. For two decoded labels from two spheres, denoting the same location on the screen, it is possible to triangulate the 3D pose by intersecting the rays casted from the camera and reflected from the sphere. Then, a 3D plane is fitted to the point cloud of estimated 3D screen pixel locations and another optimisation step is performed in order to relate the screen-plane 3D coordinates with the screen pixels.

For our setting, we propose a different approach to the screen calibration, which is simple to implement and allows a precise calibration. We place the side screen in a way that at least some of the cameras (we have 11 cameras, placed on the arc behind the turn-table) can see at least a small portion of it, see Figures 4.6 and 4.7. As the patterns are seen from different cameras (in theory, two would suffice), it is possible to establish correspondences between the cameras for the visible portions of the screen precisely using classic structured light technique and to triangulate the points in order to obtain a precise and dense set of points belonging to the visible portion of the display screens. This way is not necessary to place any additional mirrors to the setup or to perform any intermediate calibration which may be the source of additional errors, and it is not necessary to capture any additional data. The correspondences between the views for the screen surface points can be computed directly from the Gray codes displayed on the screen. Of course it is necessary to segment the labels which have been decoded on the object from the labels which have been decoded on the display screens. Due to the observation, that the number of reliably decoded labels is always higher on the screens than on the object, this segmentation is trivial. It is also worth noting, that the segmentation step could be avoided by capturing an additional set of images, without the object being placed in the scene.

The result of this step is a point cloud of size M (see Figure 4.7), representing a small portion of the screen. For each triangulated point $\mathbf{p}_m = (x_m\ y_m\ z_m)^T$, $m = 1 \ldots M$ in the point cloud the offset $\mathbf{q}_m = (q_{x,m}\ q_{y,m})^T$ from the top-left corner of the screen can be determined from the sequence of decoded bits in addition to the 3D spatial information, rendering the fitting of 2D screen points to 3D points belonging to the surface of the screen unnecessary as we have the information already. Having this information, it is possible to compute the coordinate frame of the screen: its upper-left point $\mathbf{o}$ as the origin and two vectors that span the screen vector space: the horizontal vector $\mathbf{a}$ and the vertical vector $\mathbf{b}$, see Figure 4.7. The objective function we seek to optimise to compute the base vectors is:

$$Q = \sum_{m=1}^{M} (\mathbf{p}_m - (\mathbf{o} + q_{x,m}\,\mathbf{a} + q_{y,m}\,\mathbf{b}))^2 \tag{4.2}$$

We compute the solution using the linear least-squares method by solving the system of linear equations $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{x}$ is a 9×1 vector of unknowns, $\mathbf{b}$ is a

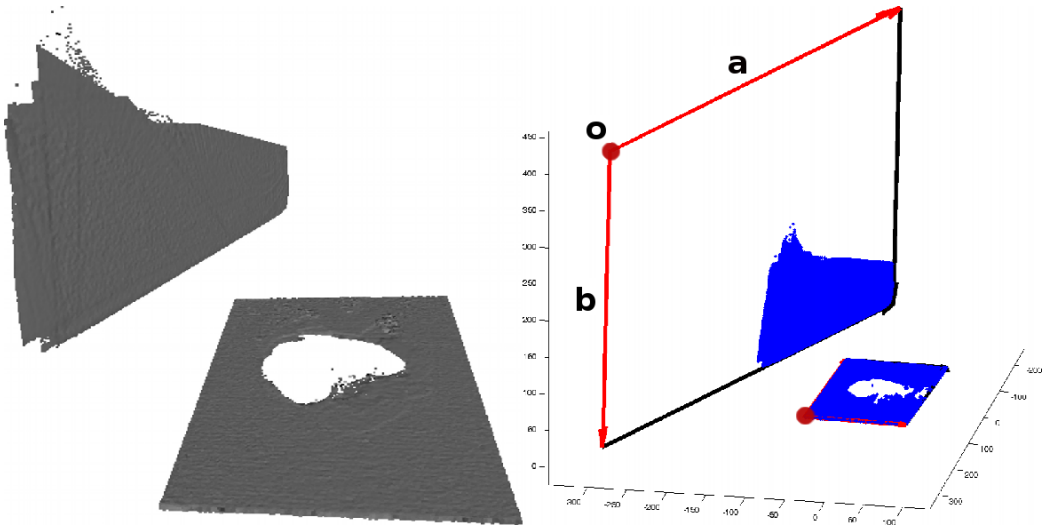

Figure 4.7: The point cloud, representing a portion of the displays (LCD monitor and tablet) *(left)* and fitted screen-base frames, coloured with red *(right)*.

$3 \times M$ matrix $[\mathbf{p}_1\ \mathbf{p}_2\ \ldots\ \mathbf{p}_M]^T$ and the matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_{3\times3} & q_{x,1}\mathbf{I}_{3\times3} & q_{y,1}\mathbf{I}_{3\times3} \\ \mathbf{I}_{3\times3} & q_{x,2}\mathbf{I}_{3\times3} & q_{y,2}\mathbf{I}_{3\times3} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \mathbf{I}_{3\times3} & q_{x,M}\mathbf{I}_{3\times3} & q_{y,M}\mathbf{I}_{3\times3} \end{bmatrix}. \tag{4.3}$$

The system is solved by computing $\mathbf{x} = \mathbf{A}^+\mathbf{b}$, where $\mathbf{A}^+$ is a Moore-Penrose pseudo-inverse of $\mathbf{A}$.

Once the screen is calibrated, i.e. the origin of the display screen and the two orthogonal vectors spanning the coordinate frame are known, and relations between projected surface points and screen points illuminating them are established, it is simple to compute the respective 3D position in the global coordinate frame. The 3D position $\mathbf{x}_e$ on the screen for the decoded screen offset $\mathbf{q} = (q_x\ q_y)^T$ is computed as follows

$$\mathbf{x}_e = \mathbf{o} + q_x\,\mathbf{a} + q_y\,\mathbf{b}. \tag{4.4}$$

4.2.3 Generating Normal Hypotheses and Normal Integration

After obtaining light maps for each camera, the vector field $\mathbf{N}(\mathbf{x})$ should be computed and integrated using the methods explained in Chp. 3. However, a minor

adaptation is required for the case when light maps are the input. In the multi-view normal integration problem considered in Chp. 3, normals have already been determined and are propagated in the volume along the respective viewing rays. Here, due to the violation of the distant light assumption, normals can not be computed a priori - the known camera pose and source of illumination define a family of possible normals (see Figure 4.8), depending on the distance of observed surface point from the camera plane. This is an chicken-egg problem: in order to compute correct normal, the pose of surface point would have to be known. It was first time proposed by [SWN88] to resolve this ambiguity by adding additional views, the correct surface point is then the one, where normals suggested by multiple views will agree. In the context of [BS03], the family of normals along the camera viewing ray, based on known source of illumination, are called *normal hypotheses*. The idea of resolving the depth-normal ambiguity using multiple views was in practice utilized in [BS03, NWR08], however, on very simple objects (spoon, coin, curved mirror). Note, that this kind of objects are rather easy to illuminate as they do not cause self-occlusions. For the same reason, inter-reflection effects are minimal. In order to deal with all these effects and address a wider range of objects, a very robust algorithm for fusion and integration of these normal hypotheses is essential.

