UNIVERSITY of CALIFORNIA
Santa Barbara

Detection and Modeling of Depth Discontinuities with Lighting and Viewpoint Variation

A dissertation submitted in partial satisfaction of the
requirements for the degree

Doctor of Philosophy
in
Computer Science

by

Rogério Schmidt Feris

Committee in charge:

Prof. Matthew Turk, Chair
Dr. Ramesh Raskar
Prof. Tobias Höllerer
Prof. Yuan-Fang Wang
Prof. Steve Seitz

September 2006

The dissertation of Rogério Schmidt Feris is approved.

Dr. Ramesh Raskar

Prof. Tobias Höllerer

Prof. Yuan-Fang Wang

Prof. Steve Seitz

Prof. Matthew Turk, Committee Chair

September 2006

Detection and Modeling of Depth Discontinuities with

Lighting and Viewpoint Variation

by

Rogério Schmidt Feris

To my parents and my sister, whom I love,

respect, and admire.

Acknowledgements

I am especially grateful to my advisor Matthew Turk, for his guidance and financial support throughout my graduate studies. His wisdom, insightful advices, criticism, and the freedom he gave me to explore my own ideas had a significant influence in my work. More than a mentor, he was a great friend, and provided me with an excellent research environment.

I wish to thank Ramesh Raskar for his invaluable mentorship. He led me to the subject of this dissertation, which is built upon his initial work on multi-flash imaging. I am very fortunate to have had the opportunity to work with him and very thankful for so many discussions and research advices on my thesis topic. I would also like to thank the other members of my committee: Steve Seitz, for his insights and knowledgeable critique; Tobias Höllerer, for discussions and for his teaching, from which I learned so much; Yuan-Fang Wang for insights and for sharing with me his experience and strong background in computer vision.

During my PhD work, I had many fruitful collaborations: special thanks goes to Longbin Chen, who helped me with experiments on global stereo based on belief propagation, as well as with the SVM classifier for detection of colored shadows. Karhan Tan was a great co-worker at MERL, and shared with me the madness of a SIGGRAPH deadline. Gosuke Ohashi kindly sent me his shape descriptor code that I used in my experiments for fingerspelling recognition. In a different research project related to facial image analysis, I want to thank Changbo Hu for his friendship and the long hours working together in the lab.

My sincere gratitude goes to my friends Gabriel Gomes, Marco Zuliani, Dimitry Fedorov, Marcelo Davanco, Laili Zandieh, Moises Ribeiro, Daniela Ushizima, Mylene

Farias, Marcelo Carvalho, and many others whom I shared a great time in Santa Barbara. Thanks to my labmates Ya Chang, Haiying Guan, and Mathias Kolsch for many research discussions and all the Four Eyes lab people for their friendship and for providing a nice work atmosphere. Also, I want to thank the staff of the UCSB computer science department, for helping me in innumerable ways.

I am deeply grateful to my fiancee Eliane Dutra, for her love, support, patience, endless encouragement, and for making my life truly enjoyable during these years. Most of all, I want to thank my parents Ari and Elisabeth, and my sister Alessandra, for their love and special presence in my life. Finally, I thank God for making all of this possible.

Curriculum Vitæ

Rogério Schmidt Feris

Education

2006	Ph.D. in Computer Science (expected), University of California, Santa Barbara, USA.
2001	Master of Science in Computer Science, University of Sao Paulo (USP), Brazil.
1998	Bachelor of Science in Computer Engineering, University of Rio Grande (FURG), Brazil.

Experience

2005	Research intern, IBM T.J. Watson Research Center, Hawthorne, New York.
2003	Research intern, Mitsubishi Electric Research Labs (MERL), Cambridge, Massachusetts.
2001	Research intern, Microsoft Research, San Francisco, California.

Selected Publications

- R. Feris, R. Raskar and M. Turk. Dealing with Multi-scale Depth Changes and Motion in Depth Edge Detection. *Proceedings of SIBGRAPI'06 Brazilian Symposium on Computer Graphics and Image Processing*, Manaus, Brazil, October 2006.

- R. Feris, L. Chen, M. Turk, R. Raskar and K. Tan. Discontinuity Preserving Stereo with Small Baseline Multi-Flash Illumination. *International Conference on Computer Vision (***ICCV 2005***)* – oral presentation, Beijing, China, 2005.

- R. Feris, M. Turk, R. Raskar, K. Tan and G. Ohashi. Recognition of Isolated Fingerspelling Gestures Using Depth Edges. *B. Kisacanin, V. Pavlovic and T. Huang (eds.), Real-time Vision for Human-Computer Interaction*, Springer-Verlag, 2005.

- K. Tan, R. Feris, R. Raskar, J. Kobler, J. Yu and M. Turk. Harnessing Real-World Depth Edges with Multi-Flash Imaging. IEEE Computer Graphics and Applications (**IEEE CG&A**), vol. 25, no. 1, pp. 32-38, January 2005.

- R. Raskar, K. Tan, R. Feris, J. Yu and M. Turk. Non-photorealistic Camera: Depth Edge Detection and Stylized Rendering using Multi-Flash Imaging. *ACM Transactions on Graphics (***SIGGRAPH 2004***)*, Vol. 23, Issue 3, August 2004. Also accepted in SIGGRAPH Emergent Technologies, 2004.

- R. Feris, R. Raskar, K. Tan and M. Turk. Specular Reflection Reduction with Multi-Flash Imaging. *Proceedings of SIBGRAPI'04 Brazilian Symposium on Computer Graphics and Image Processing*, Curitiba, Brazil, October 2004 – also accepted as a poster in SIGGRAPH 2004.

- R. Feris, M. Turk, R. Raskar, K. Tan and G. Ohashi. Exploiting Depth Discontinuities for Vision-based Fingerspelling Recognition. *IEEE Workshop on Real-Time Vision for Human-Computer Interaction (in conjunction with CVPR 2004)*, Washington DC, USA, June 2004.

- K. Tan, J. Kobler, R. Feris, P. Dietz and R. Raskar. Shape Enhanced Surgical Visualizations and Medical Illustrations with Multi-flash Imaging. *International Conference on Medical Imaging Computing and Computer Assisted Intervention (MICCAI 2004)*, Rennes, France 2004.

Abstract

Detection and Modeling of Depth Discontinuities with
Lighting and Viewpoint Variation

by

Rogério Schmidt Feris

Discontinuity modeling and detection has a long history in the field of computer vision, but most methods are of limited use because either they deal with intensity edges, which may not be informative regarding intrinsic object properties, or they attempt to detect discontinuities from noisy dense maps such as stereo or motion, which are particularly error-prone near discontinuities in depth (also known as depth edges or occluding contours).

We propose to systematically vary imaging parameters (in particular illumination and viewpoint) in order to detect and analyze depth discontinuities in real-world scenes. We build on promising preliminary research on multi-flash imaging [85], which uses small baseline active illumination to label depth edges in images. We show that by varying illumination parameters (such as the spatial position, number, type, and wavelength of light sources), we are able to handle fundamental problems in depth edge detection, including multi-scale depth changes, specularities and motion.

By combining active illumination with viewpoint variation, we provide a framework for robust depth-edge preserving stereo. We propose novel feature maps based on qualitative depth and occlusion analysis, which are useful priors for stereo. Based on these feature maps, we demonstrate enhanced local and global stereo algorithms

which produce accurate results near depth discontinuities.

Finally, we show the usefulness of our techniques in non-photorealistic rendering, with applications in comprehensible rendering, medical imaging and human facial illustrations. We also demonstrate the importance of depth contours in visual recognition, showing improved results on the problem of fingerspelling recognition.

Contents

List of Figures

Chapter 1

Introduction

Many computer vision tasks rely on the use of intensity edges as low-level features. These edge-based methods, however, are limited in their ability to reveal scene structure: many sharp intensity transitions are produced by texture or illumination variations that are not informative regarding object shape or boundaries. In addition, important gray level discontinuities along occlusion boundaries may have low contrast or appear blurred due to the imaging process. As a result, intensity edge maps often include undesirable edges (due to albedo changes, specularities or shadows) or omit key edges along important shape boundaries.

Ideally, we would like to detect and analyze discontinuities in the physical surfaces rather than (or in addition to) edges in the image intensities. The latter are somewhat arbitrary and do not always correspond to physical properties of objects; the former are well-defined and less dependent of the specific imaging conditions [37]. For example, abrupt changes in depth, motion, and surface normal are often directly related to the 3D scene geometry and can provide extremely important low-level features for image

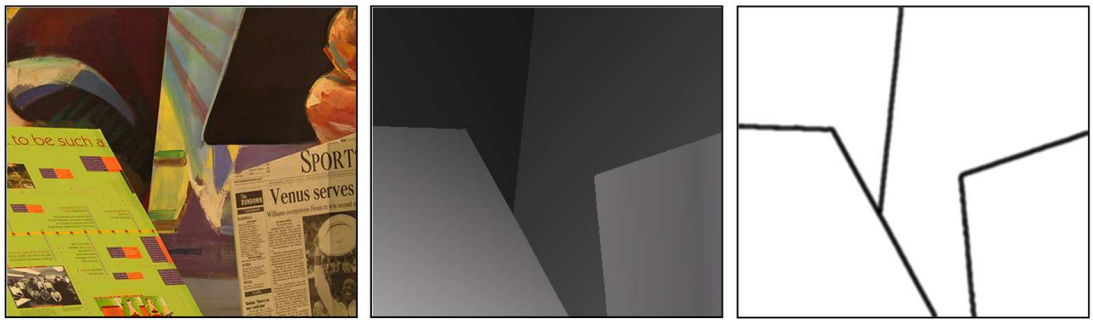
understanding, since they tend to outline the boundaries of objects in the scene.

Discontinuities can be classified according to their physical nature into discontinuities in depth, surface normal, illumination (e.g., boundaries of shadows or specularities), reflectance (material transitions), and motion. This dissertation addresses the particular problem of detection and analysis of discontinuities in depth.

1.1 Depth Discontinuities

Depth discontinuities, also known as depth edges or occluding contours, correspond to sharp changes (C0 discontinuities) in a depth map of the scene. A depth map consists in a two dimensional array which stores the depth (*z values*) of each surface point correspondent to a pixel in the image. Figure 1.1 shows an input photo, the correspondent depth map (where closer points are lighter in intensity), and the depth edge map. We will show later that depth edges can be obtained directly, bypassing the depth map computation.

The imaging geometry is illustrated in Figure 1.2. Depth discontinuities arise when a light ray associated with an image pixel meets a discrete change in the depth of the surfaces in the world [11]. This occurs when the light ray is tangent to the surface, i.e., a depth edge pixel is associated to a surface point whose normal is perpendicular to the light ray. Note that depth edges are view-dependent – depending on the camera viewpoint, surface points may correspond to depth edges or not. For example, in Figure 1.2, the surface point P corresponds to a depth edge in camera A, but not in camera B.



(a) (b) (c)

Figure 1.1: *(a) Original Photo. (b) Depth map. (c) Depth edges correspond to sharp C0 discontinuities in the depth map.*

1.1.1 Applications

Reliable detection of depth edges clearly facilitates segmentation, establishes depth-order relations, and provides valuable features for visual recognition [31], tracking, and 3D reconstruction [19]. Since depth edges tend to coincide with object boundaries, they can also be used for many graphics applications, including matting, synthetic aperture photography [49], and non-photorealistic rendering [85]. Figure 1.3 shows the usefulness of a depth edge map by comparing it with intensity edges detected with Canny operator [16]. Note that the representation based on depth edges contains important shape boundaries directly tied to the 3D scene geometry. Background clutter due to reflectance discontinuities is completely removed.

David Marr [66], in his computational vision model, emphasizes the importance of depth discontinuities as a representation for scene geometry in early vision. Compared to dense depth maps, depth edges provide an enormous reduction in the amount of data, while preserving the salient features. This reduction, in contrast to intensity edges, is done in a purely geometric way. Depth edges are completely independent of changes in lighting or object material properties.

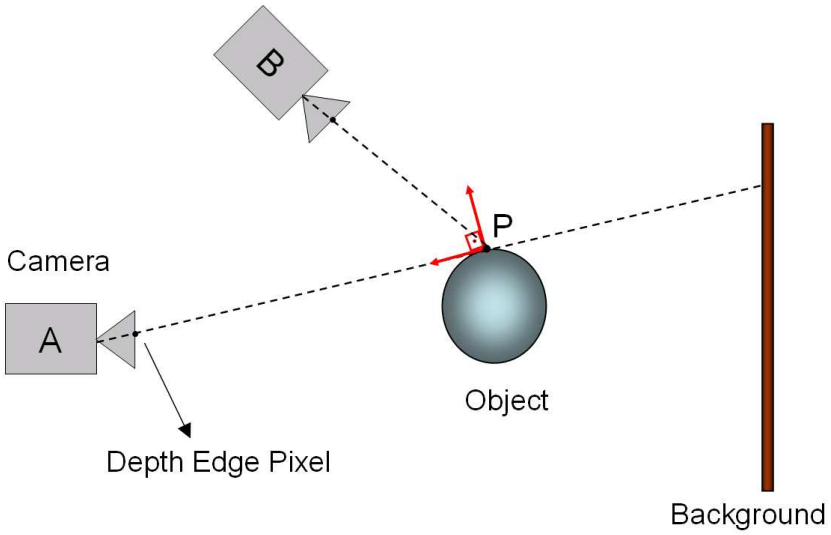


Figure 1.2: *Depth discontinuities arise when a light ray associated with an image pixel meets a surface point whose normal is perpendicular to the light ray. They are view-dependent: the surface point P corresponds to a depth edge in camera A, but not in camera B.*

The importance of depth discontinuities can also be seen in its connection with partial occlusions in stereo and motion algorithms. The detection of occluded pixels is extremely useful for estimation of dense depth and motion fields [53, 13].

1.2 Problem and Approach

A natural way to detect depth discontinuities is to first compute the dense depth map of the scene and then look for discontinuities in this data. However, the majority of 3D reconstruction methods produce inaccurate results near depth discontinuities, due to occlusions and the violation of smoothness constraints. As a result, most previous approaches proposed for detection of depth discontinuities treat them as an annoyance, rather than as a positive source of information [11, 46].

Recently, steady progress has been made in discontinuity preserving stereo match-

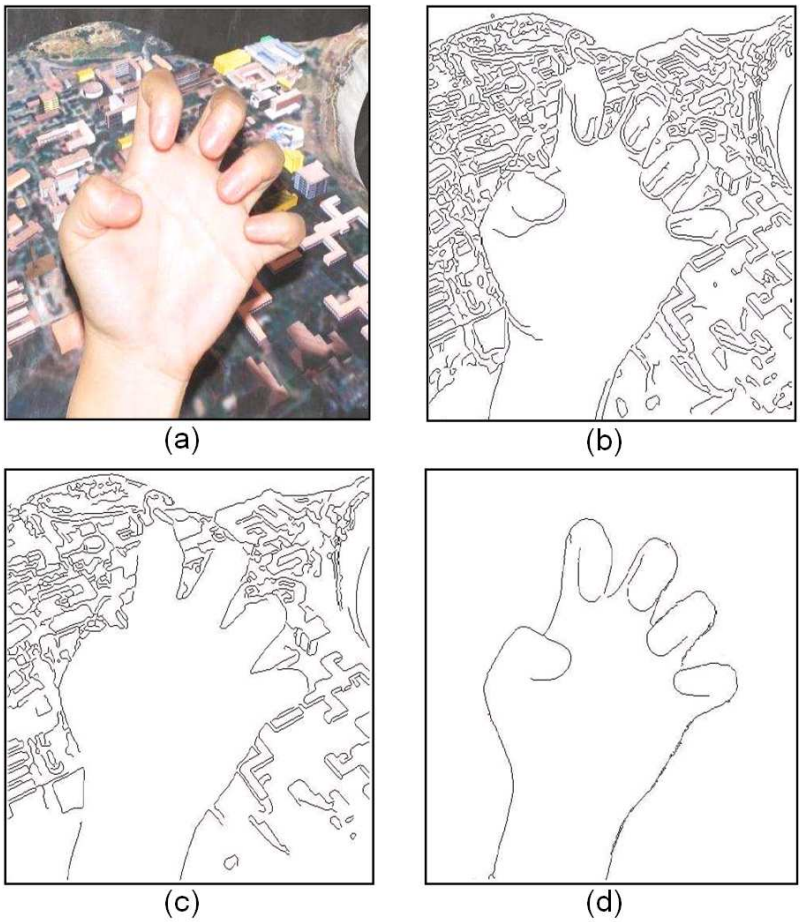


Figure 1.3: *(a) Original image with a planar cluttered background. (b) Canny intensity edges with* $\sigma = 0.3$*. (c) Canny intensity edges with* $\sigma = 1.0$*. (d) Depth edges. Note that depth edges are directly tied to the 3D geometry of the hand.*

ing [5], mainly with global optimization algorithms based on belief propagation or graph cuts (see [111] for a comparison). However, these methods fail to capture depth edges associated with sufficiently small changes in depth. Moreover, obtaining clean, non-jagged contours along shape boundaries is still a challenging problem even for methods that rely on more expensive hardware [39].

We propose to systematically vary imaging parameters (specifically illumination and viewpoint) in order to detect and analyze depth edges in real-world scenes. Re-

cently, multi-flash imaging [85] has been proposed to bypass 3D reconstruction and reliably detect depth discontinuities directly. Images are taken under different illumination conditions and the resulting cast shadows are exploited to robustly separate shape contours from reflectance discontinuities. We extend this work in multiple ways, by intentionally varying parameters of the imaging process and analyzing the resulting images.

1.3 Thesis Statement

This dissertation introduces novel methods based on the variation of imaging parameters (in particular illumination and viewpoint) to detect and analyse depth discontinuities in real-world scenes. It demonstrates the usefulness of computed depth edges as low-level features for scene understanding tasks, including stereo and visual recognition. In a broader perspective, it motivates new imaging technologies for general scene analysis based on physical surface discontinuities, contrasting with current holistic or edge-based computer vision approaches.

1.4 Contributions

Our main contributions are listed below:

- We improve and extend preliminary work on multi-flash imaging, generalizing the idea by varying illumination parameters (such as the number, spatial position, type, and wavelength of light sources) in order to handle fundamental problems in depth edge detection. As part of this framework, we propose the following:

  - A multi-baseline approach for detecting depth edges under multi-scale depth changes.
  - A gradient-domain method for handling specular reflections in depth edge detection.
  - A technique that exploits flashes with different wavelength to detect depth edges in motion.

- We integrate viewpoint variation with small baseline illumination to obtain high quality disparity maps near depth discontinuities. In particular, we propose novel feature maps based on qualitative depth and occlusion analysis, showing their usefulness in dense stereo matching.

- We demonstrate the importance of automatically computed depth edges in visual recognition. A novel method for recognizing fingerspelling gestures based on depth contours is proposed, which shows superior performance over intensity edge-based techniques.

1.5 Outline

This dissertation is organized as follows: in Chapter 2, we discuss related work, covering methods that are directly related to our proposed techniques.

In Chapter 3, we describe our basic algorithm for depth edge detection and then show how to improve it in a wider variety of imaging conditions, by intentionally varying illumination parameters, such as the spatial position, number, type, and wavelength of light sources. We highlight novel algorithms to handle multi-scale depth changes,

specular reflections, and motion.

In Chapter 4, we combine lighting with viewpoint variation to achieve high quality, discontinuity preserving disparity maps. We propose novel feature maps based on qualitative depth and occlusion analysis, which are useful priors for stereo. Based on these feature maps, we demonstrate enhanced local and global stereo algorithms which produce accurate results near depth discontinuities.

Chapter 5 shows the usefulness of depth discontinuities and multi-flash imaging in comprehensible rendering, medical imaging and human facial illustrations. In Chapter 6, we demonstrate the importance of depth contours in visual recognition, showing improved results on the problem of fingerspelling recognition. Finally, Chapter 7 provides general conclusions and motivates ideas for future work.