As the input in this case are light maps $\mathcal{L}_{i,j}$ rather than normal-fields $\mathcal{N}_i$, $i = 1 \dots \kappa_c$, $j = 1 \dots \kappa_d$, normals have to be computed during the normal-integration process. In contrast to the classic multi-view normal integration problem discussed in Chp. 3, in this case, we do not back-project normals to the volume, but rather labels. Based on the back-projected labels, for each camera-screen pair, a normal hypothesis $\mathbf{h}_{i,j}(\mathbf{x})$ is computed at the spatial point $\mathbf{x} \in V$.

The combined generation of normal hypotheses and vector field computation is implemented as follows. At each spatial point $\mathbf{x} \in V$, a normal hypothesis $\mathbf{h}_{i,j}$ is computed with respect to each camera-screen pair. To compute a single normal hypothesis, the 3D point $\mathbf{x}$ is projected to the light map $\mathcal{L}_{i,j}$. The back-projected point $\mathbf{u}_{i,\mathbf{x}} \in \Omega$ relates the label $\mathcal{L}_{i,j}(\mathbf{u}_{i,\mathbf{x}})$ to the screen offset $\mathbf{q} = (q_x \; q_y)^T \in D_j$ from the origin (upper-left corner) of the display screen. As the screen is calibrated, the 3D point $\mathbf{x}_e$ which illuminated $\mathbf{x}$ can be computed according to equation (4.4). Then the normal hypothesis $\mathbf{h}_{i,j}(\mathbf{x})$ can be computed as the bisector of the vectors $\mathbf{l}$ and $\mathbf{v}$, where $\mathbf{v} = \frac{C_i.COP - \mathbf{x}}{\|C_i.COP - \mathbf{x}\|}$ and $\mathbf{l} = \frac{\mathbf{x}_e - \mathbf{x}}{\|\mathbf{x}_e - \mathbf{x}\|}$, see Figure 4.1.

In order to avoid propagating the labels belonging to the portions of the screen observed by the camera, a ray is casted from the camera through $\mathbf{u}_{i,\mathbf{x}}$ and intersected with the geometry representing the screens. In case the ray hits the plane representing the screen D_j, the 3D location of the intersection is compared with the 3D point decoded from $\mathcal{L}_{i,j}(\mathbf{u}_{i,\mathbf{x}})$. In case the distance between these two points is very small, the respective label was in fact decoded in the area, representing the

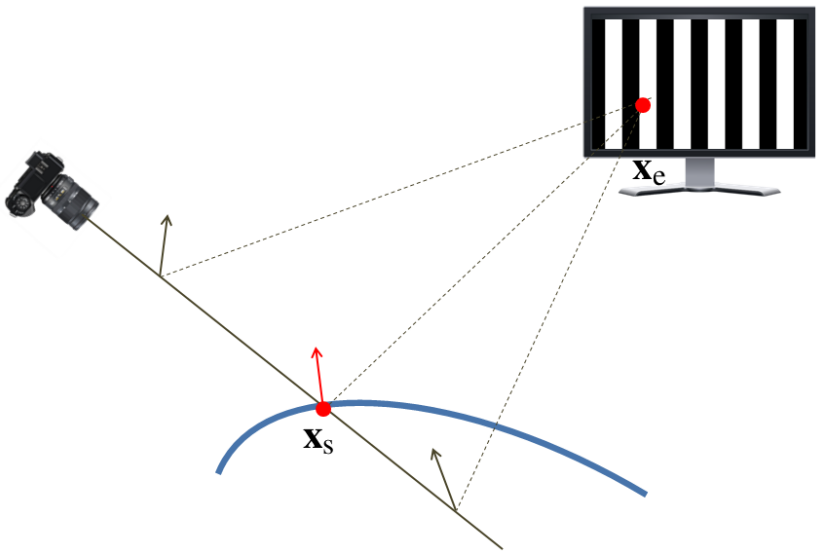

Figure 4.8: The depth-normal ambiguity along the viewing ray.

screen and not the specular object.

The result of normal hypothesation is a set of hypothesised normals (analogue to the set of back-projected normals from Chp. 3) which are mapped to the feature space. Then an analysis of the probability density function of the observed data is performed for this set, in order to compute vector field and normal consistency value, just as explained in Chp. 3.

CHAPTER 5

EVALUATION

The evaluation of the multi-view normal field integration algorithm was performed on both synthetic and real-world data sets. In case of the synthetic data, renderings of the object were synthesized from several views using the GPU and the OpenGL libraries. The object was rendered using a pixel shader, which encodes the coordinates of normals in the RGB channels of the image. For the evaluation on the real-world data sets, first normal fields of lambertian objects were captured and computed using classic photometric stereo technique [Woo89]. After that, we evaluated performance on a mirroring bunny object, where normals were computed as described in Chp. 4.

The tests were performed on the machine with Intel Core i7 CPU with 12GB of installed memory, running on Microsoft Windows 7 64-bit platform. The algorithm is implemented in C++ and certain parts can run on multiple threads.

5.1 Synthetic Datasets

For the evaluation on synthetic data sets, we implemented a simulation environment. The simulator allows to load the existing 3D models and can synthesize renderings from views specified by the intrinsic and extrinsic camera parameters. Additionally, it it possible to design an arbitrary setup, place cameras in the scene and export the camera parameters to a XML file. For rendering the 3D objects with normal information instead of shading information, we implemented a normal-colouring pixel shader, that stores for each pixel color-coded normal information, instead of shading information. To make the simulation conditions as similar as possible to the our experimental setup, we placed virtual cameras for our synthetic experiments in the upper hemisphere, see Figure 5.1.

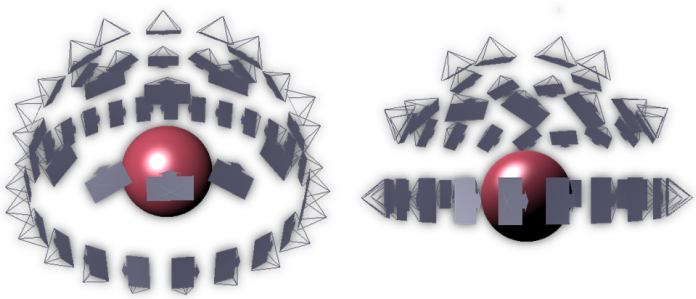

Figure 5.1: The renderings of our synthetic setup with cameras placed in the upper hemisphere around the object of interest.

5.1.1 Sphere

The first experiments were performed on a simple sphere geometry as a proof of concept. Due to placement of the cameras, no camera observed the bottom part of the sphere, see Figure 5.1. In the Figure 5.2, the results of sphere reconstruction using 30 cameras are visualized. The top row shows images taken from above the sphere and in case of the bottom row, images were synthesized from the side view. The left side shows the original (reference) sphere and on the right side, the 3D reconstruction is shown. Except in the very bottom region of the sphere (this can be observed in the side view), where no data is available, the sphere geometry is faithfully reconstructed.

5.1.2 Buddha

The next test we performed on the challenging Happy Buddha model. Reconstruction of this model is difficult due to its self-occluding geometry consisting of several fine details and thin regions. Visualizations of the reference 3D model and the respective reconstructions obtained from 75 normal fields are displayed in Figure 5.3. The Buddha reconstructions using 10 and 20 cameras are visualized in Figure 5.4.

Certain fine-detail areas of the reconstructed model are visualized from closer proximity in Figure 5.5. Especially interesting is the comparison of the top-area of the model between the original 3D model and the reconstruction, obtained from normal fields. The area directly below the Buddha's hands is partially occluded and hence, difficult to capture. While the original model produces in this area a spurious reconstruction, our reconstruction of the reference model produces in this area a much smoother result and still faithfully preserves other fine surface details.

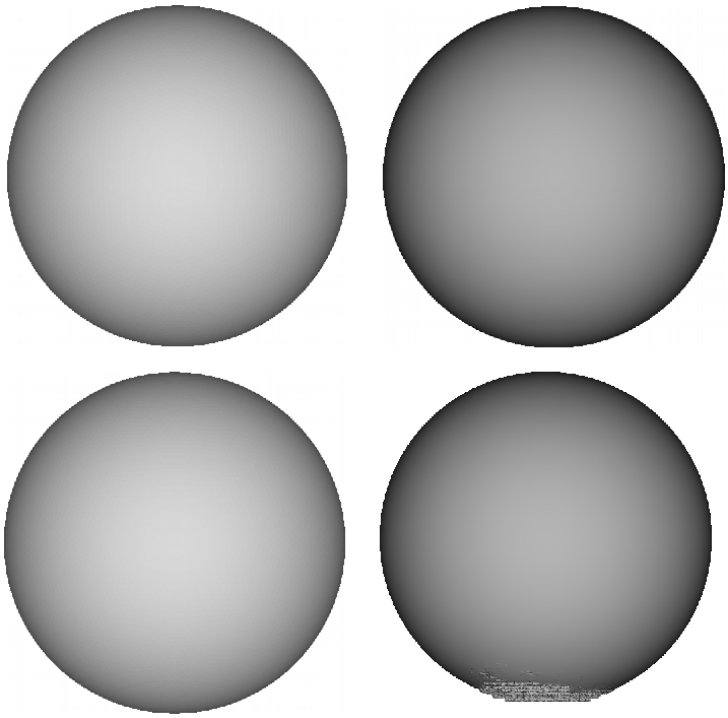

Figure 5.2: Results obtained on the simple synthetic sphere: the top row shows the results from the top view (original model and reconstruction) and the bottom row images were synthesised from the side view. In this case, 30 cameras were used. Note, that no camera observed the bottom part of the sphere.

5.2 Real Datasets

5.2.1 Setup

For the measurements of the physical objects, a turn-table based setup, visualized in Figure 5.7, was used. In the setup, 11 software controlled Vistek cameras are placed on an arc above the software-controlled turntable. In the hemisphere above the turntable, 198 LEDs are placed, which we use for our photometric stereo experiments, and four software-controlled LG projectors are available (which have not been used in this work). The cameras and lights were already geometrically and radiometrically calibrated. The discussion of the calibration procedures is out of the scope of this thesis, however, for capturing the specular data we had to add extra components - display screens. The geometric calibration of the screens is discussed in Chp. 4.

5.2.2 Nearly Lambertian Mask Dataset

As pointed out and explained in Chp. 2, there exist many techniques for normal estimation. In order to be comparable with previous approaches to multi-view normal field integration [CLL07, Dai09], we decided to use photometric stereo on nearly lambertian object as a first test on the real-data, although there are normal estimation approaches that can handle a considerably larger range of materials, e.g. [HS05, GCHS05, ZBK02]. Additionally, we used an object that exhibits also non-lambertian reflectance behaviour (see Figure 5.8), which we consider as a good test of robustness. Furthermore, the BRDF of our test object varies spatially and the distant-light assumption is violated in our setup.

In our multi-view setting, we have 198 images for each camera, one for each light source, and a 198×3 matrix $\mathbf{L}$ of light source directions. The setup is geometrically and radiometrically calibrated, i.e. light source positions are known, images were linearised and corrected for possibly varying light source intensities and light fall-off.

Normal estimates are computed using the linear-least squares fitting method described in 2.2.3. While this approach can compensate well for small deviations, it is very sensitive to strong outliers. Hence, specularities and shadows will have strong influence on the quality of the estimated normals. We implemented a simple outlier removal prior to the least-squares fitting by thresholding image intensities, based on simple image statistics i.e. strong outliers are rejected (too bright and too dark pixels). There also exist more sophisticated robust photometric stereo approaches, e.g. [WGS$^+$11, Aiz12].

Regarding the violations of the distant light source assumption, it would be possible to account for that by performing a linear-least squares normal computation at each corner, taking the corner position as the hypothesised surface point, with respect to which view and light vectors would be computed. In this case, the input to the multi-view normal field integration would not be estimated normals but light intensities for each camera directly, leading to the computation of normal estimates during the normal integration process. That would be equivalent to the concept of generating the normal hypotheses, proposed in Chp. 4 for 3D reconstruction of highly-specular objects. Alternatively, it would be possible to recompute normal fields after the first iteration of the algorithm based on a coarse geometry. In practice, light vectors for each pixel have been computed with respect to the center of the volume bounding the object, as the goal of this thesis is not an accurate normal estimation but their robust integration. For that reason, no attempts were made to estimate normals more robustly and accurately. The output of the normal estimation process are κ_c normal fields for κ_c cameras.