Chapter 2

Background

This chapter reviews the background and some earlier work related to this dissertation. We first discuss existing approaches to classify discontinuities according to their physical nature, giving special attention to the detection of depth discontinuities. Then, we discuss 3D reconstruction techniques that are directly related to the methods that we will present in the following chapters, including dense stereo, photometric, and shadow-based approaches. Finally, we cover existing flash-based techniques and methods that exploit the variation of other camera and scene parameters.

2.1 Discontinuities in Computer Vision

A good deal of attention has been given to detecting and representing discontinuities for scene understanding in the field of computer vision, especially during the 1980s [37, 62, 66, 113]. Much of this research was motivated by the seminal work of Marr [66], who articulated the importance of discontinuities in his computational

vision model. Marr's primal sketch included both depth and normal discontinuities as part of the general model.

Most of the methods proposed over the years to detect and represent image discontinuities have dealt with intensity edges [16], although some attempts were made to classify discontinuities according to their physical origin as well [37, 15, 38]. In the early 1990s, attention was re-directed to appearance-based approaches due to their success, particularly in object recognition. Although physically-based edge classification is still an active research problem, existing methods do not seem capable of providing a general solution.

Gamble and Poggio [37] proposed a scheme based on coupled Markov Random Fields, which integrates intensity edges with stereo depth and motion field information, in order to find depth and motion discontinuities. They claim that this scheme could be generalized to classify other discontinuities according to their physical nature. Relying on techniques that estimate a dense field (such as stereo and shape from shading) makes this difficult, due to the fact that such methods are noisy, especially at depth discontinuities.

Boult and Wolff [15] proposed to use polarization to distinguish occluding edges, specular edges, and albedo edges, assuming smooth dielectric surfaces. A linear polarizer is placed in front of the camera sensor and multiple pictures of the scene are taken with the polarizer at different orientations. The polarization properties of occluding, specular, and albedo discontinuities is exploited for edge classification. The method fails for rough surfaces or when the direction of the light source (which is assumed to be unpolarized) approximately aligns with the surface edge orientation. Moreover, many images (about twenty) need to be captured to achieve reasonable results.

Gevers [38] exploited color information to classify edges according to their physical origin. Different color spaces are proposed which show some invariance to object geometry, specular highlights, or shadows. Color edge detection is applied on these color spaces and a set of rules is used to classify edges into shadow or geometry edges, highlight edges, and material edges. Although the system shows some success to detect surface normal and illumination discontinuities, it is not capable of discriminating depth edges from material transitions.

Separating illumination discontinuities from albedo changes has been addressed as the problem of computing intrinsic images [112, 117]. Tappen et al. [112] use color information and a classifier trained to recognize gray-scale patterns in order to classify image derivatives as being caused by reflectance or illumination changes. The Retinex algorithm [61] also deals with the same problem, relying on the assumption that the gradients along albedo changes have a sharp transition compared to the gradual transition in illumination discontinuities.

Numerous methods have attempted to detect motion discontinuities in optical flow fields by analyzing local distributions of flow or by performing edge detection on the flow field [101, 114]. It has often been noted that these methods are sensitive to the accuracy of the optical flow and that accurate optical flow is hard to estimate without prior knowledge of the occlusion boundaries. Black and Fleet [13] proposed a Bayesian framework to detect and track motion discontinuities probabilistically. Their generative model allows explicitly modeling of which image pixels are occluded or disoccluded between frames, which was not possible with previous approaches. Birchfield [11] deals with the problem of motion discontinuity detection by tracking a sparse set of features throughout the sequence, grouping them according to an affine motion model,

and tracing the boundaries among the groups.

2.1.1 Detection of Depth Discontinuities

Many methods have been proposed to deal with depth discontinuities in 3D reconstruction, mainly in stereo matching. Standard methods detect depth edges by post-processing, i.e., by finding discontinuities in a previously computed depth map [37]. Recent techniques have been proposed to estimate depth edges simultaneously with the full correspondence map in stereo [12, 50, 5]. In the next section we will review these techniques in the context of dense stereo matching. Here, we focus on methods that estimate depth discontinuities directly, without dense reconstruction.

Few attempts have been made to detect depth edges without computing dense scene reconstruction or disparity maps. Most of the proposed techniques rely on finding T-junctions from a single image [7] or detecting partial occlussions to infer depth discontinuities from a stereo pair [26].

T-junctions correspond to photometric profiles shaped like a “T”, which is formed where the edge of an object occludes a change in intensity in the background (Figure 2.1). There have been two predominant approaches to detect T-junctions: gradient or filter-based approaches [35], which exploits the local properties of the image gradient near junctions, and model-based approaches [75], which approach the problem by fitting an explicit junction model at hypothesized junction locations. In a more recent work, Apostoloff and Fitzgibbon [7] learn the appearance of T-junctions in video sequences, by analyzing spatiotemporal volumes. They use intensity edges to interpolate T-junctions in order to find occluding contours. Since many T-junctions may also occur in texture edges, and given that many depth edge pixels are not associated with

Figure 2.1: *T-junctions as cues for depth discontinuities (from Birchfield [11]). Left: Original image with a white box highlighting a T-junction. Right: Manually-drawn depth contour with corresponding T-junctions.*

T-junctions, these techniques are fairly limited to detect depth discontinuities.

Using local detectors to find partial occlusions in an image pair is another approach to infer depth discontinuities directly. Partial occlusion points are those that are visible in only one of the two views provided by the binocular imaging system. Every occlusion is associated with a depth discontinuity and therefore can be used as a cue to detect depth edges.

Wixson [119] proposed an algorithm based on partial occlusion detection that establishes correspondence between intensity edges in the stereo pair. The left and right regions of the edges are examined, and if one of them has a low correspondence score, a depth discontinuity is declared. We refer to the recent survey conducted by Egnal and Wildes [26] for other partial occlusion detector approaches. Depth edge detection based on these techniques has two key limitations: 1) it fails in textureless regions, as the stereo pair needs to contain suficient texture to allow correspondence and 2) although every occlusion is associated with a depth discontinuity, not every depth discontinuity is associated with occlusion. For example, these methods are not able to

detect depth edges that lie along boundaries parallel to the stereo camera baseline, because these discontinuities do not give rise to occlusions.

Huggins et al. [46] analyze the appearance of occluding contours or folds under variable illumination, and find that the pattern of shading near depth discontinuities is a stable feature. Based on this analysis, they derive a filter to identify occluding edges in images. This method assumes that the surface is locally smooth, which fails for a flat foreground object like a leaf or piece of paper. They detect regions near occluding contours but not the contours themselves.

Multi-flash imaging has been recently proposed by Raskar et al. [85, 86] to extract depth edges from complex scenes and automatically create stylized images. To our knowledge, this is the first active illumination method proposed to detect depth edges directly, with significant better results than previous passive approaches. We will describe the basic idea of this method in the next chapter and then explore multi-flash imaging in a more general way to detect depth edges under a wider variety of conditions.

2.2 Depth Recovery Techniques

Extensive research has been done to recover a depth map from images, but producing accurate results near depth discontinuities is still a challenging problem for most techniques. Multi-view triangulation approaches, such as stereo or structure from motion, suffer from the partial occlusion problem near discontinuities, requiring a point to be visible in at least two views to be reconstructed. Single-view photometric methods, including shape from shading and photometric stereo (which will be discussed next

section), estimate a field of surface normals, rather than a depth map. Surface normal integration to obtain depth information is an ill-posed problem for scenes with depth discontinuities [79].

Laser striping scanning methods [78] project a moving stripe onto the scene and estimate depth by triangulation. Methods based on multiple laser stripes projected simultaneously are also available, even in compact forms [103], but are in general designed for smooth objects. Temporal laser scanning and coded structured light [45] allow each point viewed by a camera to have a specific codeword, thus facilitating correspondence and depth estimation. These techniques will be discussed in Section 2.2.1 in the context of active stereo matching. Finally, methods based on time-of-flight [39] produce accurate depth maps from complex scenes, but are expensive and require specialized hardware. Next we discuss depth recovery methods based on dense stereo matching, which will be the topic of Chapter 4.

2.2.1 Dense Stereo Matching

Stereo techniques including passive and active illumination are generally designed to compute depth values rather than to detect depth edges. Depth discontinuities present difficulties for traditional stereo; it fails due to partial occlusions, which confuse the matching process [53, 12]. A comprehensive survey of passive stereo matching techniques was recently presented by Scharstein and Szeliski [93]. Next we will give emphasis to techniques that model depth discontinuities and occlusions explicitly. We consider only binocular stereo; the reader is referred to the survey presented by Seitz et al. [96] for multiple view stereo methods. We will also discuss active illumination approaches for stereo matching.

Passive Stereo

In general, dense stereo techniques can be classified as local or global, depending whether they rely on local window-based computations or the minimization of a global energy function. In local-based methods, the disparity computation at a given point depends only on intensity values within a finite window. Clearly, these techniques assume that all pixels within the window have the same disparity and thus are sensitive near object boundaries. Okutomi and Kanade [52] attempt to alleviate this problem by varying the window size at each pixel, and choosing the size that minimizes the disparity uncertainty. In similar work, Kang et al. use shiftable windows [53] for dealing with discontinuities in local stereo.

Occlusion has been modeled explicitly through global optimization approaches based on dynamic programming [9, 47, 12]. Stereo matching is formulated as finding a minimum cost path in the matrix of all pairwise matching costs between two corresponding scanlines. These techniques, however, often show a streaking effect (as scanlines are matched independently) and assume ordering constraints, which may be violated with thin objects in the scene.

More recently, global stereo approaches based on Markov Random Fields have received great attention [111, 93]. These methods minimize an energy function (using e.g., belief propagation [106] or graph cuts [56]) that includes a data term and a smoothness term. Although discontinuities and occlusion can be explicitly modeled [56, 104], intensity edges and junctions are generaly used as cues for depth discontinuities. Ideally, smoothness constraints should be supressed only at occluding edges, not at texture or illumination edges.

Active Stereo

The correspondence problem can be significantly simplified by using active illumination methods based on structured light [90, 94]. A structured light system is based on the projection of a single pattern or a set of patterns onto the scene which is then viewed by a single camera or a set of cameras. Since the pattern can be coded, correspondences between image points and points of the projected pattern can be easily found. In general, coded structured light methods can be classified in the following groups [90]: time-multiplexing, neighborhood codification, and direct codification.

Time-multiplexing or temporal coding [81, 45] is the most common pattern projection technique. The basic idea is to project a set of different patterns successively onto the scene, so that each point viewed by a camera has a specific codeword (formed by the sequence of illumination values accross the projected patterns). For binary codes, only two illumination levels are used, so that the codeword for each pixel is formed by a sequence of 0's and 1's, as shown in Figure 2.2. The main drawback of using binary codes is the large number of patterns which need to be projected to give each pixel a unique code. Techniques based on n-ary codes [45] handle this problem, usually projecting patterns with grey-level stripes. Detecting multiple illumination levels in the image makes the system more sensitive to noise, though.

In general, methods based on time-multiplexing allow accurate computation of correspondence maps, but are limited to handle dynamic scenes, since the captured images need to be registered. This issue is addressed by *neighborhood codification* techniques, which project a single pattern with pixel coding based on a spatial neigborhood [124]. However, the methods are limited in their ability to produce accurate results near depth discontinuities, as local smoothness of the measuring surface is assumed in order to

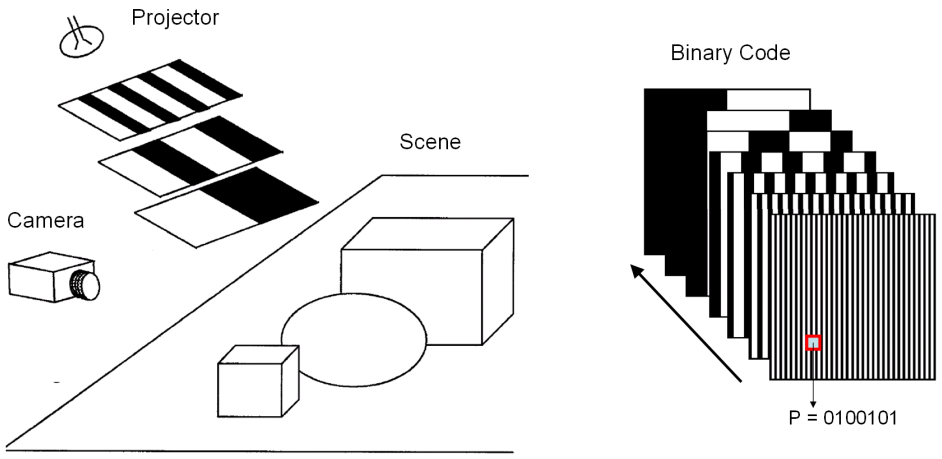


Figure 2.2: *Structured light methods based on temporal coding project multiple patterns successively onto the scene, so that each point viewed by a camera has a specific codeword.*

correctly decode the pixel neighborhood.

Direct codification techniques use projected patterns that allow the entire codeword to be contained in a unique pixel. In order to achieve this, it is necessary to use a large range of colour values in the pattern or introduce periodicity. As an example, Tajima and Iwakawa [108] use a color-based, rainbow pattern for pixel coding and triangulation. The perceived colors depend not only on the projected color, but also on the intrinsic color of the measuring surface. This means that in most cases one or more reference images must be captured. Direct codification methods are useful for achieving large spatial resolution with few projecting patterns, but are quite sensitive to camera sensor noise.

Space-time stereo [126, 22] was proposed as a unifying framework for passive and active stereo techniques. Passive stereo typically identifies features purely in the spatial domain, i.e., correspondence is found by determining similarity of pixels in the image plane. Coded structured light makes use of features which lie predominantly in the

temporal domain. That is, features with similar appearance over time are likely to correspond. Space-time stereo identifies corresponding features in both the space and time domains.

Overall, compared to passive stereo techniques, structured light methods offer high quality correspondence maps and 3D acquisition, but are in general much more expensive and limited to indoor scenes.

2.3 Photometric Techniques

Many photometric methods for 3D reconstruction such as shape from shading and photometric stereo have been proposed over the last decades. Differently from the stereo techniques presented in the last section, these methods avoid the correspondence problem and estimate surface normals, rather than depth, by capturing images with the same viewpoint, but under different lighting conditions.

Shape from shading [44] attempts to estimate a field of surface normals (which may be integrated to produce a surface) from a single image captured with a known light source position. The idea relies on the fact that the intensity of each pixel in the image depends on the light position, the local surface normal, and the surface reflectance properties. The technique requires a number of restrictions. The illumination must come from a single light source which is either collimated or distant enough to approximate a collimated light source. Also, surface reflectance is assumed to be Lambertian with uniform albedo. Even with these restrictions, the problem is still ill-posed - the intensity at a particular pixel provides only one constraint, whereas the correspondent surface normal is specified by two parameters. As a result, most shape from shading

methods apply another constraint, tipically the smoothness of the reconstruction, which causes problems at occluding boundaries.

Photometric stereo was introduced by Woodham [121] to overcome some of these problems. Rather than using a single image, multiple pictures of the scene are captured with the same viewpoint, but with light sources placed in different positions. By considering a Lambertian surface, and capturing at least three images with variable illumination, not only the field of surface normals can be estimated, but also the surface reflectance factor (albedo).

Hertzmann and Seitz [43] recently introduced a photometric stereo approach to handle objects with arbitrary and spatially varying BRDFs (Bidirectional Reflectance Distribution Functions). In fact most real-world objects reflect light in a wide range of different ways, violating the Lambertian scene assumption made in most previous work on photometric stereo. They use reference objects in the scene with similar materials and known geometry. Shape inference is based on the assumption that points with the same surface orientation must have the same or similar appearance in an image.

Photometric stereo methods are capable of reconstructing the shape of objects with uniform albedo, thus complementing conventional stereo, which is best for textured surfaces with varying reflectance. They work well for smooth surfaces, but are typically unstable around depth discontinuities [91]. Other sources of error in photometric stereo include camera sensor noise, innacurate calibration of light source directions, and artifacts such as shadows and specular highlights. A common limitation of existing methods is that the light sources need to surround the object in order to create sufficient shading variation from (estimated or known) 3D light positions. This requires a fixed lighting rig, which limits the application of these techniques to industrial

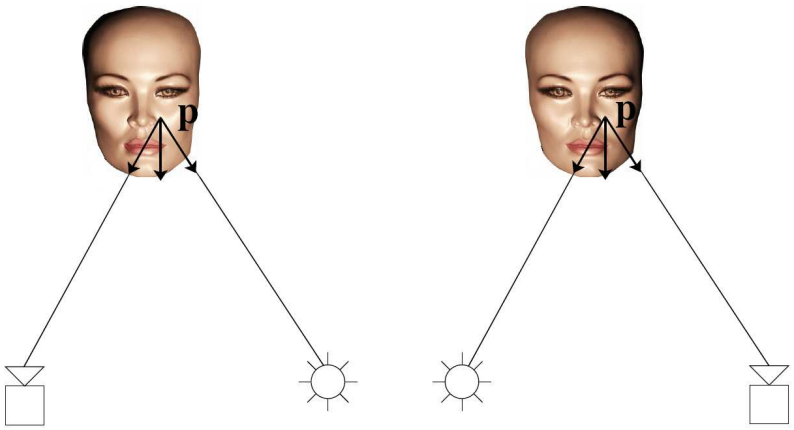


Figure 2.3: *Helmholtz Stereopsis (from Zickler et al. [127]). First an image is acquired with the scene illuminated by a single point source as shown on the left. Then, a second image is acquired after the positions of the camera and light source are exchanged as shown on the right.*

settings; such a setup is impractical to build into a self-contained camera.

Helmholtz stereo [127] combines active lighting with viewpoint variation to estimate both surface normals and depth with arbitrary surface reflectance. The idea behind Helmholtz stereopsis is to exploit the symmetry of surface reflectance, commonly referred to as Helmholtz reciprocity. The image acquisition proceeds in two simple steps: first, an image is acquired with the object/scene illuminated by a single point light source. Then, the positions of the camera and light source are exchanged, and the second image is acquired, as shown in Figure 2.3. By acquiring the images in this manner, they ensure that for all corresponding points in the images, the ratio of the outgoing radiance to the incident irradiance is the same. This is in general not true for stereo pairs - unless the surfaces have Lambertian reflectance.

The ability to estimate object geometry in scenes containing surfaces with arbitrary, spatially varying BRDFs (without requiring reference objects, as in [43]) is a very powerful feature of Helmholtz stereo. Specular highlights in reciprocal image

pairs essentially become features for correspondence, rather than a source of error. In general many reciprocal pairs need to be captured for obtaining accurate 3D reconstruction results. A binocular approach has been recently proposed [128], where the problem is formulated as solving a partial differential equation, but it works only for smooth surfaces. The authors in fact mention that shadows could be used as cues for detecting depth discontinuities and enhancing the approach. Another disadvantage is that the camera and light source must be calibrated and moved in a precise and controlled fashion.

2.4 Shadow-Based Methods

Although shadows are often treated as a source of noise in segmentation and photometric techniques, they carry valuable 3D information about surfaces in the scene. Techniques for 3D reconstruction based on shadows have the advantage that they do not rely on correspondences, on a model of the surface reflectance characteristics, and they may be implemented using inexpensive lighting and/or imaging equipment [92].