In this case, a subset of six cameras available in the setup was used and 12 rotations of the turntable were performed, leading to 72 views (normal fields) in

total. The visualization 5.9 illustrates four synthesized images of the mask reconstruction, that reproduce the original object shown in Figure 5.8 with a rather high precision. It should be noted, that the three small holes present in the reconstructed model (especially notable in the first mask reconstruction image), are actually part of the original object, although it might not be obvious from the mask images in Figure 5.8.

5.2.3 Mirroring Bunny Dataset

In this section, very promising results obtained on the mirror bunny object shown in Figure 5.10 are presented. From the reconstruction results displayed in Figure 5.13, it can be concluded that the proposed method is able to produce an accurate reconstruction result from normals measured on a highly-specular object. While in most of the areas the surface is reconstructed up to a very high precision, there are a few areas which are not perfectly reconstructed due to physical limitations of the employed setup, see Figures 5.7 and 5.11. To be precise, the following regions are problematic: the area below the chin of the bunny and the two ear-concavities. Even though the algorithm is able to fill the regions where no data was recorded to some degree (the slice of computed surface consistency function highlights the regions where the data is missing, see Figure 5.11), it is clear that the reconstruction in that areas is not entirely correct. The main reason is that recording data in that regions is difficult which is due to fact, that in the setup, the lowest camera is observing the turntable with an angle of 15° and is never really able to observe the reflected pattern, see Figures 5.12 and 5.7. Of course in these regions inter-reflections are also problematic, but we believe that placing another camera to the setup would significantly improve the result.

In Figure 5.14, there is a clearly visible ridge on the back of the bunny. First observations of this ridge on the reconstructed geometry lead to the impression, that this is a systematic error. However, a closer inspection of the real bunny model showed that there is actually a ridge due to manufacturing process of the model. In this area, the two sides of the underlying material are stitched together. Interestingly, we, as human observers, did not recognize the ridge before the reconstructed geometry revealed it.

Figure 5.3: The Happy Buddha results: original model *(left column)* and reconstructions *(right column)*, using 75 cameras.

Figure 5.4: The Happy Buddha results: reconstructions using 10 cameras *(left column)* and reconstructions using 20 cameras *(right column)*.

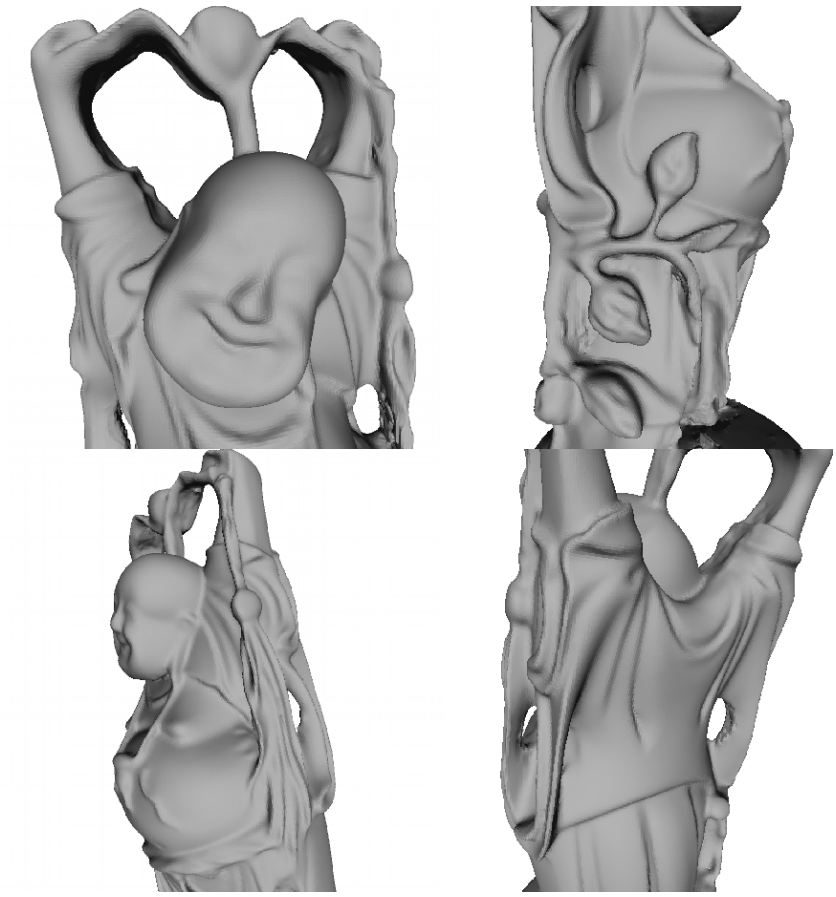

Figure 5.5: The Happy Buddha, zoomed areas.

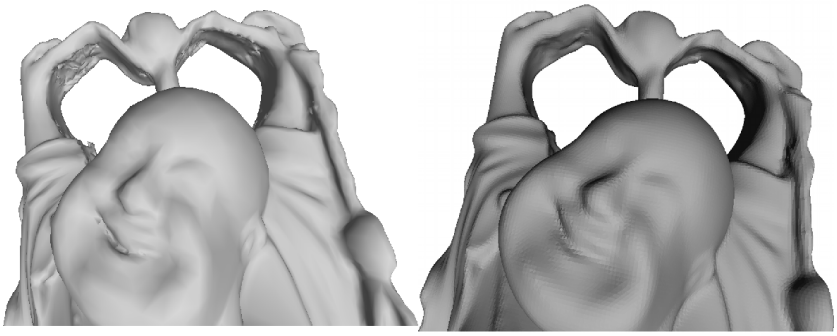

Figure 5.6: The Happy Buddha, occluded areas: original model *(left)* and reconstruction *(right)*.