Shape from shadows [54], or shape from darkness, is commonly referred to describe surface reconstruction methods based on shadows. The idea is to capture a sequence of images of the scene from the same viewpoint under different lighting conditions and then use only the shadow information to reconstruct the surface. Most methods assume that the surface is terrain-like, i.e., it can be described as a continuous function and rests on a reference plane, which is at a known distance from the viewer. The light source position needs to be known and both the camera and light are assumed to be at a sufficiently large distance from the scene, so that the asumption of ortographic

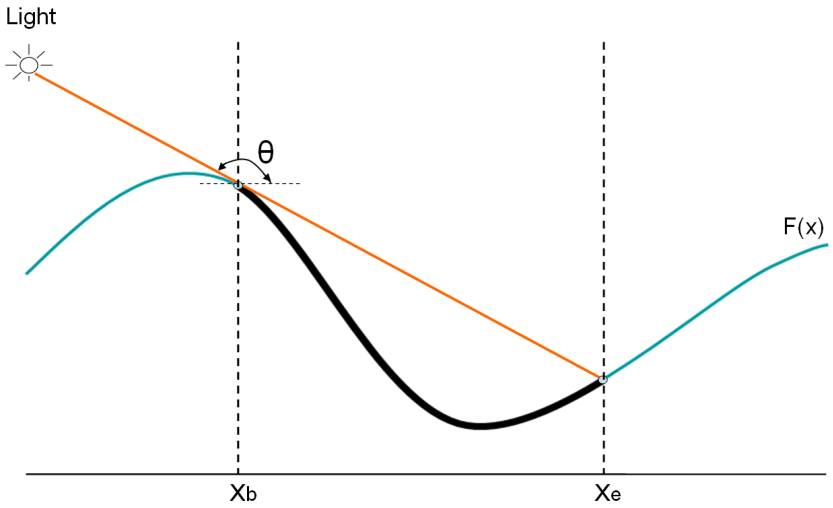


Figure 2.4: *The goal of shape from shadows is to estimate a surface slice $f(x)$ using the shadow information from multiple images. We have that $f'(x_b) = tan\theta$ and $f(x_b) - f(x_e) = f'(x_b)(x_e - x_b)$. Using data for many angles θ, an estimate of the continuous function $f(x)$ can be made.*

projection and parallel light rays holds. Also, existing methods assume that the start and end of the shadows can be found reliably in the images.

Different techniques have been proposed to estimate shape from shadows in computer vision. Hatzitheodorou and Kender [42] presented a method to recover a cross section or surface slice from a set of images with moving shadows. Assuming that the slice contour is defined by a smooth function - and that the beginning and end of each shadow region can be found reliably – each pair of points bounding a shadow region yields an estimate of the contour slope at the start of the shadow region, as well as the difference in height between the two points, as shown in Figure 2.4. This shadow information from multiple light source positions is then used to obtain an interpolating spline that is consistent with all the observed data points.

Raviv et al. [87] developed a shape from shadows method based on a representation called *shadowgram*. The object is set on a known reference plane with a camera

directly above. A dense set of images is captured as the light source continuously moves in an arc over the object. The shadowgram consists in a binary function $s(x, \theta)$, where x is the spatial dimension and θ is the angle of the light rays with the reference plane. For each position x and angle θ, $s(x, \theta)$ indicates whether or not the pixel x was in shadow when the light angle was θ. Surface reconstruction can be obtained by integrating this representation.

More recently, Daum and Dudeck [21] considered surface reconstruction for light trajectories that are not a single arc, allowing the 3D reconstruction of the entire scene, rather than just a cross section of the surface. Yu and Chang [123] combined shape from shadows with shape from shading, using a graph-based representation for encoding shadow constraints.

The above methods are sensitive to errors due to inaccurate light source position estimation and detection of the boundaries of shadow regions. Yang [122] considers the problem of shape from shadows with error. He presents a modified form of Hatzitheodorou and Kender approach, in which linear programming is used to eliminate inconsistencies in the shadow data used to estimate the surface. Kriegman and Belhumeur [58] analyze the shape from shadows problem with unknown light directions.

Shadow carving [92] was proposed by Savarese et al. as a technique to refine the 3D reconstruction of objects given an initial conservative estimate of the object shape. The idea is similar in spirit to the space carving approach of Kutulakos and Seitz [59]. In space carving, voxels are carved out of a region of space enclosing the object if they do not project consistently into a set of captured images. In shadow carving, the consistency is considered between a camera and light views, rather than multiple camera

views. Consistency can be tested robustly by detecting shadows, without requiring a Lambertian surface. In general the method can not create a fine detail object reconstruction, but yields good shape estimates, which can be used as start point for other 3D reconstruction techniques.

Bouguet and Perona [14] proposed a method for capturing 3D surfaces based on *weak structured lighting*. The user moves a pencil in front of the light source casting a moving shadow on the object. The 3D shape of the object is extracted from the spatial and temporal location of the observed shadow in the captured images. The use of a pencil to cast shadows on the object contrasts with previous shape from shadows methods that use self-shadows (i.e., shadows cast by the object upon itself). Compared with structured lighting techniques, this approach offers less accurate results, but a much simpler and inexpensive system, which can also be used outdoors considering the sun as the light source.

Shadows have also been used in the interpretation of aerial images [48] and for interactive applications. Segen and Kumar [95] describes a system which uses shadow information to track the user's hand in 3D. They demonstrated applications in object manipulation and computer games. Leibe et al. [64] presented the concept of a *perceptive workbench*, where shadows are exploited to estimate 3D hand position and pointing direction. Their method used infrared lighting and was demonstrated in augmented reality gaming and terrain navigation applications.

2.5 Computational Photography

Computational Photography is an emerging new field created by the convergence of computer graphics, computer vision and digital photography. This field encompasses computational methods and novel imaging techniques that are used to overcome the physical limitations of a camera, such as dynamic range, resolution and depth of field. [23, 98, 2].

A recent trend in computational photography research is the idea of taking multiple photographs of a scene with varying camera/scene parameters and combining them to synthesize a new image. Examples of this approach include creating high dynamic range images by combining photographs taken at different exposures [23], creating mosaics and panoramas by combining images taken from different viewpoints [107], synthetically relighting images by combining photographs taken under different illumination conditions [6], producing a high-resolution image from a set of low-resolution images [98], and capturing images with variable focus for synthetic aperture photography [2]. Many of our techniques presented in this dissertation belong to this framework. Another recent direction in computational photography is the use of flash-based methods, which are discussed next.

2.5.1 Flash Photography

Flash-based techniques have been widely adopted in the computer graphics community in recent years. Different approaches that take advantage of flash photography have been proposed, with applications in removal of noise and reflection artifacts, image matting, and 3D geometry acquisition, to mention just a few.

Petschnigg et al. [80] present different techniques to combine images captured with flash and without flash, in order to enhance photographs shot in dark environments. Ambient (no-flash) images offer the advantage of capturing the visual richness of the environment, but suffer from noise and blur. On the other hand, flash images are desirable to avoid noise and preserve fine details in the scene. However, objects near the camera are disproportionately brightened, and the mood evoked by ambient illumination may be destroyed. In addition, the flash may introduce unwanted artifacts such as red eye, harsh shadows, and specularities, none of which are part of the scene. In order to combine the advantages of flash and no-flash image pairs, the bilateral filter is used to decompose the two images into fine detail and large scale layers. An enhanced image is then generated by using the large scale layer of the ambient image and the fine detail of the flash image. The result preserves the ambience of the original lighting, while providing the detail and sharpness of the flash image, as shown in Figure 2.5. Shadows and specular reflections are detected in the flash image using simple heuristics to reduce artifacts. Other features are also presented, including interactive change of flash intensity, white balance and red eye removal. In concurrent work, Eisemann and Durand [27] developed similar techniques to enhance photos from flash and no-flash image pairs. The main difference is that the color of the output image is from the flash image, which is less noisy, but requires more work to correct for flash shadows.

Flash and no-flash image pairs are also used by Agrawal et al. [4] to remove artifacts such as glass reflection from photographs. The approach is based on the observation that the gradient orientations in the flash image are coherent with the gradient orientations in the ambient image, except at regions with flash or ambient artifacts. By using a gradient projection technique, the component of image gradients that are intro-

Figure 2.5: *Image enhancement using flash and no-flash image pairs (from Petschnigg et al. [80]). A flash image captures the high-frequency texture, but changes the overall scene appearance to cold and gray. The no-flash image captures the overall appearance of the warm candlelight, but is very noisy. The detail information from the flash image is used to both reduce noise in the no-flash image and sharpen its detail.*

duced by undesirable reflections can be eliminated to produce an enhanced reflection-free image. Interestingly, the eliminated gradient components can be further integrated to obtain the reflection layer. They also show techniques to compensate for the falloff in the flash image brightness due to depth and improve dynamic range, using several images taken under different flash intensities and exposures. More recently, Agrawal et al. [3] demonstrate edge suppresion techniques using images of the same scene under variable illumination. Among other tasks, they show how to remove ambient shadows by supressing edges in the ambient image that do not exist in the flash image.

Paterson et al. [76] uses flash photography for object 3D geometry and reflectance acquisition. The key components of their system consist of a standard digital camera with a single flash light and a calibration target with known fiducials, which is attached to the object. Multiple pictures of the object are then captured by moving the camera at different viewpoints. The known calibration target geometry is used to calculate the camera pose, light position, and a rectified view for each source image. Using

the set of rectfied images, photometric stereo is applied to estimate object reflectance and surface normals. The main problem with this approach is that the object needs to be approximately planar (e.g., a brick wall or a hand) so that the rectified images are properly aligned. It also fails for highly specular objects.

3D surface reconstruction based on flash photography has also been addressed recently by Crispell et al. [20]. They use multi-flash imaging [85, 86] to capture multiple images of an object in a turntable. The detection of depth discontinuities is then applied to improve shape from silhouette approaches [19, 67], producing better results at surface concavities. The reconstruction output tend to have gaps in areas of very low surface curvature, so a surface fitting algorithm is necessary to bridge the gaps.

Another flash photography application that has been exploited recently by Sun et al. is image matting [105]. The key observation here is that the most noticeable difference between a flash and no-flash image pair is the foreground object if the background scene is sufficiently distant. Based on this fact, they propose a Bayesian matting algorithm that is able to extract high quality mattes from complex backgrounds. The method fails when the scene contains objects with low albedo or close to the background.

2.6 Beyond Illumination and Viewpoint

Thus far we have discussed methods based on the variation of camera viewpoint and illumination parameters. We now briefly analyse techniques that exploit other camera and scene parameters, such as focus, exposure, aperture settings, and new photographic imaging systems.

Methods based on variable camera focus [29] have been generally designed to es-

timate a depth map from a number of images captured with different focal settings. The main assumption, common to most algorithms available in the literature, is that the scene is *locally* approximated by a plane parallel to the image plane. This is called the *equifocal assumption* and it allows describing the imaging process as a linear convolution. However, this assumption smears shape details and is invalid across depth discontinuities. Recently, Zhang and Nayar [125] proposed a method that uses structured light and temporal defocus analysis to handle this problem, obtaining accurate depth estimates near discontinuities. On the other hand, many images of the scene need to be acquired for the depth map estimation.

Variable aperture has been exploited by Hasinoff and Kutulakos [41] to avoid the local window processing in depth from focus. They rely on the *confocal constancy property*, which states that as the lens aperture varies, the pixel intensity of a visible in-focus scene point will vary in a scene- independent way, that can be predicted by prior radiometric lens calibration. Good results are shown at sensor resolution for scenes with high geometric details and fine texture (like hair, flowers, etc.). The method is limited to handle large untextured areas and has a slow acquisition time, as multiple images with different aperture and focus settings need to be captured.

Methods based on the variation of camera exposure time [23] have been widely used to obtain high dynamic range images. The idea is to take multiple photographs of the scene with different amounts of exposure to recover the response function of the imaging process. With the known response function, the captured photos can be fused into a single, high dynamic range radiance map whose pixel values are proportional to the true radiance values in the scene. In a different application, Jia et al. [51] combine short with long exposure images to retain the color of the long exposure image, while

avoiding the noise of the short exposure image. More recently, Raskar et al. [83] proposed coded exposure photography, by fluttering the camera's shutter open and closed with a binary pseudo-random sequence, in order to avoid motion blur.

Time variation (i.e., capturing images at different time instants and processing them) has been exploited for different applications, including shape-time photography [36], interactive digital photomontage [2], and day-night fusion [84]. With the advent of inexpensive digital image sensors, large camera arrays [118] have been used in different scenarios, such as high speed videography, and light field rendering. Also, significant research has been done to create novel imaging sensors for computer vision and graphics. As an example, compact plenoptic cameras [1] have been proposed by inserting a microlens array between the sensor and the main lens, allowing automatic digital refocusing and applications in range finding.

Chapter 3

Varying Illumination Parameters for Robust Depth Edge Detection

In this chapter we first describe our basic algorithm for depth edge detection (initially proposed by Raskar et al. [85, 86]) which relies on a simple and inexpensive modification of the capture setup: a multi-flash camera is used with flashes strategically positioned to cast shadows along depth discontinuities in the scene, allowing accurate shape extraction. Then, we show that by varying illumination parameters, such as the number, spatial position, type, and wavelength of light sources, we are able to handle fundamental problems in depth edge detection, including multi-scale depth changes, specularities, and motion. Our techniques are simple to implement, and require neither calibration nor parameter settings.

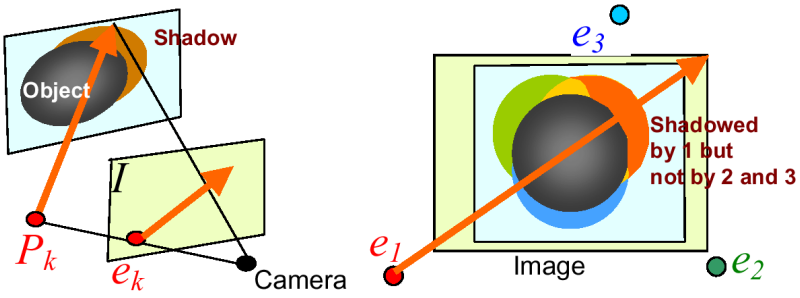


Figure 3.1: *Imaging geometry. Shadows of the gray object are created along the epipolar ray. We ensure that depth edges of all orientations create shadow in at least one image while the same shadowed points are lit in some other image.*

3.1 Depth Edges with Multi-Flash

The depth edge detection method is motivated by the observation that when a flash (*close* to the camera) illuminates a scene during image capture, thin slivers of cast shadow are created at depth discontinuities. Moreover, the position of the shadows is determined by the relative position of the camera and the flash: when the flash is on the right, the shadows are create on the left, and so on. Thus, if we can shoot a sequence of images in which different light sources illuminate the subject from various positions, we can use the shadows in each image to assemble a depth edge map using the shadow images.

3.1.1 Imaging Geometry

In order to capture the intuitive notion of how the position of the cast shadows are dependent on the relative position of the camera and light source, we examine the imaging geometry, illustrated in Figure 3.1. Adopting a pinhole camera model, the projection of the point light source at P_k is at pixel e_k on the imaging sensor. We call

this *image* of the light source the *light epipole*. The images of (the infinite set of) light rays originating at P_k are in turn called the *epipolar rays*, originating at e_k.

There are two simple observations that can be made about cast shadows:

- A shadow of a depth edge pixel is constrained to lie along the epipolar ray passing through that pixel.
- When a shadow is induced at a depth discontinuity, the shadow and the light epipole will be at opposite sides of the depth edge.

These two observations suggest that if we can detect shadow regions in an image, then depth edges can be localized by traversing the epipolar rays starting at the light epipole and identifying the points in the image where the shadows are first encountered.

3.1.2 Removing and Detecting Shadows

The approach for reliably removing and detecting shadows in the images is to position lights so that every point in the scene that is shadowed in some image is also captured without being shadowed in at least one other image. This can be achieved by placing lights strategically so that for every light, there is another on the opposite side of the camera to ensure that all depth edges are illuminated from two sides. Also, by placing the lights close to the camera, we minimize changes across images due to effects other than shadows.

To detect shadows in each image, we first compute a *shadow-free image*, which can be approximated with the MAX composite image, which is an image assembled by choosing at each pixel the maximum intensity value among the image set. The shadow-free image is then compared with the individual shadowed images. In particular, for

each shadowed image, we compute the *ratio image* by performing a pixel-wise division of the intensity of the shadowed image by the intensity of the MAX image. The ratio image is close to 1 at pixels that are not shadowed, and close to 0 at pixels that are shadowed. This serves to accentuate the shadows and remove intensity transitions due to surface material changes.

3.1.3 Algorithm

Codifying the ideas discussed we arrive at the following algorithm. Note that the shadowed images I_k in the algorithm below have ambient component I_0 removed, where I_0 is an image taken with only ambient light and none of the n light sources on.

Given n light sources positioned at $P_1, P_2...P_n$,

- Capture ambient image I_0
- Capture n pictures $I_{k,0}$, $k = 1..n$ with a light source at P_k
- Compute $I_k = I_{k,0} - I_0$
- For all pixels x, I_{max}(x) = max$_k(I_k(x))$, $k = 1..n$
- For each image k,

  ▷ Create a ratio image, R_k, where

  $R_k(x) = I_k(x)/I_{max}(x)$
- For each image R_k

  ▷ Traverse each epipolar ray from epipole e_k

  ▷ Find pixels y with step edges with negative transition

  ▷ Mark the pixel y as a depth edge

The following configuration of light sources is adopted: four flashes at left, right, top and bottom positions (Figure 3.2(a)). This setup makes the epipolar ray traversal efficient. For the left-right pair, the ray traversal is along horizontal scan lines and for the top-bottom pair, the traversal is along vertical direction. Figure 3.2(b) illustrates ratio image traversal and depth edge detection. We refer to our Siggraph 2004 paper [85] for more details on the algorithm and discussion with related techniques.

3.1.4 Limitations

In spite of the robustness of this method in real-world scenes, it has the following limitations:

- Limited to handle outdoor scenes, due to sun light;
- Fails with specular reflections;
- Limited to detect depth edges in multiple scales;
- Fails with transparent or low albedo objects;
- Depth edges are not detected if there is no background;
- Limited to handle dynamic scenes.

Next we will describe some of these problems and provide solutions in a common multi-flash imaging framework.

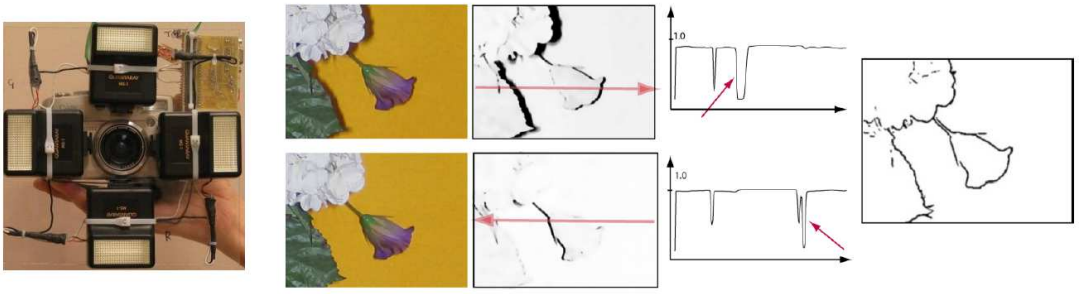


Figure 3.2: *(a) Multi-flash camera. (b) From left to right: photo, ratio image, plot along an epipolar ray (the arrow indicates negative transitions) and detected edges.*

3.2 A Multi-Baseline Approach

Depth discontinuities in real world scenes are associated with different amounts of depth changes, referred as "jumps of discontinuities" by Birchfield [11]. Ideally, we want a method that is able to detect depth edges at different scales, ranging from tiny to large changes in depth. We will show how to deal with this problem by taking advantage of the spatial position of the light sources, in our multi-flash imaging framework.

3.2.1 Baseline Tradeoff

Depth edges associated with small changes in depth might be missed due to an undetectable narrow shadow width, caused by a small camera-flash baseline. On the other hand, a larger baseline may cause detached shadows (separated from the object), leading to false depth edges.

In order to analyze this problem in more detail, we look at the imaging geometry of the shadows, depicted in Figure 3.3a. The variables involved are f (camera focal length), B (camera-flash baseline), z_1, z_2 (depths to the shadowing and shadowed

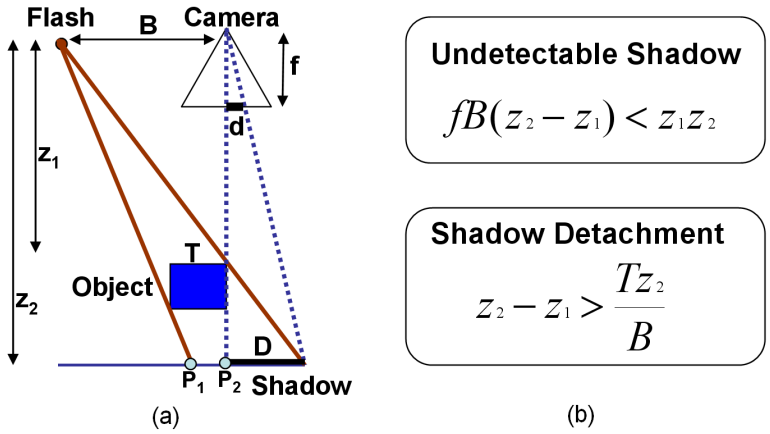


Figure 3.3: *(a) Relationship between baseline and shadow width. (b) Conditions for undetectable shadow and shadow detachment.*

edges), D (shadow width) and d (the shadow width in the image plane). We have that $\frac{d}{f} = \frac{D}{z_2}$ and $\frac{D}{z_2 - z_1} = \frac{B}{z_1}$. It follows that the shadow width in the image can be computed with the following equation:

$$d = \frac{fB(z_2 - z_1)}{z_1 z_2} \tag{3.1}$$

For small depth changes in the scene, far away from the camera, it is possible that $fB(z_2 - z_1) < z_1 z_2$. In this case, the shadow will not appear in the image, leading to missing depth edges.

We note that there are two ways of solving this problem: either we improve camera resolution, with larger focal length f, or we use a wider camera-flash baseline B, which is more convenient, since camera resolutions are limited. However, a larger baseline may cause detached shadows, mainly in narrow objects. Let T be the width of the object. Analyzing Figure 3.3a, we note that as we increase the baseline, point P_1 moves towards point P_2. Shadow detachment will occur when point P_1 passes over point P_2. When they are at the same position, from the imaging geometry, we have that

$\frac{T}{z_2 - z_1} = \frac{B}{z_2}$. It follows that if the amount of depth change $z_2 - z_1 > \frac{Tz_2}{B}$, the shadow will be separated from the object and a false depth edge will be marked. Figure 3.3b illustrates the two main conditions of the baseline tradeoff.

3.2.2 Proposed Solution

Our approach to handle the baseline tradeoff is to use a multi-baseline photometric method, where extra light sources are placed at different baselines, as shown in Figure 3.4a. With this new configuration, we are able to detect depth edges associated with small changes in depth (using large baselines), without creating shadow separation in narrow objects (using small baselines).

The main question is how to combine the information of the images taken with flashes at different baselines. A simple, yet effective way is to just take the minimum composite among the images. Provided that the light sources are sufficiently close to each other, the shadows in different baselines will merge, avoiding detached shadows, while preserving sufficiently large shadow widths.

In our implementation setup, we used two different baselines (see Figure 3.4b). Let F_S and F_L be the small and large baseline flashes, respectively. There are essentially four cases we need to consider at depth edges (Figure 3.5):

(i) F_S creates an undetectable narrow shadow and F_L creates a detectable shadow;

(ii) F_S creates a detectable small width shadow and F_L creates a larger width shadow;

(iii) F_S creates a detectable shadow but F_L creates a detached shadow that overlaps with F_S shadow;

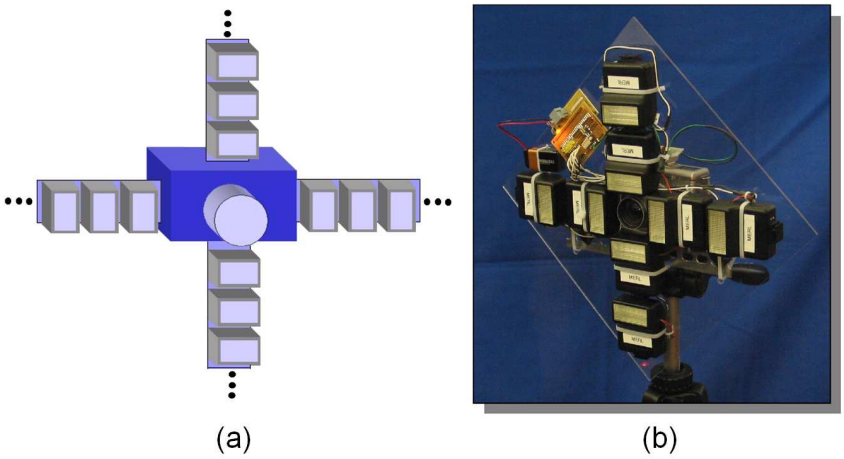


Figure 3.4: *(a) Our multi-baseline camera prototype. (b) Our implementation setup with two baseline levels.*

(iv) same as (iii) but the shadows of F_S and F_L do not overlap.

Note that in the first three cases, the minimum composite solution is suitable, but in the fourth case, there is a non-shadowed region between the shadows created by F_S and F_L. This would lead to a false depth edge along the shadow created by F_L. Next we describe an algorithm to handle this problem.

Eliminating Detached Shadows

Our algorithm is based on the observation that if the start point of a shadow created by F_S is not the start point of a shadow created by F_L, then the next shadow along the epipolar ray created by F_L is a detached shadow. Figure 3.6 shows the values of the *ratio images* along a scanline for F_S and F_L, when shadow detachment occurs. Our algorithm can be summarized in the following steps:

- Traverse along the epipolar rays the ratio images R_S and R_L, associated with F_S and F_L, respectively.

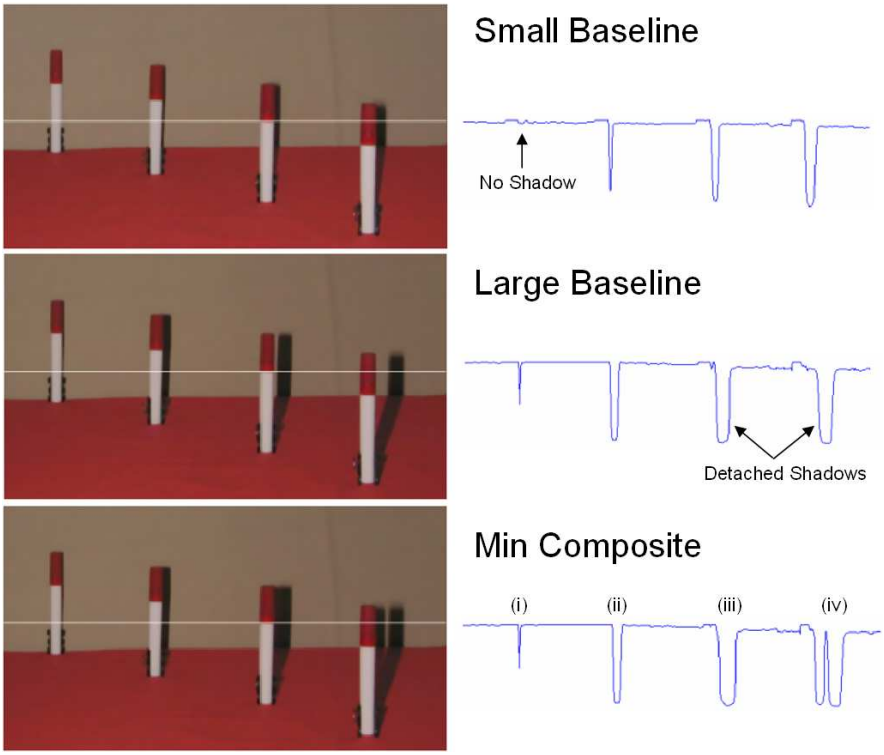


Figure 3.5: *Analysis of the four cases related to the baseline tradeoff, considering a narrow object.*

- If a depth edge appears in R_S and not in R_L (see points A_1 and A_2 in Figure 3.6):
  - Traverse R_L along the epipolar ray until the next detected depth edge (see point B_2 in Figure 3.6).
  - If at this position there is no correspondent depth edge in R_S (see point B_1 in Figure 3.6), we mark this edge as a spurious, detached shadow edge.

The last step is important to confirm the presence of a detached shadow as well as to ensure that no edges detected with the small baseline flash F_S will be marked as spurious.

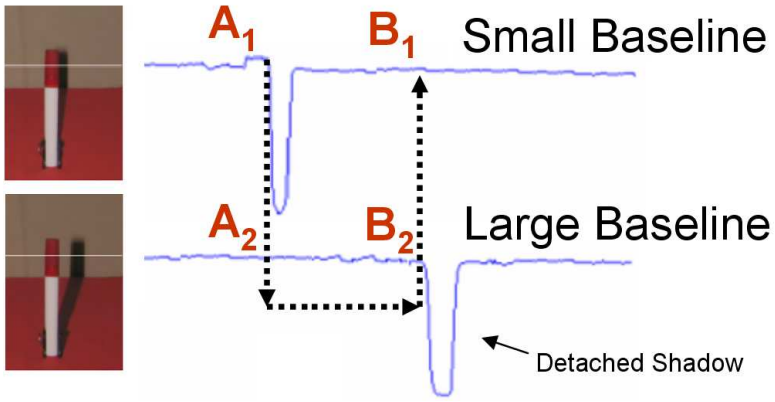


Figure 3.6: *Algorithm for eliminating detached shadows when the light sources are not sufficiently close to each other.*

Note that using this algorithm eliminates the problem in case (iv), when shadows of F_S and F_L do not overlap. This solution will fail to detect depth discontinuities when even F_L does not create a detectable shadow or for very thin objects, where even F_S creates a detached shadow. In this case, extra light sources could be added to our setup.

Another interesting point is that in addition to detect depth edges under multi-scale depth changes, we could also determine the amount of depth change, since it is proportional to the shadow width. This could be helpful in high level vision tasks, such as segmentation or 3D reconstruction.

3.2.3 Multi-Scale Depth Edges in Real-World Scenes

Figure 3.7a shows an image captured with a small camera-flash baseline. Since there are very small depth changes in the interior region of the pinecone, depth edges are missed in this region (Figure 3.7b). On the other hand, if we use a large baseline, the shadow gets detached from the basket (Figure 3.7c), leading to a false depth edge (Figure 3.7d). Using our multibaseline approach with two levels of baseline, we are

able to eliminate detached shadows, and still preserve depth edges associated with small changes in depth, as shown in Figure 3.7e.

A more complex example is depicted in Figure 3.8a, which shows the image of a car engine, containing depth edges associated with different amounts of depth changes. Figure 3.8b shows intensity edge detection for this image, using the Canny operator. Note that important shape boundaries are missing due to low contrast variations, while edges due to texture variations not associated with occluding edges confound scene structure.

We compared our multibaseline approach with the naive single-baseline algorithm in such complex scene (Figures 3.8c and 3.8d). Note that our method captures depth edges associated with tiny and larger changes in depth, which is not possible with our previous setup. Figure 3.8e better illustrates this comparison, zooming into a specific part of the engine.

In our implementation setup, the baselines are about 50mm and 100mm. Using a 4.0 MegaPixel Canon G3, we verified that we can capture depth discontinuities with changes in depth as small as 5mm at a distance no larger than 2000mm from the camera.

3.2.4 Limitations

A multibaseline system is quite dependent on the depth complexity of the scene. It is possible that for some applications (e.g., extracting internal finger contours on the hand), a single-baseline system will be sufficient. The limitations are evident for very thin objects, where the flash closest to the camera causes shadow detachment, and for objects very close to the background, where even the large baseline light may not produce a visible shadow in the image. Increasing the baseline may cause problems

due to more shading and non-Lambertian effects, where texture edges may be marked as depth edges.

Another limitation is the slow acquisition time, i.e., many images of the scene need to be captured with each flash at different periods of time in order to cover multiple baselines (a total of eight images for a two-baseline setup).

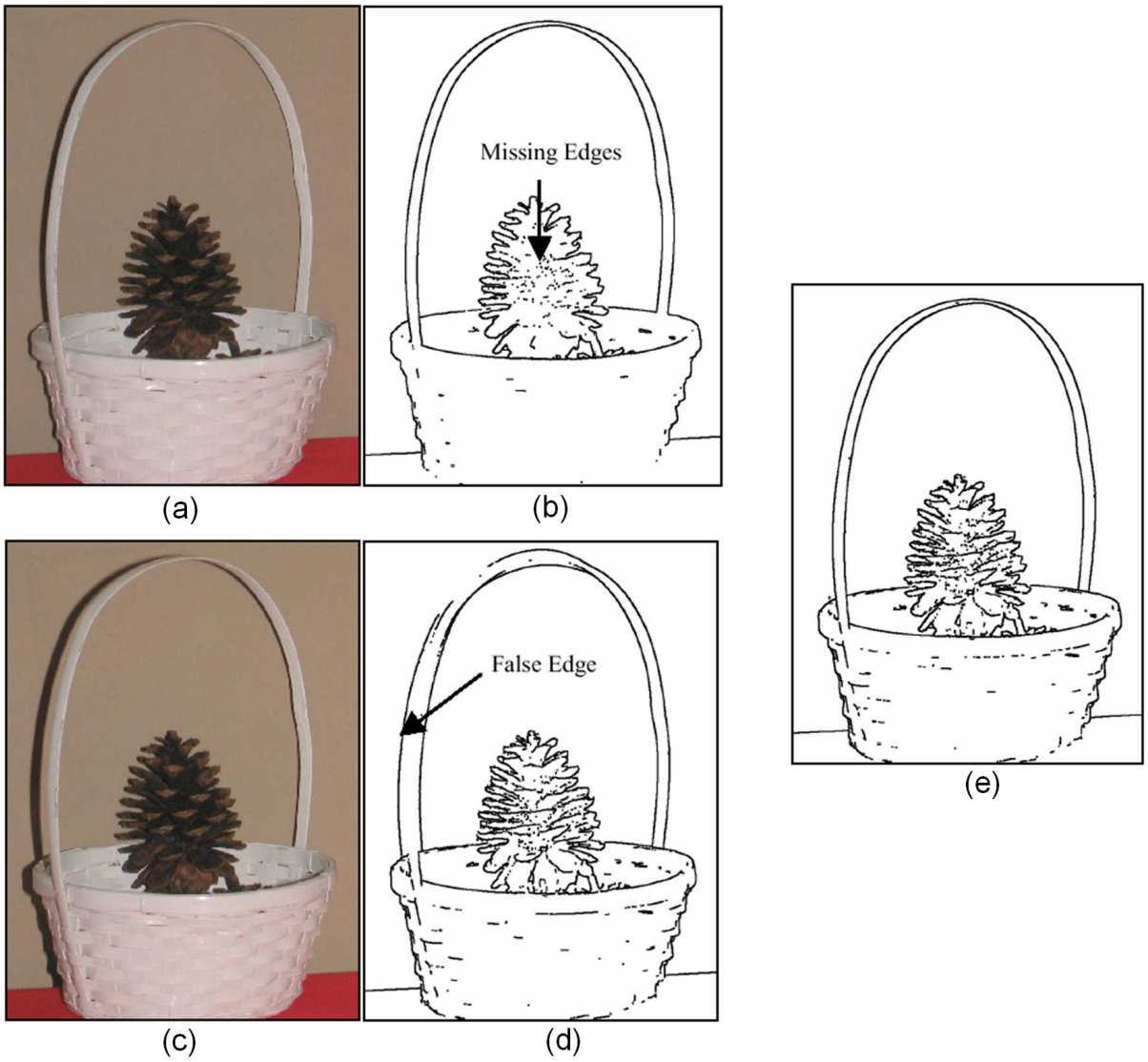


Figure 3.7: *(a) Small baseline image and (b) correspondent depth edges. (c) Large baseline image with shadow detachment and (d) correspondent depth edges. (e) Our final result using our multibaseline approach.*

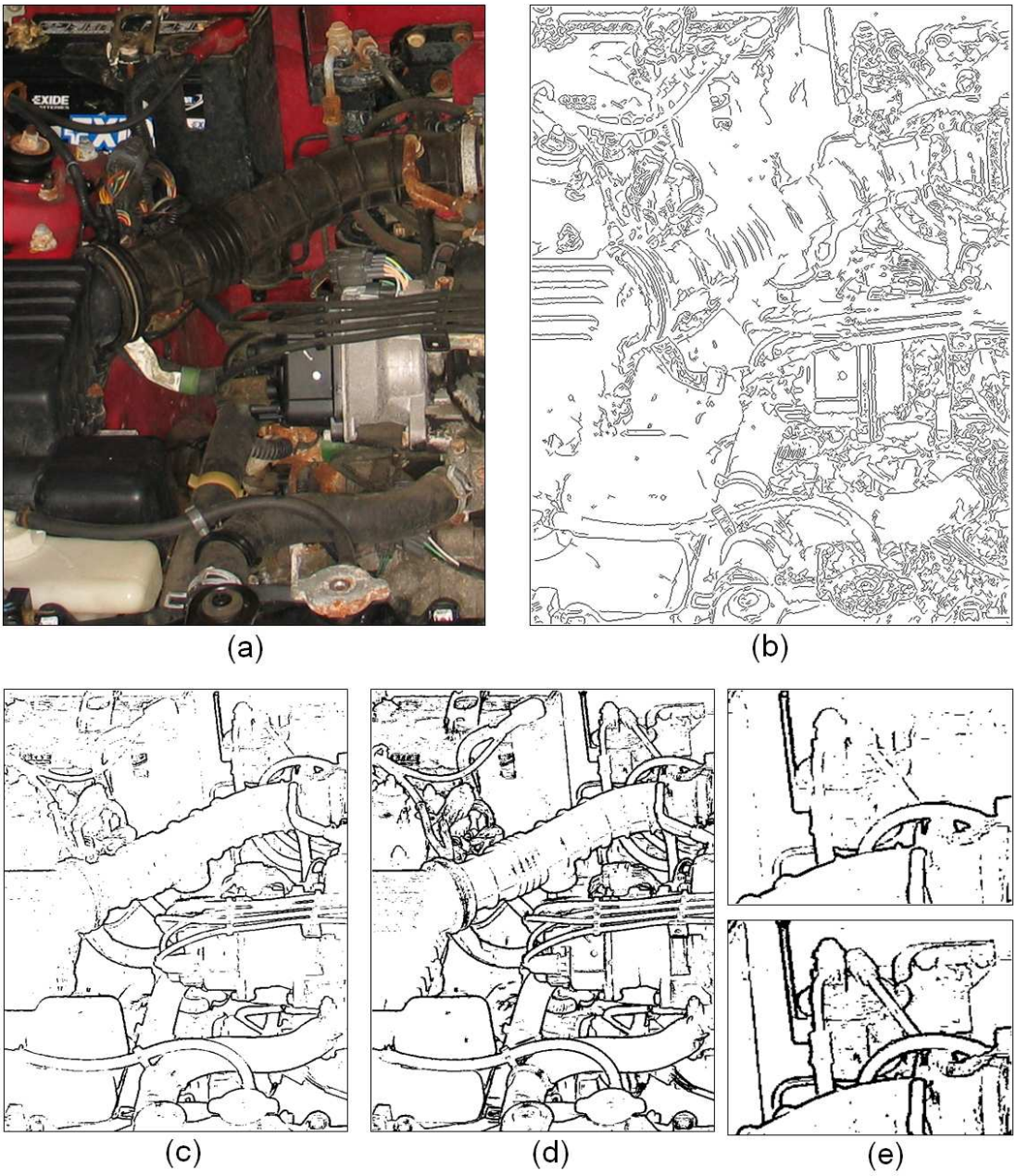


Figure 3.8: *(a) Complex scene with different amounts of changes in depth. (b) Canny edge detection. (c) Depth edges computed with one single camera-flash baseline. (d) Depth Edges computed with our multibaseline approach. (e) Part of the engine zoomed in to compare single-baseline depth edges (top) with our multibaseline method (bottom).*

3.2.5 Linear Light Source Analysis

As we mentioned in the previous section, There are two key limitations associated with our multibaseline technique: the slow acquisition time and when shadow detachment occurs even for the flash closest to the camera (where a spurious edge is marked as a depth edge).

We handle these issues in our framework by varying another illumination parameter: the type of the light source. We basically use linear lights, as shown in Figure 3.9a, so that we are able to cover different baselines with a fast image acquisition. For this particular setup, four images are required to be captured as in our original algorithm.