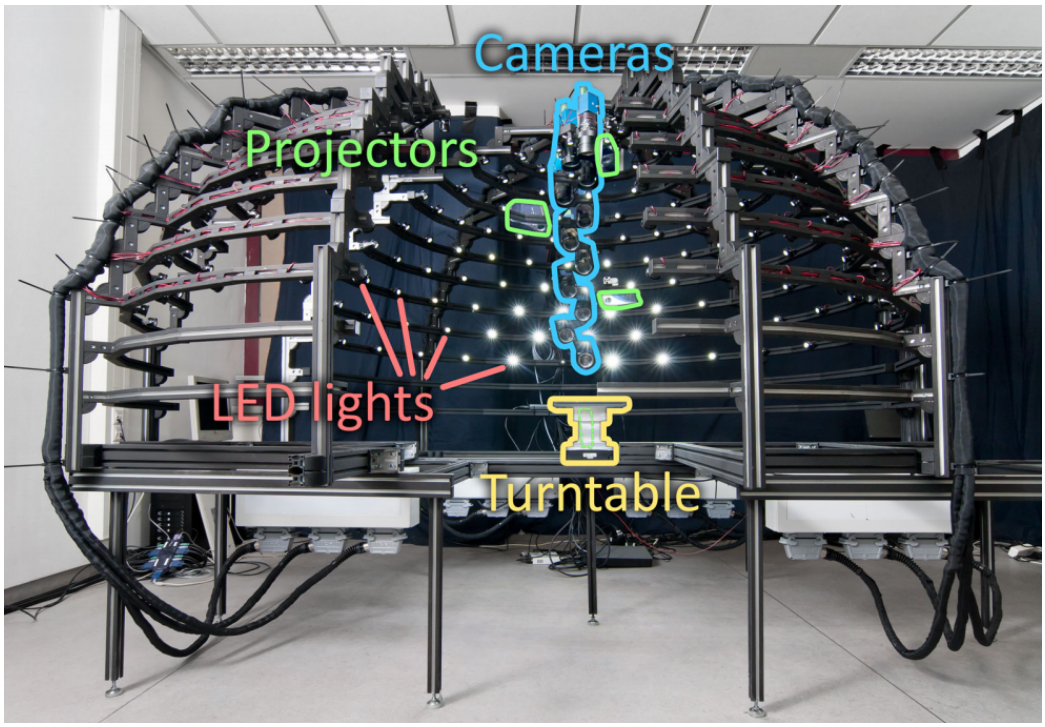

Figure 5.7: Image of the experimental setup.

Figure 5.8: Four images of the original mask object.

Figure 5.9: The reconstruction results of the mask object.

Figure 5.10: The photos of the mirror bunny.

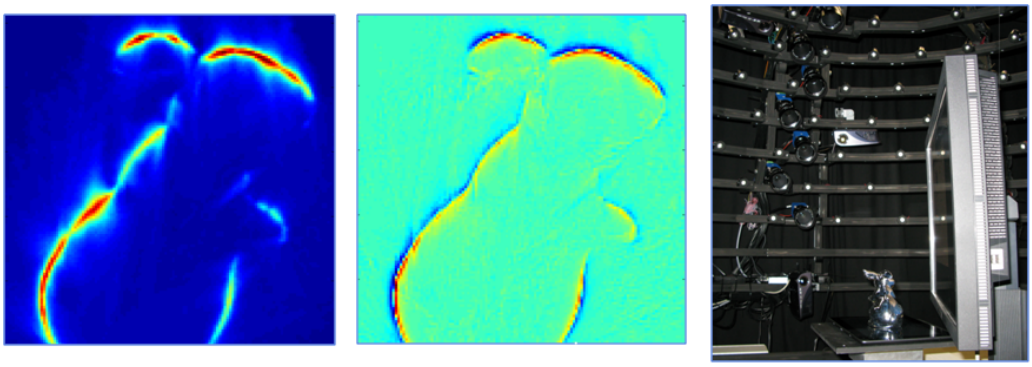

Figure 5.11: From left to right: slice of the surface consistency, computed on the data from the bunny dataset, slice of the divergence of the corresponding vector field and an image of the setup, showing the bunny, screens and lower cameras.

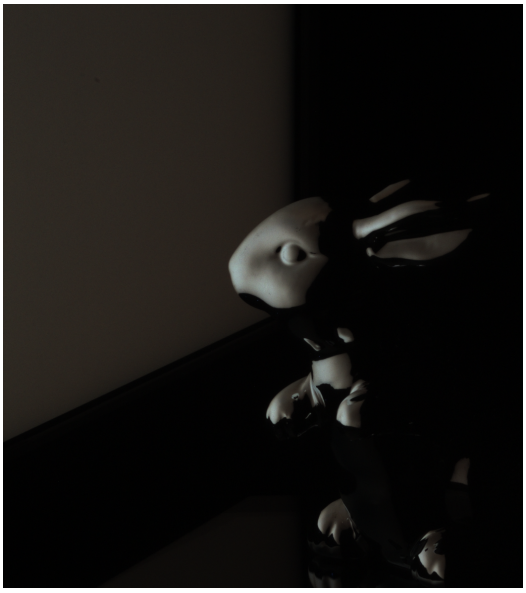

Figure 5.12: Image of mirror bunny, captured from the lowest camera.