We need to consider three cases for detecting depth edges with linear light sources:

(i) No shadow detachment occurs.

(ii) Only part of the linear light source causes shadow detachment.

(iii) Shadow detachment occurs for any point along the linear light source.

Figure 3.9b illustrates case (i). As typical for linear lights, we have the umbra region (extending from A to B), where all the light from the source is completely blocked by the object, and the penumbra region (extending from B to C), where the shadow is partial, i.e., only part of the light is blocked. In the ratio plot, we have a sharp negative transition in A and thus the depth edge can be correctly marked.

Case (ii) is shown in Figure 3.9c. Here, part of the light source causes shadow detachment (region from A to B). Right before point A in the figure, the object is being illuminated by all the light from the source, but at point A most of this light is blocked (remaining only the light due to shadow detachment). This causes a drop in

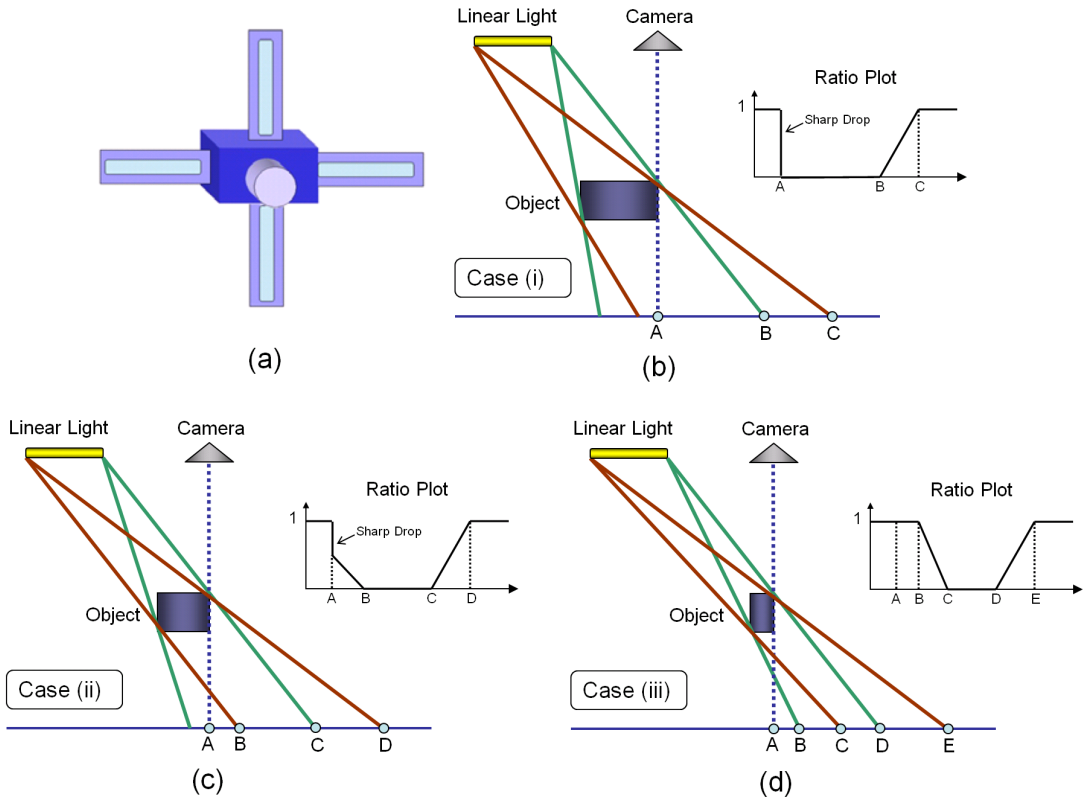


Figure 3.9: *(a) Prototype setup with four linear light sources. (b) Case (i) analysis. (c) Case (ii) analysis. (d) Case (iii) analysis.*

the ratio plot, and thus the depth edge can be correctly marked. From A to B, there is a smooth transition until the light is completely blocked. Then, as in case (i), we have a smooth positive transition from C to D.

Finally, Figure 3.9d illustrates case (iii), where all the points along the linear source cause shadow detachment. This means that the region from A to B is fully illuminated by the linear light and thus the depth edge can not be detected, as there is no sharp negative transition in the ratio plot. On the other hand, no spurious depth edge is marked due to shadow detachment. The reason is that there is a smooth negative transition

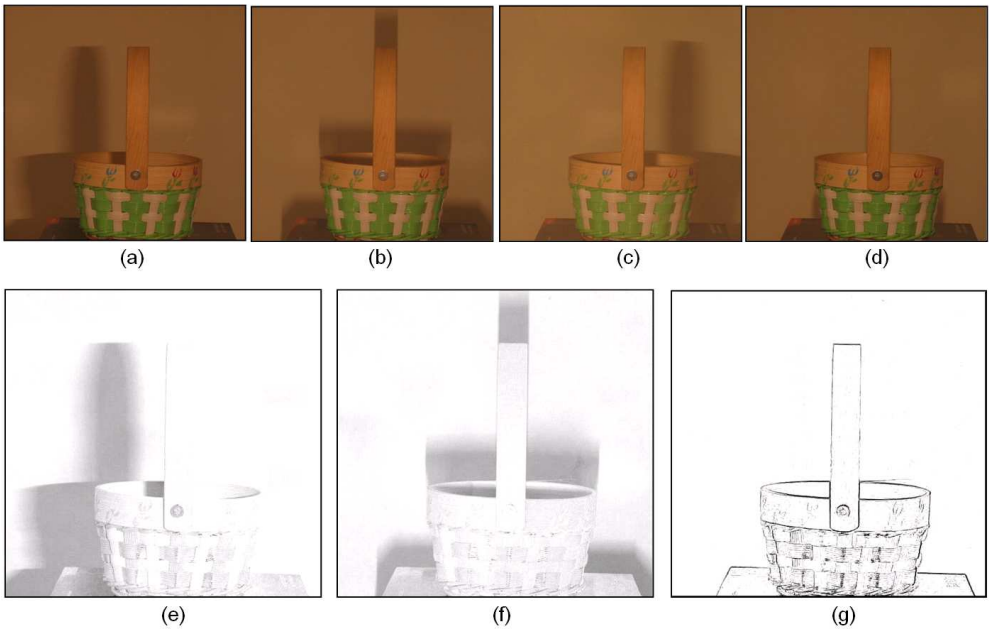


Figure 3.10: *(a) - (d) Image capture with four linear light sources. (e) Ratio image associated with right flash. Note the smooth negative and positive transitions along the detached shadow. (f) Ratio image associated with bottom flash. Here we have sharp negative transitions along depth edges. (g) Depth edge confidence map. No spurious edges are marked due to detached shadows.*

(rather than a sharp drop) from B to C until the light is completely blocked. This is important for specific applications, such as non-photorealistic rendering, where marking spurious edges may be much more undesirable than missing some depth contours. As in cases (i) and (ii), we have a smooth positive transition from D to E. Depending on the length of the linear source, it could be possible that the shadow detachment region is large enough so that point C passes over point D, but still the same properties would remain valid.

The usefulness of linear lights in real scenes is demonstrated in Figure 3.10. The image capture with four linear light sources (40W, 6-inch tubular bulbs) is illustrated in Figures 3.10a-d. The ratio image in Figure 3.10e shows a detached shadow with

a smooth positive and negative transition (case (iii)). In Figure 3.10f, the ratio image contains only attached shadows with sharp negative transitions at depth contours. The depth edge confidence map, which is basically formed by assembling the results of a Sobel operator on the four ratio images, is shown in Figure 3.10g. Note that there are no spurious edges due to detached shadows. We are not processing specular reflections for this scene, so additional edges appear on the basket.

Discussion

Linear light sources are useful for detecting depth edges with multi-scale depth changes. They offer the advantage of fast acquisition, compared to the technique based on point light sources described in Section 3.2.2. Also, spurious edges due to detached shadows are avoided, even if all points along the linear source cause detached shadows. This is important for e.g., non-photorealistic rendering, where avoiding emphasizing contours not associated with object shape or intensity changes is critical.

There are situations, however, where the use of our technique based on point light sources leads to better results. For example, case (ii) for linear lights (Figure 3.9c) is more sensitive to noise, since the sharp drop in the ratio plot is smaller in magnitude. Another source of error arises in complex scenes, where depth edges may lie in shadowed regions, leading to undetected depth contours. This is illustrated in Figure 3.11. Note that this case is much more common when using linear sources or flashes with large baseline. Missing edges due to this problem could be detected using the images taken with small baseline point light sources.

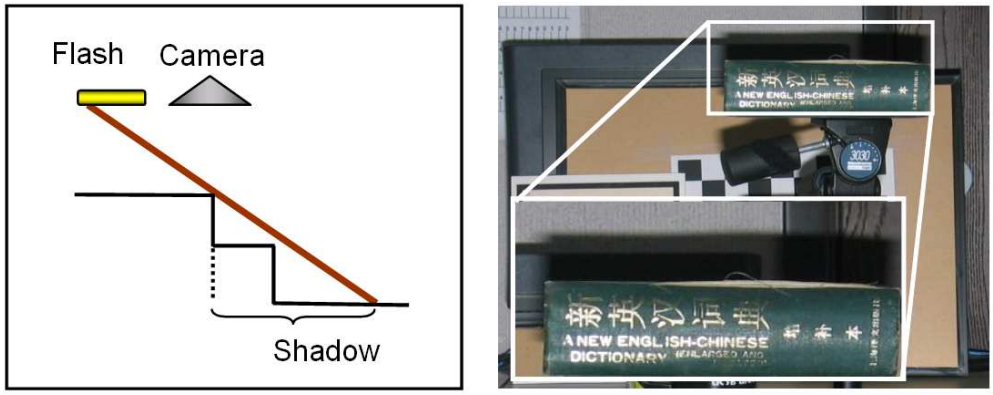


Figure 3.11: *Large baseline light sources may miss depth edges that lie in shadowed regions.*

3.2.6 Lack of Background

What happens if the amount of depth change is infinite, i.e., there is no background present in the scene? In this case, the external object contours (edges shared by the foreground and the distant background) are missed in our method, as no shadows are created on the background.

By capturing an image with flash and another without flash, we note that the background remains the same in both images (as the flash does not affect the far away background, due to its intensity fall-off), but the object becomes brighter in the flash image. This information can be used to extract the external contours of the object.

Figure 3.12a shows the max composite flash image of a toy placed in front of a window to show the lack of background. Figure 3.12b shows the correspondent no-flash image. By taking the ratio between the images, the external contour of the object can be easily segmented, as shown in Figure 3.12c. The ratio is near 1.0 in background and close to zero in the interior of the foreground. In this image, we eliminated small blobs due to object motion (for example, cars moving in the background). Figure 3.12d

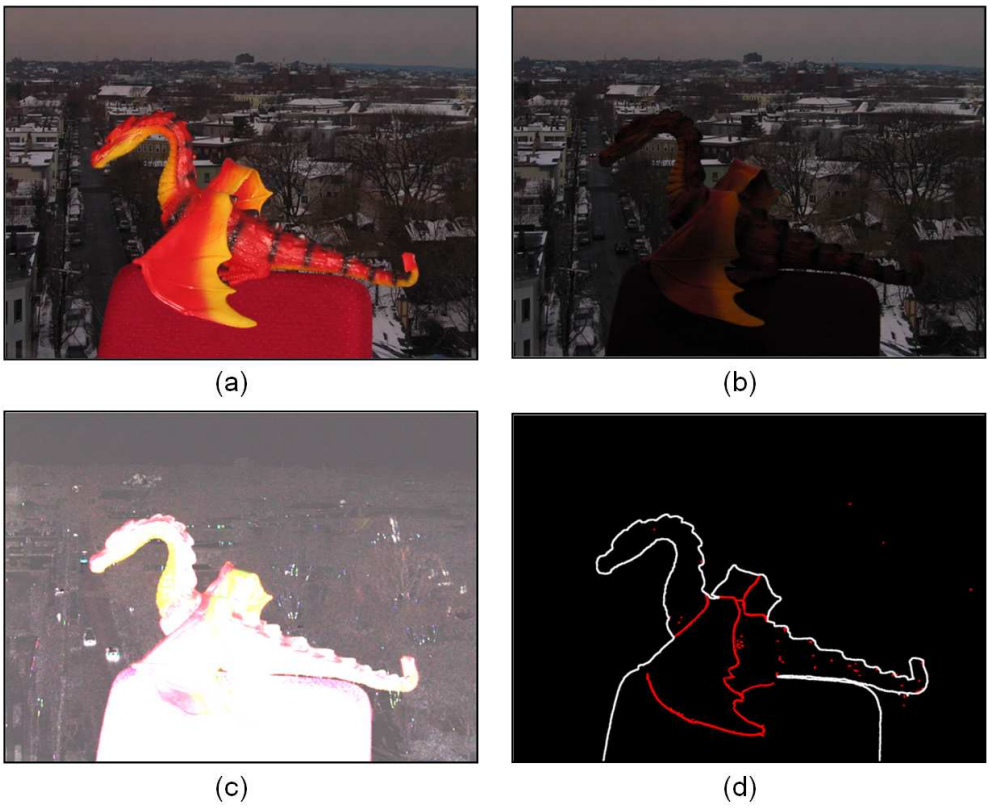


Figure 3.12: *(a) Max composite image. (b) No-Flash image. (c) Ratio between no-flash and max composite images. (d) external contour (white) and internal depth edges (red).*

shows in white edges the external contour of the object, while the red edges correspond to the internal depth contours computed with our multi-flash technique.

Our idea of detecting external depth contours with flash and non-flash image pairs has been recently adapted and applied to image matting [105]. Limitations include the presence of dark objects on the scene and also a medium-far background, which can be affected by the flash without detectable shadows.

3.3 Dealing with Specularities

The reflection of light from surfaces in real scenes is generally classified into two main categories: diffuse and specular. The diffuse component results from light rays penetrating the surface, undergoing multiple reflections, and re-emerging [70]. In contrast, the specular component is a surface phenomenon - light rays incident on the surface are reflected such that the angle of reflection equals the angle of incidence. Light energy due to specular reflections is often concentrated in a compact lobe, causing strong highlights (bright regions) to appear in the image.

These bright spots, also known as specularities, play a major role in many computer graphics and vision problems. They provide a true sense of realism in the environment, reveal important local curvature information [73] and may even provide additional cues for object recognition [74]. However, in most cases, specular highlights are undesirable in images. They are often considered as annoyance in traditional photography and cause vision algorithms for segmentation and shading analysis to produce erroneous results. If the sensor or lighting direction is varied, highlights shift, diminish rapidly, or suddenly appear in other parts of the scene.

Since specularities shift among images captured with differently positioned light sources, they pose a serious problem for our depth edge detection method based on multi-flash imaging. Specular highlights that appear at a pixel in one image but not others can create spurious transitions in the ratio images. More specifically, the ratio between a non-shadowed pixel and a specular pixel in other image will cause a drop in the ratio image, leading to false detected depth edges (see Figure 3.13a).

A variety of photometric techniques have been proposed for detecting and remov-

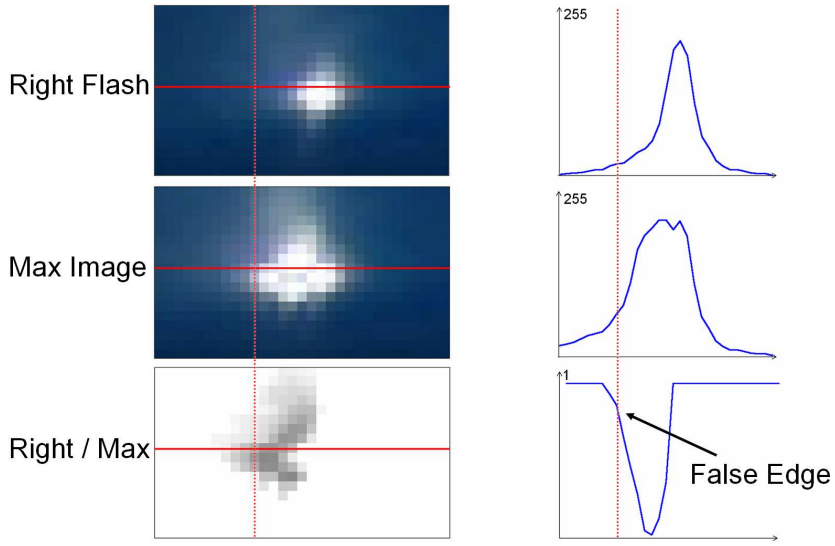


Figure 3.13: *Specularities can create spurious transitions in the ratio images, leadind to false detected depth edges.*

ing specularities using color [55], polarization [120], multiple views [63] and hybrid methods [70]. However, most of these techniques assume that highlights appear in regions with no variation of material type or surface normal. In fact, reliably removing specularities in textured regions remains a challenging problem. Next we will provide a robust solution to handle this problem, within our multi-flash imaging framework.

3.3.1 Gradient-Domain Approach

We need to consider three cases of how specular spots in different light positions appear in each image:

(i) shiny spots remain distinct on a high specular surface.

(ii) some spots overlap.

(iii) spots overlap completely (no shift).

In order to remove specularities, we could compute the minimum composite image by selecting the minimum intensity pixel of each correspondent point in the input images. This would considerably reduce the effect of specularities, but the resultant image would be full of shadows. A more reasonable approach is to take the median of the images, which removes shadows and specular reflections when they do not overlap (case i). However, when specularities overlap (case ii), the median image contains spurious specular pixels. Later in this section we demonstrate synthetic results for case analysis, illustrating this problem.

Active illumination approaches that handle specularities [100], in general assume that specularities do not overlap among images, considering light sources distant from the camera. In our case, flashes are positioned close to the center of projection of the camera, and therefore this assumption is not valid.

We note that although specularities overlap in the input images, the boundaries (intensity edges) around specularities in general do not overlap. The main idea is to exploit the gradient variation in the n images, taken under the n different lighting conditions, at a given pixel location (x,y). If (x,y) is in a specular region, in cases (i) and (ii), the gradient due to the specularity boundary will be high in only one or a minority of the n images. Taking the median of the n gradients at that pixel will remove this outlier(s). We then reconstruct the specular-reduced image from the gradient field. Our method is motivated by the intrinsic image approach [117], where the author removes shadows in outdoor scenes by noting that shadow boundaries are not static.

Let I_k, $1 \leq k \leq n$ be an input image taken with light source k and $Imax(x) = max_k(I_k(x))$ be the maximum composite of the input images. We compute the specular-reduced image through the following steps:

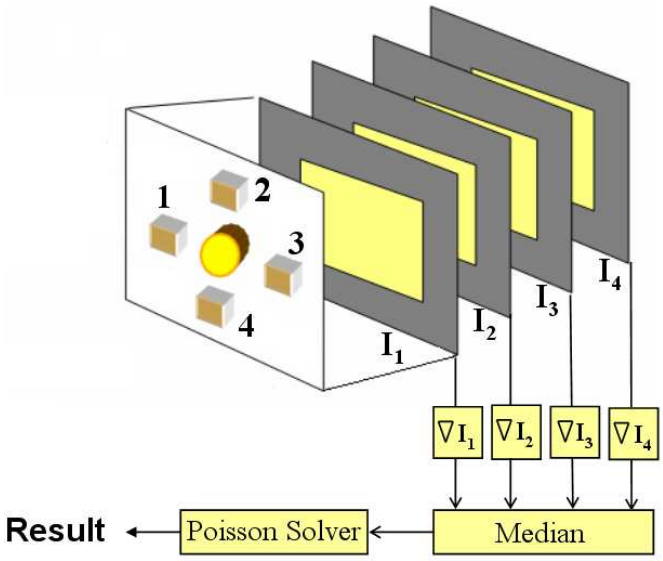


Figure 3.14: *Our gradient-domain approach to reduce the effect of specularities in images.*

- Compute intensity gradient, $G_k(x, y) = \nabla I_k\,(x, y)$
- Find median of gradients, *G(x,y)*= median$_k(G_k(x, y))$
- Reconstruct image $\hat{I}$ which minimizes $\left|\nabla \hat{I} - G\right|$

This algorithm is illustrated in Figure 3.14, considering a camera with four flashes. We will detail the method used for reconstructing image $\hat{I}$ (step 3) in Section 3.3.2.