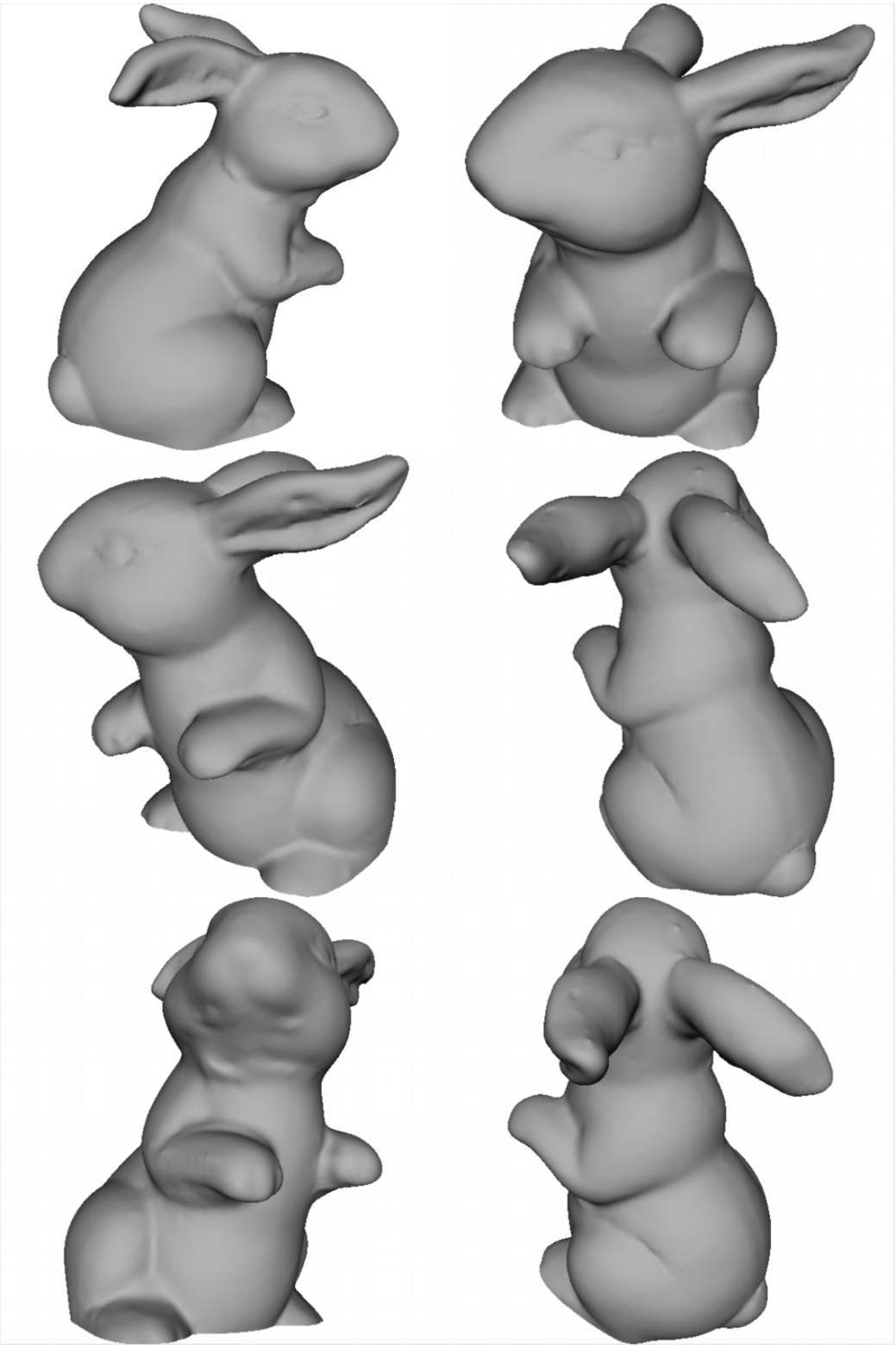

Figure 5.13: The reconstruction results of the mirror bunny object.

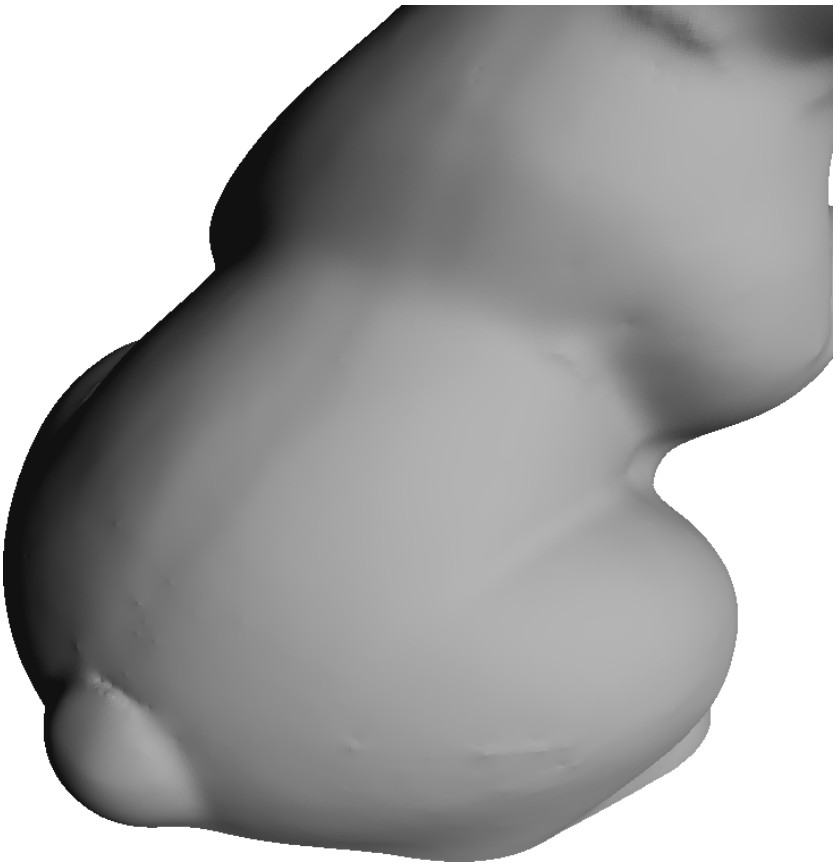

Figure 5.14: Closer look at the back of the bunny, where the vertical ridge due to manufacturing process is clearly visible.

CHAPTER 6

CONCLUSIONS

6.1 Conclusion

In the scope of this master thesis, we demonstrated a novel, outlier-robust multi-view normal field integration algorithm formulated in a variational framework. The method is based on multiple successive global optimisation steps via convex-relaxation based continuous max-flow method in an octree discretised volume, optimising a minimal surface based energy functional. The main novelty of this algorithm is a new, robust computation of surface in-out constraints and surface consistency based on a spatial mean-shift clustering of the observed normal fields.

By conducting the experiments on synthetically generated datasets, we showed that our algorithm can handle precise geometry reconstruction of challenging objects with complex, self-occluding geometry. Furthermore, the algorithm only relies on normal information data, which is essential in scenarios, where other visual cues are hard to obtain (for example, in case of highly-specular objects).

This method is the first one successfully applied on the normals estimated from real-world measurements. The algorithm was first applied on nearly lambertian objects, using classic photometric stereo [Woo89] for the estimation of normal fields. The algorithm is not only able to successfully reconstruct geometry of the object from captured normal fields, but is also able to deal with object, whose surface reflectance characteristics vary spatially and even violate lambertian assumptions.

In further evaluations, we demonstrated state-of-the-art results in the area of 3D reconstruction of highly-specular objects. For acquiring such objects, a turntable based setup was employed, which is capable of nearly completely illuminating the object with structured Gray code based patterns from a close proximity of the object. For decoding Gray codes, where the number of patterns which can be used for decoding varies across the image domain, a robust, fuzzy decoding was implemented. To recover the full surface, the multi-view normal field integration was adapted for the input in form of light maps. Ambiguities, coming from vio-

lations of the distant-light assumption are resolved by hypothesising normals and integrating them with the proposed multi-view normal field integration algorithm. The result is a full, high-quality geometry reconstruction of a mirroring bunny test object.

6.2 Future Work

Even though we believe this work presents significant improvement in the area of gradient-based 3D object reconstruction and carries a potential for addressing 3D reconstruction of objects consisting of large variety of materials, there is still space for improvements. First, since clustering is done on a spherical surface, it would be natural to use a Riemannian metric. It is unclear, how the Euclidean approximation of distances between samples affects the mean-shift clustering results. Secondly, using the geometry from initial iterations in future iterations as an approximation of visibility could be beneficial. Using this visibility approximation, back-projected normals with large angular deviations from normals of surface from previous iteration could be rejected. That could not only improve the quality of reconstruction but also speed up the execution: the size of the input to the mean-shift clustering step could be approx. halved if angular deviations only up to $\frac{\pi}{2}$ would be permitted.