Figure 3.15 shows a simple example to illustrate our method in all three cases mentioned above. For each case, we created four images with manually drawn specularities. The first column in the figure corresponds to the max composite of the four images (Imax), the second corresponds to the median of intensities (Imedian) and the third column is the output of our method - the reconstruction from the median of gradients (Iintrinsic).

Note that if we consider Imedian, specularities are not eliminated in case (ii), where spots overlap. On the other hand, our method is able to handle cases (i) and (ii), which

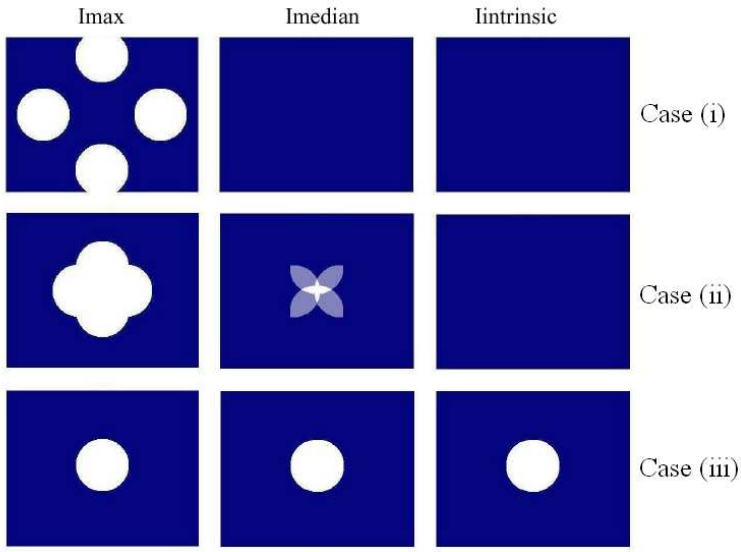


Figure 3.15: *Illustration of the three cases. Note that if we consider only the median of image intensities (instead of median of gradients), we have problems in case (ii). Our method based on the intrinsic image handles cases (i) and (ii) which often occur in practice. If specularities do not move among images our method fails to remove them.*

often occur in practice. If specularities do not move among images, our method fails to remove them. However, this is not a problem for depth edge detection, as no spurious edges are detected in this case.

3.3.2 Image Reconstruction from Gradient Fields

Image reconstruction from gradients fields, an approximate invertibility problem, is still a very active research area. In R^2, a modified gradient vector field G may not be integrable. In other words, there might not exist an image $\hat{I}$ such that $G = \nabla\hat{I}$. In fact, the gradient of a potential function must be a conservative field, i.e., the gradient $\nabla\hat{I} = (\frac{\partial\hat{I}}{\partial x}, \frac{\partial\hat{I}}{\partial y})$ must satisfy:

$$\frac{\partial^2\hat{I}}{\partial x\partial y} = \frac{\partial^2\hat{I}}{\partial y\partial x} \tag{3.2}$$

This condition is rarely verified for the gradient field G.

As noted by Frankot and Chellappa [34], one possible solution to this problem is to orthogonally project G onto a finite set of Fourier basis functions spanning the set of integrable vector fields. In our method, we employ a more direct and efficient approach, in a similar way to the work of Fattal et al. [28]. We search the space of all 2D potential functions for a function $\hat{I}$ whose gradient is the closest to G in the least-squares sense. In other words, $\hat{I}$ should minimize the integral:

$$\int\int F(\nabla\hat{I}, G)dxdy, \tag{3.3}$$

where $F(\nabla\hat{I}, G) = \|\nabla\hat{I} - G\|^2 = (\frac{\partial\hat{I}}{\partial x} - G_x)^2 + (\frac{\partial\hat{I}}{\partial y} - G_y)^2$.

According to the Variational Principle, a function $\hat{I}$ that minimizes the integral in (3.3) must satisfy the Euler-Lagrange equation:

$$\frac{\partial F}{\partial\hat{I}} - \frac{d}{dx}\frac{\partial F}{\partial\hat{I}_x} - \frac{d}{dy}\frac{\partial F}{\partial\hat{I}_y} = 0, \tag{3.4}$$

which is a partial differential equation in $\hat{I}$. Substituting F we obtain the following equation:

$$2(\frac{\partial^2\hat{I}}{\partial x^2} - \frac{\partial G_x}{\partial x}) + 2(\frac{\partial^2\hat{I}}{\partial y^2} - \frac{\partial G_y}{\partial y}) = 0 \tag{3.5}$$

Dividing by 2 and rearranging terms, we obtain the well-known Poisson Equation:

$$\nabla^2\hat{I} = divG \tag{3.6}$$

where ∇^2 is the Laplacian operator defined as $\nabla^2\hat{I} = \frac{\partial^2\hat{I}}{\partial x^2} + \frac{\partial^2\hat{I}}{\partial y^2}$, and $divG$ is the divergence of the vector field G, defined as $divG = \frac{\partial G_x}{\partial x} + \frac{\partial G_y}{\partial y}$.

We have used the standard full multigrid method [82] to solve the Poisson equation. We pad the images to square images of size the nearest power of two, before applying the integration, and then crop back the result to the original size.

3.3.3 Specular Mask

In real images, specular reflections may not have sharp boundaries. In this case, when they overlap among images, our method attenuates the specular regions, but do not completely remove them, possibly leading to spurious depth edges. We handled this problem by first taking the ratio between the specular-reduced image $\hat{I}$ and the maximum composite image $Imax$:

$$S = \frac{\hat{I}}{Imax} \tag{3.7}$$

Since both images have no shadows, the ratio S will have low values exactly along specular regions. By thresholding S, we obtain a specular mask S', where specular pixels are set to zero and non-specular pixels are set to one. Let D' be the depth edge map with spurious specular pixels. Then, we can compute the final depth edge map $D = D' * S'$, by applying the specular mask to remove spurious specular edges.

Note that by computing the specular mask, we are in fact *detecting* specular pixels, and not only attenuating the effect of specularities in images.

3.3.4 Experimental Results

This section presents our experiments, which were carried out with synthetic and real images, considering objects with different local curvature and textured regions.

Synthetic Images

We first demonstrate our method in synthetic images, showing that it can reliably remove specularities in textured regions, as long as spots shift among images.

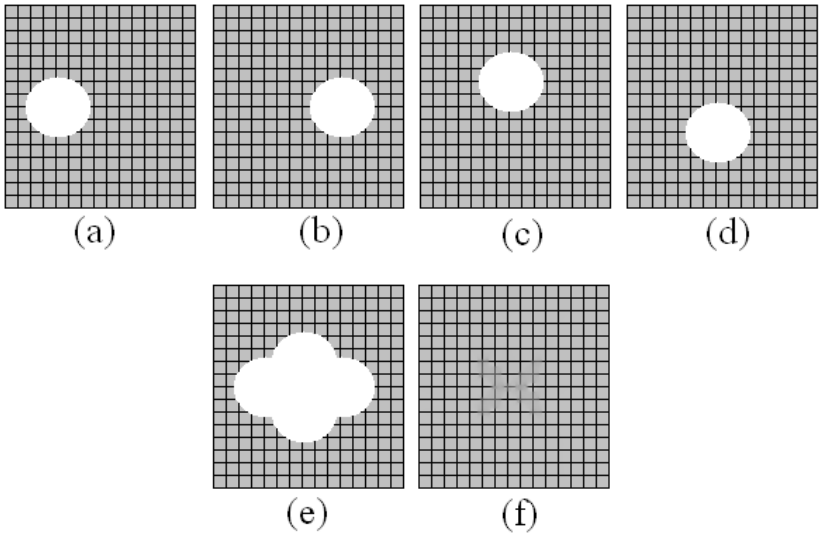


Figure 3.16: *(a-d) Four images with manually drawn specularities along a textured region. (e) Max composite image. (f) Result of our method.*

Figure 3.16a-d shows four images of a textured region, with manually drawn specularities (white circles). Figure 3.16e shows the max composite image (case (ii), where spots overlap) and our result is shown in Figure 3.16f. Note that we are able to eliminate the specularities, while preserving the texture.

Real Images

Figure 3.17(a-d) illustrates our capture process of a specular scene with objects of different curvature and textured regions. Here we consider only four light sources to demonstrate the robustness of our method against specularities. Figures 3.17e and 3.17f show that the median of the magnitude of gradients is considerably attenuated along specular regions. This allows us to reconstruct a specular-reduced (or intrinsic) image, as shown in Figure 3.17g. Note that most of the specular reflections are eliminated, despite textured regions in the scene. Moreover, shadows are also eliminated, since their boundaries do not overlap among images.

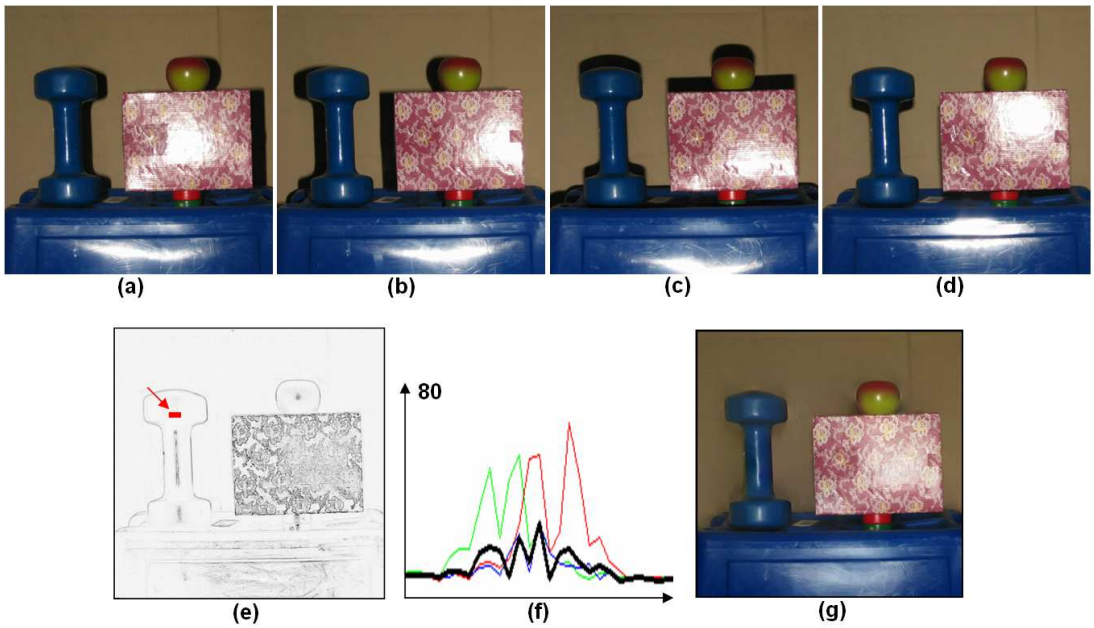


Figure 3.17: *(a)-(d) Image capture process. (e) Median of magnitude of gradients. (f) Plot of magnitude of gradients along a scanline in a specular region. The black line is the median, which is clearly attenuated. (g) Our specular-reduced image*

The intrinsic image is therefore a good alternative for the max composite image, which is full of specularities, and cause spurious edge pixels in the depth edge map (see Figures 3.18a and 3.18b). As we mentioned before, we also compute a specular mask, due to the fact that in real images, specularities are smooth, leading to attenuation, but not complete removal of these bright spots. Figures 3.18c and 3.18d illustrate the specular mask computation. Our final result is shown in Figure 3.18e. More examples of depth edge detection with specular reflections are shown in Figure 3.19.

Figure 3.20a shows a challenging scene, involving a transparent object and a high albedo background. Note that we are able to detect most specularities (Figure 3.20c) without false positives. As we can see in our final result for this example (Figure 3.20d), shadows are eliminated and specularities are significantly reduced.

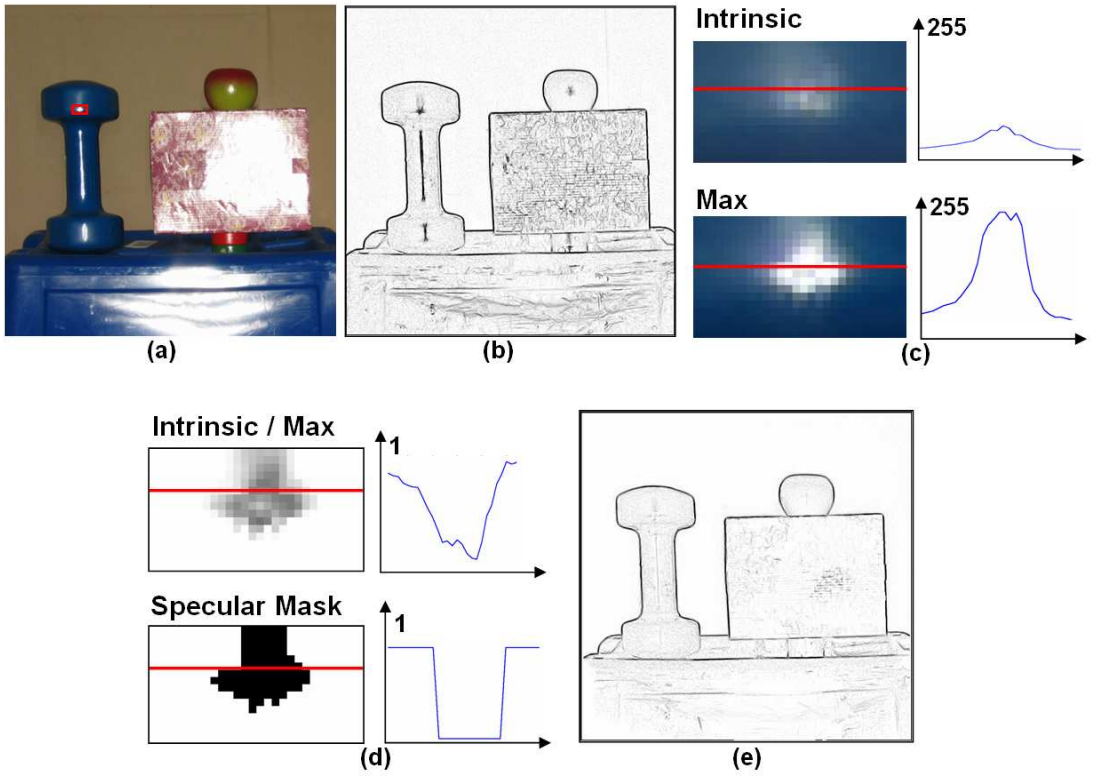


Figure 3.18: *(a) Max composite image and (b) correspondent depth edges. (c) Intensity plots along the red scanline of the region highlighted in leftmost figure. (d) Specular mask computation (e) Our final result*

The computational time required to obtain the specular-free images is about one second on a 3GHz Pentium 4, considering images with resolution 640x480.

3.3.5 Discussion

As we already mentioned, our method fails to remove specular highlights when they do not shift among images. Fortunately, this case is not problematic for many computer vision methods, including depth edge detection.

In our method, regions covered by specular highlights in one image may be specular-free in others, since bright spots often shift due to our multi-flash scheme. This allows

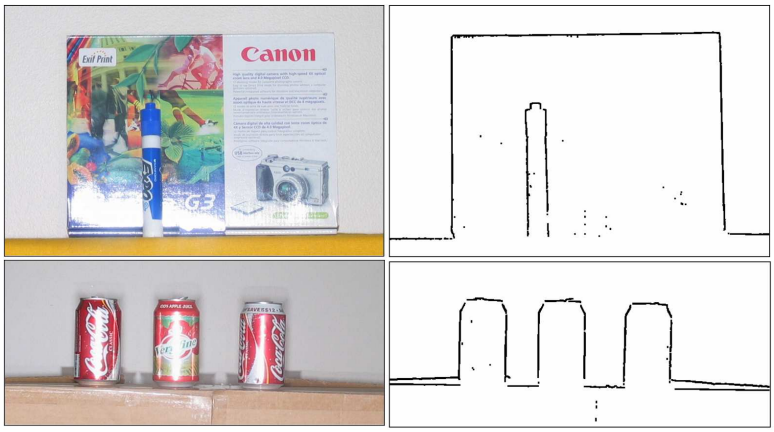


Figure 3.19: *Depth edge detection in specular scenes.*

Figure 3.20: *(a) Image taken with one of the flashes. (b) Maximum composite image. (c) Detection of specularities. (d) Specular-reduced image.*

us to "uncover" these regions, posing an advantage over previous approaches that attempt to remove specularities based on a single image.

By solving the Poisson partial differential equation, we eliminate most highlights and also shadows in the image. This compares favorably with the max composite (which has no shadows but is full of specularities), the min composite (which is specular-reduced but also full of shadows) and the median of intensities (which has problems when spots overlap, as shown in Figure 3.15). In addition, our method is not affected

by objects as bright as specularities in the scene (see Figure 3.20), which is clearly a problem for thresholding-based techniques.

We believe that more accurate results could be obtained by using more light sources (not just four as in our experiments) for specular reflection removal. Using only two flashes would not be a good choice, due to the fact that the gradient of the boundary of a specularity in one image would be only attenuated (not removed) when taking the mean with the other image. Choosing the minimum gradient could help in this case, but it would create a problem for textured regions.

The detection of specularities based on our specular mask could be useful in other applications, such as object shape information acquisition [70] and interactive creation of technical illustrations, where the user may decide whether to keep or remove specularities. Our detected specular regions could also be used as input for image inpainting methods, which often require manual labeling of such regions [110].

When specular boundaries are smooth and overlap among the images, the intrinsic image only attenuates the effect of specularities. The computation of the specular mask is useful in this case, but still it might leave some specular pixels undetected due to thresholding. It is also possible that specular boundaries may overlap in the majority of images, leading to no specular attenuation. Figure 3.21 shows an example where we can not remove specular artifacts. Low albedo objects are much more challenging because even attenuated specular pixels with small gradient magnitude may cause spurious edges.

Finally, we need to mention that we are processing “active specularities”, created by our light sources, rather than highlights due to ambient light. The use of flashes might be useful in many situations, such as in dark scenes, or even for vision methods

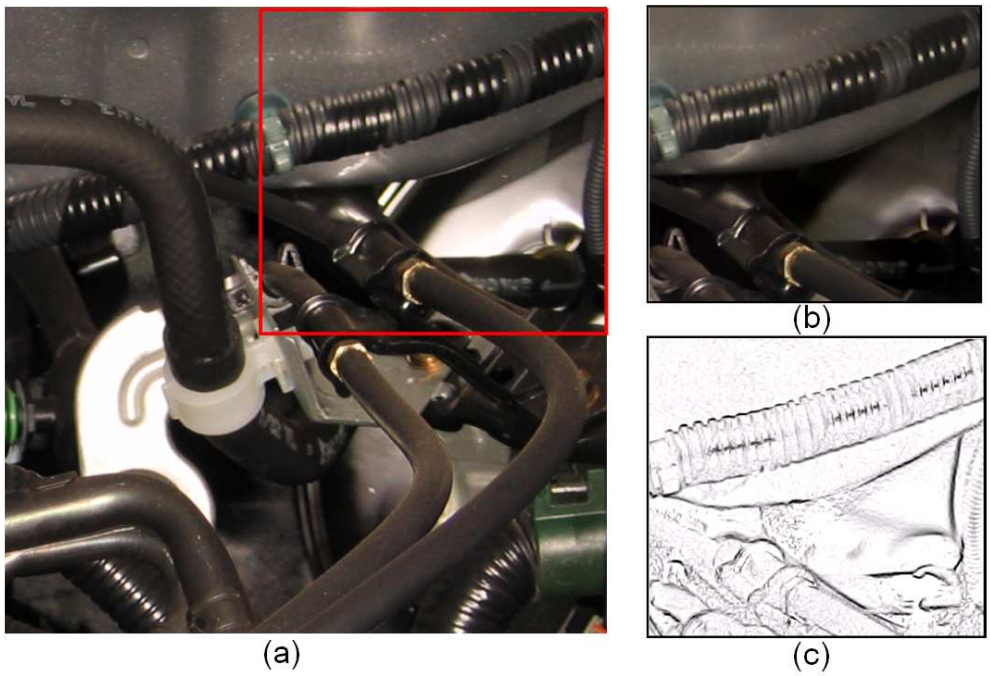


Figure 3.21: *A failure case for removal of spurious edges due to specularities. (a) car engine photo. (b) Intrinsic image, with few attenuation of specularities. (c) Depth edge confidence map.*

based on active lighting. In order to detect specularities in the ambient image, a method based on gradient direction analysis between flash and no-flash image pairs could be used, in the same spirit of the work of Agrawal et al. [4].

3.4 Variable Wavelength

So far, our method requires taking multiple pictures of the same static scene. This clearly poses a problem for detecting depth edges in motion. Due to the lack of simultaneity, the base maximum composite image will have misaligned features, leading to spurious edges.

In order to handle this problem, we could use a high speed camera, with flashes triggered in a rapid cyclic sequence, syncronized with the camera video frames. We note that a high speed camera can reduce the amount of motion between frames, but still the frame simultaneity cannot be assumed. A reasonable approach is to apply motion compensation techniques to correct the introduced artifacts. Finding optical flow and motion boundaries, however, is a challenging problem, mainly in textureless regions [11].

As in the static case, we bypass the hard problem of finding the rich per-pixel motion representation and focus directly on finding the discontinuities, i.e., depth edges in motion. Our approach is part of our common multi-flash framework, relying on the variation of the wavelength of the light sources. Very little attention has been given to photometric vision methods that make use of flat colored lights. We demonstrate here that by using light sources with different colors, we can trigger them all at the same time, in one single shot, and then exploit the colored shadows to extract depth edges.

Figure 3.22a shows our setup with three lights of different color: red, green and blue. When the lights are triggered at the same time, they sum to white light. We strategically placed the lights to produce shadows along all depth discontinuities in the scene: one is positioned below the camera, while the others are placed on the left and

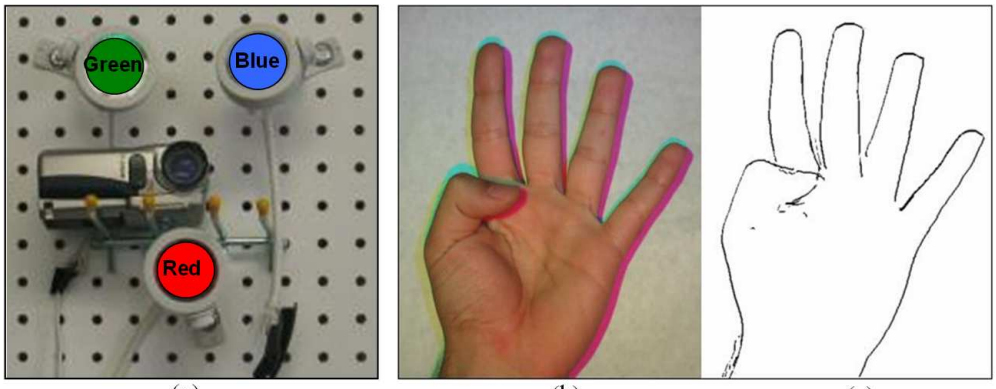


Figure 3.22: *(a) Our setup for dynamic scenes with different wavelength light sources. (b) Input image. Note the shadows with different colors. (c) Depth edge detection.*

right upper diagonals of the camera center of projection.

Our algorithm to detect depth edges using this setup follows a similar idea of the algorithm described in section 3.1. Given the input image with colored shadows, we need to first distinguish which shadows were created by which light source. With this information, we can traverse the image along the correspondent epipolar rays, marking depth edges at shadows associated with correspondent light sources. In our setup, for the lights placed along the camera diagonal, the traversal is not aligned with the pixel grid. For efficiency, we may keep the traversal along the pixel grid, but detecting negative transitions with e.g., steerable kernels [35] tuned to specific directions.

A simple way to distinguish the shadows created by each light would be to decompose the input image into the red, green and blue channels. However, finding shadows in each channel is not an easy problem. In this case, the maximum composite image of the three channels is not suitable for computing a shadow-free image, due to the fact that non-shadowed regions of each channel may have different intensities, depending

on the albedo of the objects.

In the following section we will describe an approach to segment shadows using a reference image of the scene, captured with white light sources. This method does not solve the motion problem, but considerably reduces acquisition time, allowing motion compensation algorithms to work better. Then, in section 3.4.2, we rely on shadow detection from a single image to extract depth edges in dynamic scenes. Although separating shadow edges from reflectance edges is a difficult problem for general scenes, we show that our technique can be useful for specific applications, such as extracting internal hand contours or lip segmentation from video.

3.4.1 Using a Reference Image

The basic idea of our method based on a reference image is to take two pictures of the scene, one with the three red, green, and blue lights triggered at the same time, and the other with white light sources. At non-shadowed regions, the ratio between the images should be close to one, due to the fact that red, green and blue sum to white light. The ratio in each channel can be used to distinguish which shadows were created by which light source.

We now describe this idea in more detail. First consider the RGB color ρ_k, $k = R, G, B$ formed at a particular pixel, for illumination with spectral power distribution $E(\lambda)$ impinging on a surface with spectral reflectance function $S(\lambda)$. If the three camera sensor sensitivity functions form a set $Q_k(\lambda)$, $k = R, G, B$, then we have:

$$\rho_k = \sigma \int E(\lambda)S(\lambda)Q_k(\lambda)\, d\lambda, \quad k = R, G, B \tag{3.8}$$

where σ is Lambertian shading, i.e., the inner product between lighting direction and

surface normal at a particular surface point. Assuming that camera sensitivities are Dirac delta functions [32], $Q_k(\lambda) = q_k\delta(\lambda - \lambda_k)$, then equation 3.8 is simplified:

$$\rho_k = \sigma E(\lambda_k)S(\lambda_k), \quad k = R, G, B \tag{3.9}$$

The capture process of our technique consists in first taking an image I_{color} of the scene with three light sources red, green, and blue triggered at the same time. Then, we replace the colored lights with white lights of same intensity and capture a reference image I_{white}. The white light sources could be placed near the colored lights (to avoid replacing the lights), provided that the scene depth is sufficiently large when compared to the baseline between camera and lights. We assume the two images are properly registered or the scene is static between the shots. Note that we keep all imaging parameters constant, except the spectral power distribution of the light sources. By taking the ratio between each channel of I_{color} and I_{white}, we have:

$$S_k = \frac{I_{color_k}}{I_{white_k}} = \frac{\sigma E_{color}(\lambda_k)S(\lambda_k)q_k}{\sigma E_{white}(\lambda_k)S(\lambda_k)q_k} = \frac{E_{color}(\lambda_k)}{E_{white}(\lambda_k)}, \quad k = R, G, B \tag{3.10}$$

where E_{color} and E_{white} correspond to the combined spectral distribution of the three colored and white light sources, respectively. As we can see from the equation above, the reflectance term is canceled, which is important for detecting shadows without depending on the albedo of objects in the scene.

Given that the light sources may have different intensities, we convert I_{color} and I_{white} to a chromatic space before taking the ratio in equation 3.10. More specifically, we define $I'_{color_k} = \frac{I_{color_k}}{\sum_{i=R,G,B} I_{color_i}}$ and $I'_{white_k} = \frac{I_{white_k}}{\sum_{i=R,G,B} I_{white_i}}$ for $k = R, G, B$. These intensity normalized images are shown in Figures 3.23c and 3.23d. The ratio $\frac{I'_{color}}{I'_{white}}$ is

shown in Figure 3.23e. Ratio values greater than one are clamped to one. Note that shadows can be easily segmented in this image for depth edge detection. The reason is that at non-shadowed regions, the ratio between the images is close to one, due to the fact that red, green and blue sum to white light and thus $E_{color} \approx E_{white}$. On the other hand, at shadowed regions, at least one of the light sources does not illuminate the local region, implying a drop in the ratio image due to the different spectral distribution of E_{color} and E_{white}.

For red, green and blue lights, we used 50W Ushio Popstar MR-16 Halogen light bulbs, with 12° beam angle spread. For white light sources, we used three 50W Ushio Whitestar MR-16 Halogen light bulbs, also with 12° beam angle spread. The presence of some artifacts in Figure 3.23d are mostly due to the narrow beam angle spread of light sources. Since the lights are not precisely calibrated, parts of the scene may have more incidence of light sources at specific wavelengths. This may create spurious transitions on the ratio image, leading to false shadow segmentation (such as along the horse foot).

3.4.2 Learning Shadow Color Transitions

We now turn to the problem of detecting depth edges with a single image, captured with red, green and blue light sources triggered at the same time, as shown in Figure 3.22a. In this case, detection of depth edges can be achieved by using algorithms that detect shadows [33] or separate illumination from reflectance using a single image [112].

Finlayson et al. [33] proposed a method to remove shadows from images by deriving a 1D illumination invariant image representation based on log-chromaticity coor-

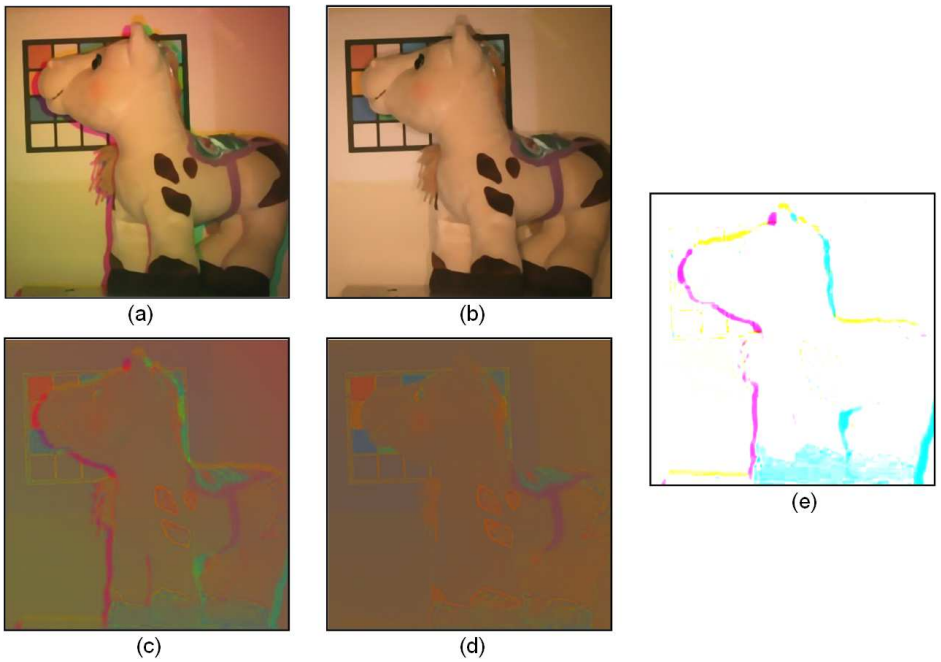


Figure 3.23: *(a) Image I_{color} taken with red, green and blue light sources. (b) Image I_{white} taken with white light sources. (c) Conversion to chromatic space: I'_{color} (d) I'_{white} (e) ratio between I'_{color} and I'_{white}. The color of the segmented shadows indicates which light source corresponds to each shadow.*

dinates. Tappen et al. [112] use color information and a classifier trained to recognize gray-scale patterns in order to classify image derivatives as being caused by reflectance or illumination changes. The Retinex algorithm [61] addresses the same problem, but relying on the assumption that the gradients along reflectance edges have larger magnitude compared to those caused by illumination variation.

Although these algorithms could be exploited to segment shadows in general scenes, we have implemented a simpler method that relies on learning the color transitions between object and shadows. This allows us to distinguish which light source created a specific shadow. More specifically, in a training stage we collect sample pixels along the shadow transitions and project a support vector machine classifier for each light

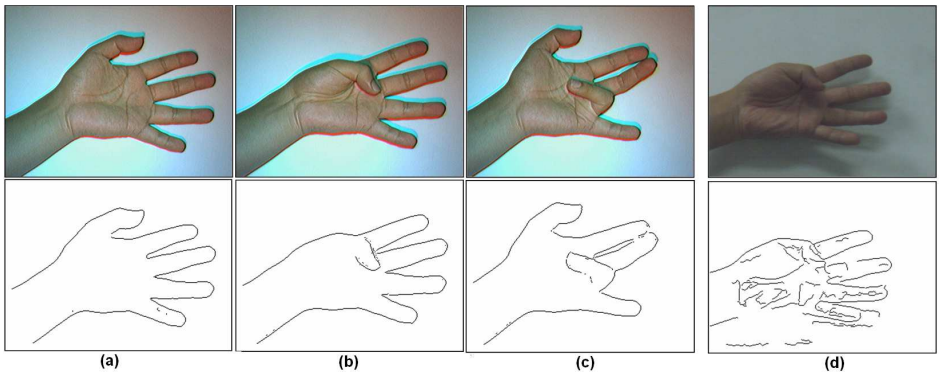


Figure 3.24: *(a)-(c) Sample frames of a video sequence and correspondent depth edge detection. (d) Comparison with Canny edge detection.*

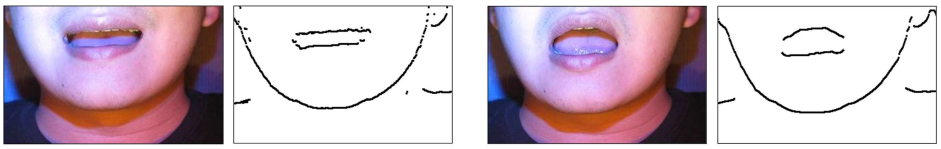


Figure 3.25: *Lip contour extraction using two red and blue lights placed above and below the camera.*

source. During the epipolar ray traversal, we use the output of the correspondent classifier to mark depth edges. Anisotropic diffusion [77] was applied as pre-processing for noise filtering.

Our method may generate false positives along reflectance edges with similar learned color transitions. Depending on the background albedo, shadow color transitions may differ from the learned model, thus leading to false negatives. Despite these limitations, our technique can be useful for diferent applications. For example, extracting depth edges due to finger occlusions is extremely important for hand gesture analysis and recognition. In fact, we have recently demonstrated that knowlege about occluding edges in the hand significantly improves recognition rate over standard intensity edges [31]. Other examples include extraction of lip contours and interior edges of the ear for

recognition.

Figures 3.22b and c shows an input image and our final result, respectively. Note that we are able to capture the self-occluding thumb finger edge, while eliminating texture edges such as wrinkles in the hand. This would not be possible with intensity edge detectors. Although background clutter could be a source of noise for our method, the hand could be extracted using standard segmentation techniques (e.g., based on skin color or background subtraction), while applying our technique just to extract interior occluding contours.

When video sequences are available, we use space-time consistency to improve depth edge detection. We basically consider the depth edge frames as a 3D surface, filling out edge gaps among frames to ensure surface smoothness. Figure 3.24 shows sample frames of a video sequence captured with our light sources with different wavelength. We compare our results with intensity edges detected with the Canny operator. Note that Canny edges include undesirable texture edges, such as wrinkles and nails, while missing important self-occluding edges due to low intensity variations.

Another example is shown in Figure 3.25, using only two red and blue lights, placed below and above the camera, respectively. Note that we are able to reliably detect the upper and lower lip contours, while reducing noise inherent in intensity edge detection. Lip contour extraction would not be possible if the mouth is closed, but still detecting whether the mouth is open or not could be useful for speech analysis or facial expression recognition. Only two lights are sufficient for this example, since lip contour edges are mostly horizontal. We also extract the facial contour, which is a challenging task for intensity edge detectors due to the low contrast variation between the face and neck area.

3.4.3 Discussion

In our previous work [85], we have addressed the problem of detecting depth edges in motion using light sources triggered in a rapid cyclic sequence, syncronized with the camera video frames. However, this method assumes that motion is monotonic with small magnitude, which may cause spurious edges for thin structures or objects with high frequency texture in the scene.

Most approaches that use active colored illumination in computer vision aim to recover the full 3D information of the scene through structured lighting. Tajima and Iwakawa [108] use a rainbow pattern for pixel coding and triangulation. Zhang et al. [124] also project a pattern of alternating colors in the scene and use a multi-pass dynamic programming algorithm to solve the correspondence problem in active stereo. Sa et al. [89] adopt color coding along projected colored stripe boundaries in time, which is useful for 3D reconstruction from dynamic scenes. Compared to these techniques, our approach offers the advantage of being simple, inexpensive, and easily built into a self-contained device. It could also complement existing stereo techniques to obtain depth edge preserving 3D reconstruction.

Limitations

Our method is not suitable for distant objects or outdoor scenes, where the intensity of the light sources may be insufficient compared to the sun light. Objects very close to the camera may suffer from pixel saturation, also violating the assumption that the scene depth is significantly larger than the camera-light baseline. Another problem occurs when the color of the shadow is the same as the background. For example, if a background pixel has color yellow and is not illuminated by the blue light source,

an yellow shadow (formed by the combination of red and green lights) would not be detectable.

Our technique based on a single shot capture using lights with different wavelength (section 3.4.2) is limited to handle general scenes, as false negatives and positives may arise due to the albedo of objects in the scene. Research on shadow segmentation and intrinsic image computation from a single frame [112, 33] is very important to achieve a more general solution to the problem. We also believe that solutions involving new camera setups would be possible. For example, infra-red lighting could be used (with different wavelength light sources) to capture an image with shadows, while using another camera at the same viewpoint (this is possible using a beamsplitter) to capture an ambient image simultaneously. This ambient image could be used as reference to segment shadows more reliably.

Chapter 4

Varying Viewpoint: Depth Edge Preserving Stereo

Stereo vision algorithms have been investigated for many years in computer vision as a technique to infer 3D structure from images captured with different viewpoints. The most challenging problem in stereo reconstruction is the establishment of visual correspondence among images. This is a fundamental operation that is the starting point of most geometric algorithms for 3D shape reconstruction and motion estimation.

Intuitively, a complete solution to the correspondence problem would produce the following:

- A mapping between pixels in different images where there is a correspondence, and
- Labels for scene points that are not visible from all views – where there is *no correspondence*.

In the past two decades, intense interest in the correspondence problem has produced many excellent algorithms for solving the first half of the problem. With a few exceptions, most algorithms for dense correspondence do not address occlusions explicitly [93]. The occlusion problem is difficult partly because distinguishing image intensity variations caused by surface geometry from those caused by reflectance changes remains a fundamental unsolved vision problem [68].

A promising method for addressing the occlusion problem is to use active illumination. In fact, many techniques that make use of lighting changes have been proposed to solve the correspondence problem in stereo reconstruction [124, 22, 127]. In general these techniques offer a tradeoff between accuracy and cost of the equipment (and other issues like compactness, light source calibration, and number of images to be acquired).

In this chapter, we combine lighting with viewpoint variation to produce high quality disparity maps. Differently from most approaches in active stereo that use large baseline light sources, our method uses small baseline multi-flash illumination in order to acquire important cues, including: depth edges, the sign of the depth edge (which indicates the side of the foreground object), and information about object relative distances.

Using these cues, we show how to produce rich feature maps for 3D reconstruction. We start by deriving a qualitative depth map from a single multi-flash camera. In a multiview setup, we show how binocular half-occluded pixels can be explicitly and reliably labeled, along with depth edges. We demonstrate how the feature maps can be used by incorporating them into two different dense stereo correspondence algorithms, the first based on local search and the second based on belief propagation.

4.1 Qualitative Depth Map

In this section, we formulate a qualitative depth map using a single multi-flash camera. Our method is related to shape from shadow techniques [21], but differs significantly in methodology. At this point we are not interested in quantitative depth measurements. Rather, we want to segment the scene, while simultaneously establishing object depth-order relations and approximate relative distances. This turns out to be a valuable prior information for stereo. Two key important mesurements are used to construct our qualitative depth map:

- The sign of the depth edge, which indicates which side of the edge corresponds to foreground.
- The shadow width, which encodes object relative distances.