Furthermore, it is not clear how our choice of numerical scheme for the optimisation affects the results. It would be certainly interesting to make a comparison to the SOR algorithm proposed in the context of 3D reconstruction in [KKB+07]. The bottle-necks of our approach are actually the mean-shift clustering step and the numerical optimisation by continuous min-cut, that are both solid candidates for parallelisation.

Although the proposed algorithm has a property of being able to rely purely on normal information, additional visual cues can always improve the result, when they are available. Additionally, to see the real potential of the method, testing (and possibly even combining) it with other normal estimation techniques would be interesting.

In this context also the reconstruction of heterogeneous objects could be addressed, using varying normal estimation approaches according to the reflectance characteristics of the materials. The output of all normal estimation steps would then be normal fields, that can be easily integrated together using this algorithm.

Furthermore, for the reconstruction of specular objects, testing different illumination coding strategies would be in place with the especially interesting case of phase shifting. The quality of the results might also be improved by weighting the normals according to the area of the screen, which was identified to be a source of pixels illumination - naturally a larger area raises the ambiguity in the

normal estimate.

BIBLIOGRAPHY

[ABS11] Y. Adato and O. Ben-Shahar. Specular flow and shape in one shot. BMVC, 2011.

[Aiz12] K. Aizawa. Robust photometric stereo using sparse regression. CVPR, 2012.

[AVBSZ07] Y. Adato, Y. Vasilyev, O. Ben-Shahar, and T. Zickler. Towards a theory of shape from specular flow. ICCV, 2007.

[BHB11] J. Balzer, S. Holer, and J. Beyerer. Multiview specular stereo reconstruction of large mirror surfaces. CVPR, 2011.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[BJK07] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *Int. J. Comput. Vision*, May 2007.

[BK03] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. ICCV, 2003.

[BK04] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE TPAMI*, September 2004.

[BM92] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE TPAMI*, February 1992.

[BS03] T. Bonfort and P. Sturm. Voxel carving for specular surfaces. ICCV, 2003.

[BVZ01] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, November 2001.

[BW10] J. Balzer and S. Werling. Principles of Shape from Specular Reflection. *Measurement*, 2010.

[BZK86] A. Blake, A. Zimmerman, and G. Knowles. Surface descriptions from stereo and shading. *Image Vision Comput.*, November 1986.

[CGS06] T. Chen, M. Goesele, and H.-P. Seidel. Mesostructure from specularity. CVPR, 2006.

[Che95] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE TPAMI*, August 1995.

[CKS97] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int. J. Comput. Vision*, February 1997.

[CL96] B. Curless and M. Levoy. A volumetric method for building complex models from range images. SIGGRAPH, 1996.

[Cla10] J. J. Clark. Photometric stereo using lcd displays. *Image Vision Comput.*, April 2010.

[CLL07] J. Y. Chang, K. M. Lee, and S. U. Lee. Multiview normal field integration using level set methods. CVPR, 2007.

[CM02] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI*, May 2002.

[CSC+10] Y. Cui, S. Schuon, D. Chan, S. Thrun, and C. Theobalt. 3d shape scanning with a time-of-flight camera. CVPR, 2010.

[CT11] F. Calakli and G. Taubin. Ssd: Smooth signed distance surface reconstruction. *Comput. Graph. Forum*, 2011.

[CV99] T. Chan and L. Vese. An active contour model without edges. SCALE-SPACE, 1999.

[Dai09] Z. Dai. A Markov random field approach for multi-view normal integration. Master's thesis, The University of Hong Kong, Pokfulam, Hong Kong, 2009.

[Dig51] J. van Diggelen. A photometric investigation of the slopes and heights of the ranges of hills in the maria of the Moon. *Bulletin of the Astronomical Institute of the Netherlands*, 11, 1951.

[DP00] J.-D. Durou and D. Piau. Ambiguous shape from shading with critical points. *J. Math. Imaging Vis.*, April 2000.

[ES04] C. H. Esteban and F. Schmitt. Silhouette and stereo fusion for 3d object modeling. *Comput. Vis. Image Underst.*, December 2004.

[FC88] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE TPAMI*, July 1988.

[FCMB09] Y. Francken, T. Cuypers, T. Mertens, and P. Bekaert. Gloss and normal map acquisition of mesostructures using gray codes. *Advances in Visual Computing*, November 2009.

[FF62] L. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.

[FH06] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE TIF*, September 2006.

[FHB09] Y. Francken, C. Hermans, and P. Bekaert. Screen-camera calibration using gray codes. CRV, 2009.

[FK98] O. D. Faugeras and R. Keriven. Complete dense stereovision using level set methods. ECCV, 1998.

[FS02] P. Favaro and S. Soatto. Learning shape from defocus. ECCV, 2002.

[FY07] N. Funk and Y.-H. Yang. Using a Raster Display Device for Photometric Stereo. `http://www.njfunk.com/research/crv07-slides.pdf`, 2007. [Online; accessed 12-March-2012].

[GCHS05] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and spatially-varying brdfs from photometric stereo. ICCV, 2005.

[GPS89] D. M. Greig, B. T. Porteous, and A. H. Seheult. Exact Maximum A Posteriori Estimation for Binary Images. *Journal of the Royal Statistical Society*, 1989.

[GT86] A. V. Goldberg and R. E. Tarjan. A new approach to the maximum flow problem. STOC, 1986.

[HB86] B. K.P. Horn and M. J. Brooks. The variational approach to shape from shading. *Comput. Vision Graph. Image Process.*, February 1986.

[HDD+92] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. SIGGRAPH, 1992.

[HLKF95] J.-W. Hsieh, H.-Y. M. Liao, M.-T. Ko, and K.-C. Fan. Wavelet-based shape from shading. *Graph. Models Image Process.*, July 1995.

[HO11] M. Harker and P. O'Leary. Least squares surface reconstruction from gradients: Direct algebraic methods with spectral, tikhonov, and constrained regularization. CVPR, 2011.

[Hor70] B. K.P. Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. Technical report, Cambridge, MA, USA, 1970.

[Hor90] B. K. P. Horn. Height and gradient from shading. *Int. J. Comput. Vision*, September 1990.

[HS05] A. Hertzmann and S. M. Seitz. Example-based photometric stereo: Shape reconstruction with general, varying brdfs. *IEEE TPAMI*, August 2005.

[HYJK09] T. Higo, Matsushita Y., N. Joshi, and Ikeuchi K. A hand-held photometric stereo camera for 3-d modeling. ICCV, 2009.

[HZ03] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2nd edition, 2003.

[IKL+08] I. Ihrke, K. N. Kutulakos, H. P. A. Lensch, M. Magnor, and W. Heidrich. State of the art in transparent and specular object reconstruction. In *STAR Proceedings of Eurographics*, 2008.

[ISM84] S. Inokuchi, K. Sato, and F. Matsuda. Range Imaging System for 3-D Object Recognition. ICPR, 1984.

[KB05] V. Kolmogorov and Y. Boykov. What metrics can be approximated by geo-cuts, or global optimization of length/area and flux. ICCV, 2005.

[KBC06] K. Kolev, T. Brox, and D. Cremers. Robust variational segmentation of 3D objects from multiple views. In *Pattern Recognition (Proc. DAGM)*, September 2006.

[KBH06] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Eurographics symposium on Geometry processing*, SGP, 2006.

[KC08] K. Kolev and D. Cremers. Integration of multiview stereo and silhouettes via convex functionals on convex domains. ECCV, October 2008.

[KKB+07] K. Kolev, M. Klodt, T. Brox, S. Esedoglu, and D. Cremers. Continuous global optimization in multiview 3d reconstruction. EMMCVPR, August 2007.

[KKDH07] M. Kazhdan, A. Klein, K. Dalal, and H. Hoppe. Unconstrained isosurface extraction on arbitrary octrees. SGP, 2007.

[Kol12] K. Kolev. *Convexity in Image-Based 3D Surface Reconstruction*. PhD thesis, Department of Computer Science, Technical University of Munich, Germany, January 2012.

[Kov05] P. Kovesi. Shapelets correlated with surface normals produce surfaces. ICCV, 2005.

[Koz97] R. Kozera. Uniqueness in shape from shading revisited. *J. Math. Imaging Vis.*, March 1997.

[KS00] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. J. Comput. Vision*, July 2000.

[KSK+08] M. Klodt, T. Schoenemann, K. Kolev, M. Schikora, and D. Cremers. An experimental comparison of discrete and continuous shape optimization methods. ECCV, 2008.

[KZ02] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. ECCV, 2002.

[Lau94] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE TPAMI*, February 1994.

[LB07] V. Lempitsky and Y. Boykov. Global optimization for shape fitting. CVPR, 2007.

[LC87] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, August 1987.

[LQ05] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE TPAMI*, March 2005.

[LT09] D. Lanman and G. Taubin. Build your own 3d scanner: 3d photograhy for beginners. In *SIGGRAPH '09: ACM SIGGRAPH 2009 courses*, New York, NY, USA, 2009.

[MOC+98] R. A. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissanov. Structured light using pseudorandom codes. *IEEE TPAMI*, March 1998.

[MPS08] J. Manson, G. Petrova, and S. Schaefer. Streaming surface reconstruction using wavelets. *Computer Graphics Forum (Proceedings of the Symposium on Geometry Processing)*, 2008.

[MS89] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 1989.

[NEC06] M. Nikolova, S. Esedoglu, and T. F. Chan. Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models. *SIAM Journal on Applied Mathematics*, 2006.

[NLD11] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. ICCV, 2011.

[NN94] S. K. Nayar and Y. Nakagawa. Shape from focus. *IEEE TPAMI*, August 1994.

[NWR08] D. Nehab, T. Weyrich, and S. Rusinkiewicz. Dense 3D reconstruction from specularity consistency. CVPR, June 2008.

[OBA+03] Yutaka Ohtake, Alexander Belyaev, Marc Alexa, Greg Turk, Hans-Peter Seidel, and Mpi Saarbruecken. Multi-level partition of unity implicits. *ACM Transactions on Graphics*, 22, 2003.

[OS88] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, November 1988.

[PA82] J. L. Posdamer and M. D. Altschuler. Surface Measurement by Space Encoded Projected Beam System. *CGIP*, 1982.

[RB06] S. Roth and M. J. Black. Specular flow and the recovery of surface structure. CVPR, 2006.

[ROF92] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 1992.

[SBM98] J. Salvi, J. Batlle, and E. Mouaddib. A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Recogn. Lett.*, September 1998.

[SCS90] T. Simchony, R. Chellappa, and M. Shao. Direct analytical methods for solving poisson equations in computer vision problems. *IEEE TPAMI*, May 1990.

[SD97] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. CVPR, 1997.

[Sna09] K. N. Snavely. *Scene reconstruction and visualization from internet photo collections*. PhD thesis, Seattle, WA, USA, 2009.

[SPB04] J. Salvi, J. Pags, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 2004.

[SW04] Scott Schaefer and Joe Warren. Dual marching cubes: Primal contouring of dual grids. PG, 2004.

[SWN88] A. C. Sanderson, L. E. Weiss, and S. K. Nyar. Structured highlight inspection of specular surfaces. *IEEE TPAMI*, January 1988.

[TI90] J. Tajima and M. Iwakawa. 3-d data acquisition by rainbow range finder. ICPR, 1990.

[TLGS05] M. Tarini, H. P. A. Lensch, M. Goesele, and H.-P. Seidel. 3d acquisition of mirroring objects using striped patterns. *Graph. Models*, July 2005.

[VETC07] G. Vogiatzis, C. H. Esteban, P. H. S. Torr, and R. Cipolla. Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency. *IEEE TPAMI*, December 2007.

[WGS+11] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust photometric stereo via low-rank matrix completion and recovery. ACCV, 2011.

[WLDW11] C. Wu, Y. Liu, Q. Dai, and B. Wilburn. Fusing multiview and photometric stereo for 3d reconstruction under uncalibrated illumination. *IEEE TVCG*, August 2011.

[Woo89] R. J. Woodham. Shape from shading. Cambridge, MA, USA, 1989.

[WRO+12] M. Weinmann, R. Ruiters, A. Osep, C. Schwartz, and R. Klein. Fusing structured light consistency and helmholtz normals for 3d reconstruction. BMVC, September 2012.

[WWMT11] C. Wu, B. Wilburn, Y. Matsushita, and C. Theobalt. High-quality shape from multi-view stereo and shading under general illumination. CVPR, 2011.

[YAC06] T. Yu, N. Ahuja, and W. Chen. Sdg cut: 3d reconstruction of non-lambertian objects using graph cuts on surface distance grid. CVPR, 2006.

[YBT10] J. Yuan, E. Bae, and X. Tai. A study on continuous max-flow and min-cut approaches. CVPR, 2010.

[YPW03] R. Yang, M. Pollefeys, and G. Welch. Dealing with textureless regions and specular highlights-a progressive space carving scheme using a novel photo-consistency measure. ICCV, 2003.

[ZBK02] T. E. Zickler, P. N. Belhumeur, and D. J. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. *IJCV*, September 2002.