4.1.1 Sign of Depth Edge

The sign of each depth edge pixel may be computed easily based on the depth edge detection algorithm described in Section 3.1. At the negative transition along the epipolar ray in the ratio image, $R_{k,}$the side of the edge with higher intensity is the foreground and lower intensity (corresponding to shadowed region) is the background.

Figure 4.1 illustrates this idea. Note that the sign is associated with each depth edge pixel, and not to an image region.

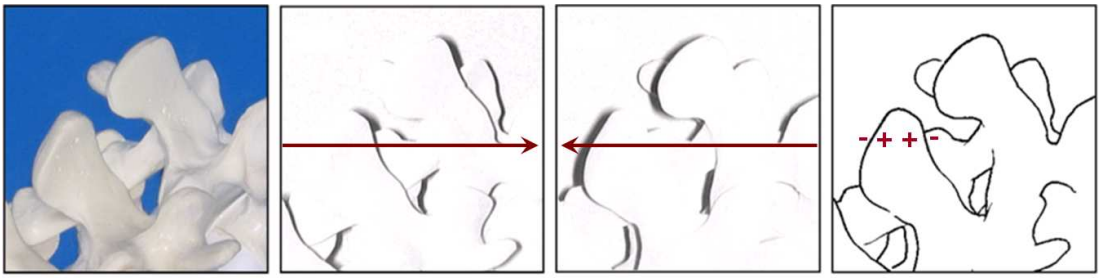


Figure 4.1: *From left to right: original image, left flash ratio image, right flash ratio image, signed edges.*

4.1.2 Shadow Width Estimation

A natural way of extending our depth edge detection method to estimate shadow width is to measure the length of regions delimited by a negative transition (which corresponds to the depth edge) and a positive transition along the epipolar ray in the ratio images. However, finding the positive transition is not an easy task, due to interreflections and the use of a non-point light source.

Figure 4.2a-c illustrates this problem: note that the intensity profile along the vertical scanline depicted in the ratio image has spurious transitions due to interreflections and a smooth transition near the end of the shadow. Estimation of the shadow width based on local-area-based edge filtering leads to unrealiable results. In contrast, we take advantage of the global shadow information. We apply the mean-shift segmentation algorithm [18] in the ratio image to segment the shadows, allowing accurate shadow width estimation (see Figure 4.2d).

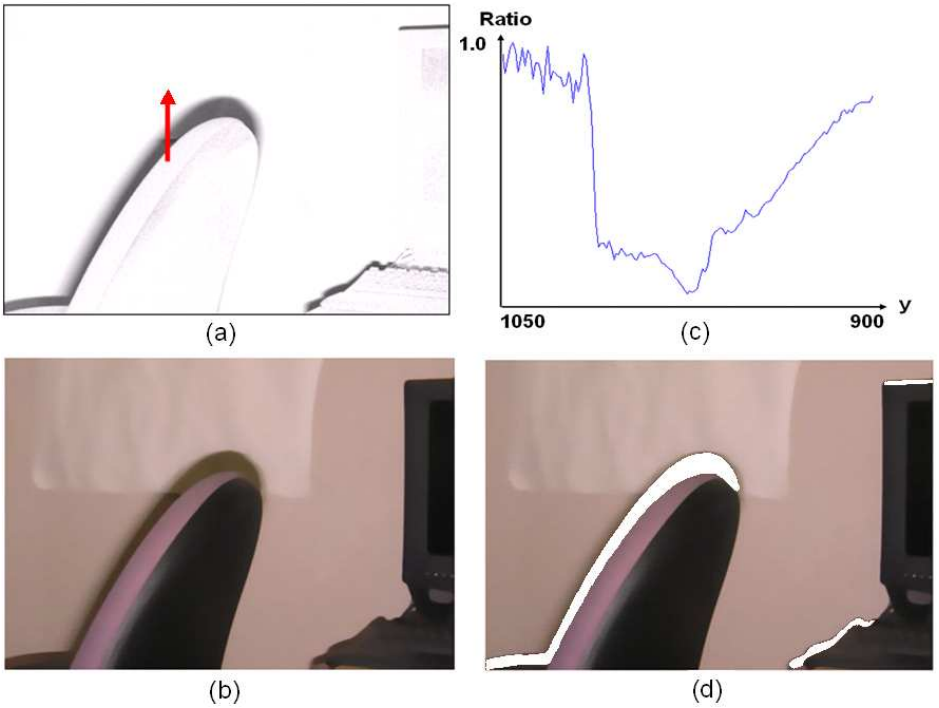


Figure 4.2: *(a) Ratio Image. (b) Original Image. (c) Intensity plot along the vertical scanline depicted in (a). Note that there is no sharp positive transition. (d) Meanshift segmentation to detect shadow, shown in white color*.

4.1.3 Shadows and Relative Depth

As we discussed in Section 3.2.1, the shadow width can be computed as:

$$d = \frac{fB(z_2 - z_1)}{z_1 z_2} \tag{4.1}$$

where the variables involved are f (camera focal length), B (camera-flash baseline), z_1, z_2 (depths to the shadowing and shadowed edges). For now, assume that the background is flat and whose distance z_2 from the camera is known.

Working on this equation, we have:

$$\begin{aligned} \frac{dz_2}{fB} &= \frac{(z_2 - z_1)}{z_1} \\ \frac{dz_2}{fB} &= \frac{z_2}{z_1} - 1 \\ log(\frac{dz_2}{fB} + 1) &= log(\frac{z_2}{z_1} - 1 + 1) \\ log(\frac{dz_2}{fB} + 1) &= log(z_2) - log(z_1) \end{aligned} \tag{4.2}$$

Note that for each depth edge pixel, we can compute the left hand side of equation 4.2, which encodes the relative object distances (difference of log depth magnitudes). This allows us to create a gradient field that encodes sharp depth changes(with gradient zero everywhere except at depth discontinuities) and perform 2D integration of this gradient field to obtain a qualitative depth map of the scene. This idea is described with more details below.

4.1.4 Gradient Domain Solution

In order to construct a sharp depth gradient map, we need to know the direction of the gradient at each depth edge pixel. This information can be easily obtained through the sign of the depth edge pixel in each orientation, which tells us which part of the edge is the foreground and which part is the background.

Le E be the set of depth edge pixels and $G = (G_h, G_v)$ the sharp depth gradient map, where G_h and G_v correspond to its horizontal and vertical components, respectively, with:

$$\begin{aligned} G_h(x,y) &= 0 \text{ if } (x,y) \notin E \\ &= log(\frac{d_h(x,y)z_2}{fB} + 1)s_h(x,y) \text{ otherwise} \end{aligned} \tag{4.3}$$

where $s_h(x, y)$ is the sign $(-1, +1)$ of the depth edge pixel (x, y) and $d_h(x, y)$ is the shadow width along the horizontal direction. The component G_v is calculated in the same way as equation 4.3 for the vertical direction.

Our qualitative depth map can be obtained with the following steps:

- Compute the sharp depth gradient $G(x, y)$.
- Integrate G by determining M which minimizes $|\nabla M - G|$.
- Compute the qualitative depth map $Q = exp(M)$.

It is important to note that the gradient vector field G may not be integrable. In order to determine the image M, we use the same integration approach for specular reflection reduction in Section 3.3. The optimization problem to minimize $|\nabla M - G|^2$ is equivalent to solving the Poisson differential equation $\nabla^2 M$ = *div G*, which can be solved using the standard full multi-grid method. The final qualitative depth map is obtained by exponentiating M, since M contains the logarithm of the real depth values.

For many applications, the background may be not flat and its distance to the camera unknown. In this case, we can set $\frac{z_2}{FB}$ to 1.0. Now we cannot obtain the absolute distances from the background. Instead we get relative distances proportional to the shadow width and a qualitative depth map with segmented objects. We will show in section 4.3.2 that this is a very useful prior for stereo matching.

4.1.5 Synthetic Example

Figure 4.3 shows our qualitative depth map computation using synthetic images. We basically used as input four images with manually created shadows corresponding to the top, bottom, left and right flashes, as shown on the top of the figure. The resultant

qualitative depth map, as well as the correspondent 3D plot, are shown on the bottom of the figure. Note that the elevations of the rectangular areas are proportional to the associated length of shadows in the images.

4.1.6 Real Images

Figure 4.4 illustrates results obtained for the qualitative depth map computation from real images, using a single multi-flash camera. As we can see, our method effectively segments the scene, encoding object relative distances through the shadow width information. Note that the images have low intensity variation and small depth changes, a challenging scenario for most 3D reconstruction methods.

Our qualitative depth map also offers the advantage of creating a slope in intensity when there are gaps in the depth contours. Note in the hand image the smooth transition between the thumb finger and the palm of the hand. This is a useful property for setting smoothness constraints in stereo matching.

In Figure 4.5, we show a more complex example. The scene contains many depth discontinuities and specular reflections, which poses a serious problem for most 3D reconstruction methods. We used our specular mask described in Section 3.3 to eliminate spurious edges in the depth edge map. The qualitative depth map and the 3D plot are shown in Figures 4.5b-c.

Clearly, our method is not able to handle slanted surfaces or rounded objects, since the depth variation is smooth without a sharp discontinuity. This is not a problem if we use it as a prior for stereo reconstruction.

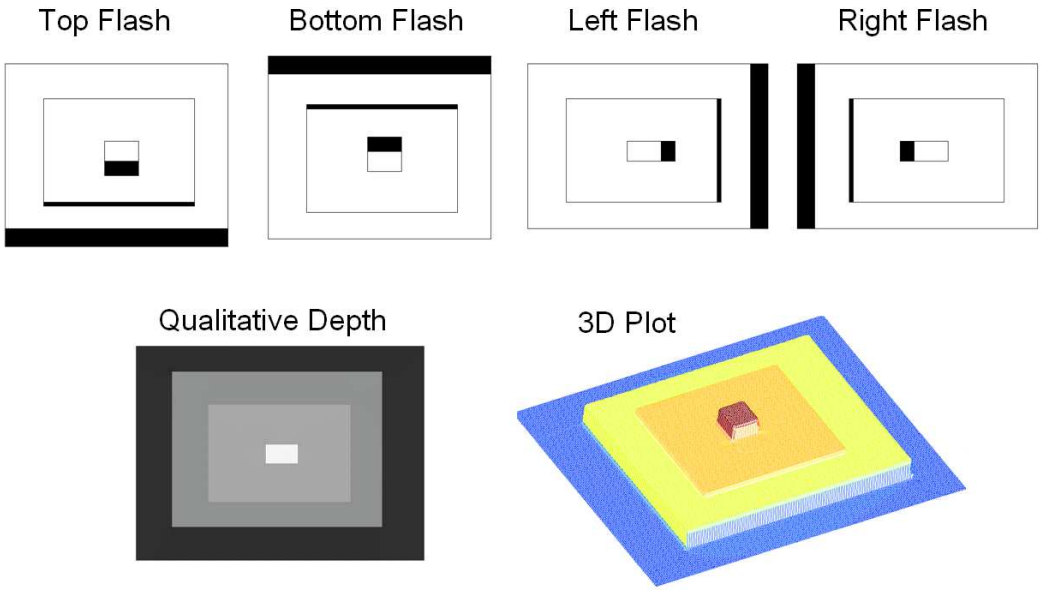


Figure 4.3: *Top: Synthetic images with manually created shadows corresponding to the top, bottom, left and right flashes. Bottom: Qualitative depth map and corresponding 3D plot.*

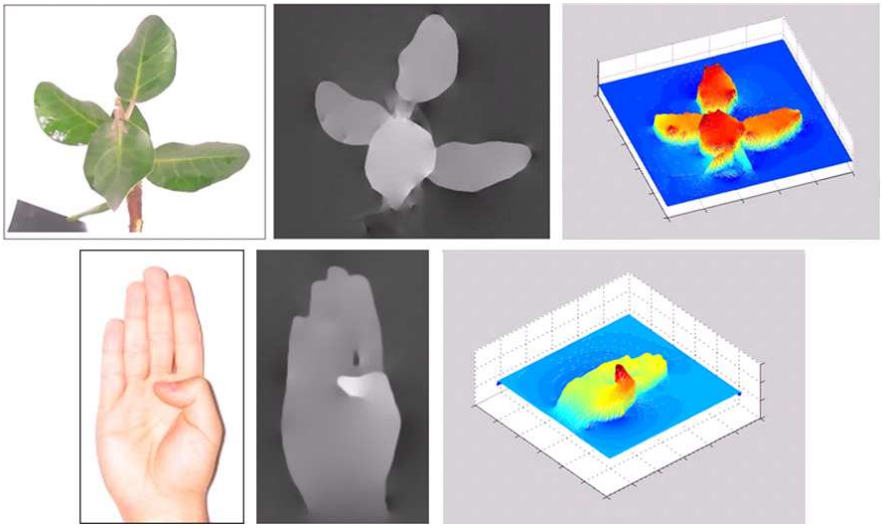


Figure 4.4: *From left to right: original image, qualitative depth map and the corresponding 3D plot. Note that our method captures small changes in depth and is robust in the presence of low intensity variations across depth contours.*

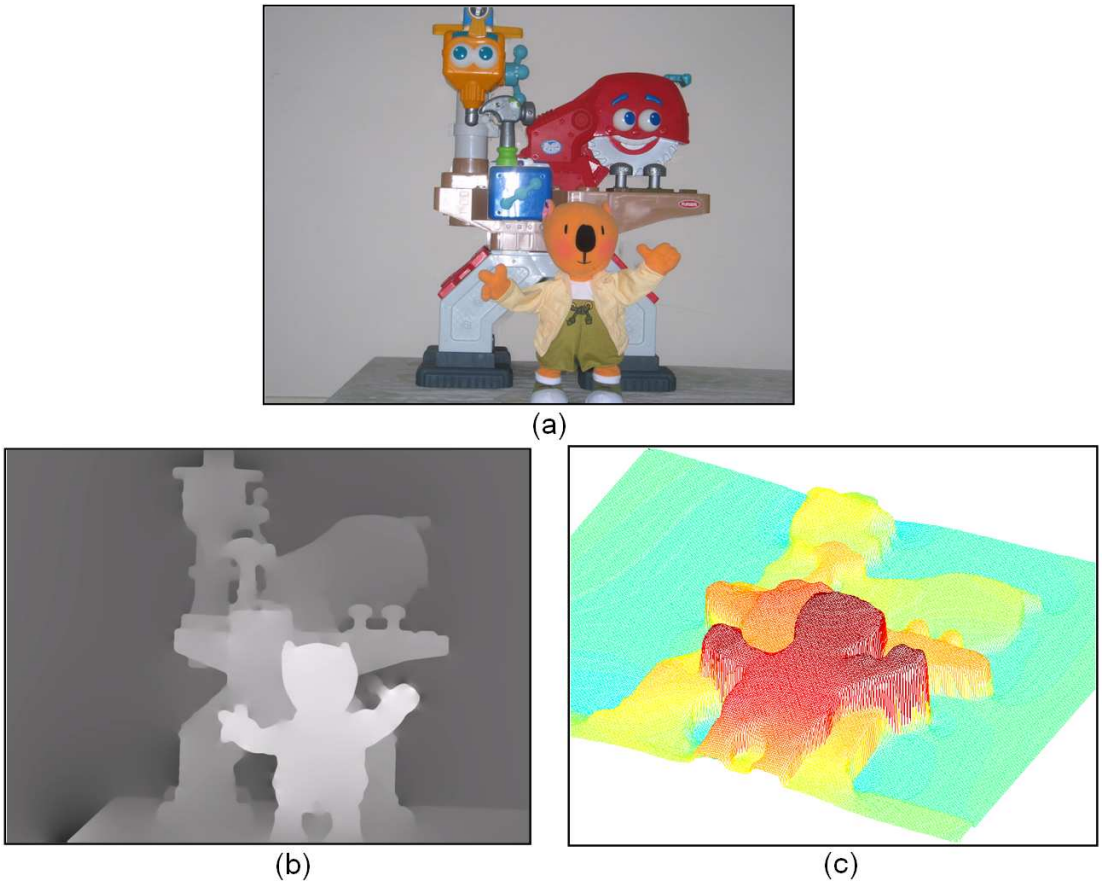


Figure 4.5: *(a) Complex scene with many depth discontinuities and specular reflections. (b) Qualitative depth map. (c) Corresponding 3D plot.*

4.2 Occlusion Detection

Binocular half-occlusion points are those that are visible in only one of the two views provided by a binocular imaging system [26]. They are a major source of error in stereo matching algorithms, due to the fact that half-occluded points have no correspondence in the other view, leading to false disparity estimation.

Current approaches to detect occlusion points are passive (see [26] for a comparison among five different techniques). They rely on the correspondence problem and thus are unable to produce accurate results for many real scenes. In general, these methods report a high rate of false positives and have problems to detect occlusions in areas of the scene dominated by low spatial frequency structure.

4.2.1 Occlusions Bounded by Shadows

Rather than relying on the hard correspondence problem, we exploit active lighting to detect binocular half-occlusions. Assume we have a stereo pair of cameras with horizontal parallax and light sources arranged as in Figure 4.6. By placing the light sources close to the center of projection of each camera, we can use the length of the shadows created by the lights surrounding the other camera to bound the half-occluded regions.

This idea is illustrated in Figure 4.6. Note that the half-occluded region S is bounded by the width of the shadows S_1 and S_2. Observing the figure, let I_{L_1}, I_{R_1} and I_{R_2} be the images taken by the left camera with light sources F_{L_1}, F_{R_1} and F_{R_2}, respectively. The width of S_1 and S_2 can be determined by applying the meanshift segmentation algorithm in the ratio images $\frac{I_{R_1}}{I_{L_1}}$ and $\frac{I_{R_2}}{I_{L_1}}$ (as described in section 4.1.2).

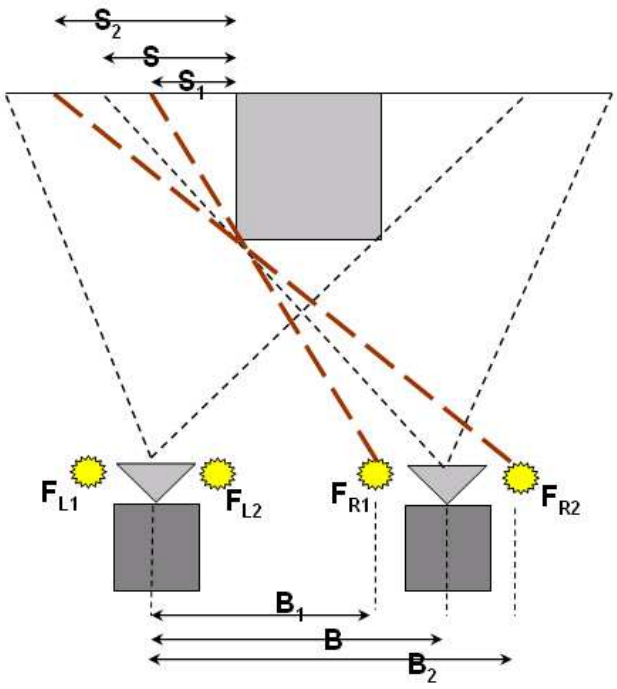


Figure 4.6: *The length of the half-occluded region is bounded by shadows created by flashes surrounding the other camera.*

We then determine the half-occluded region by averaging the shadowed regions: $S = \frac{B}{B_1+B_2}(S_1 + S_2)$, where B, B_1, and B_2 are the baselines of the camera and each light source, as shown in the figure.

The occluded region is determined with precision for planar shadowed region and with close approximation for non-planar shadowed region. In the non-planar case, the linear relationship between baseline and shadow width does not hold, but the length of the occluded region is guaranteed to be bounded by the shadows.

We could also use Helmholtz stereopsis [127] by exchanging the position of a multi-flash camera with a light source. The shadowed region caused by the light source in this configuration would denote exactly the half-occluded region. However, the de-

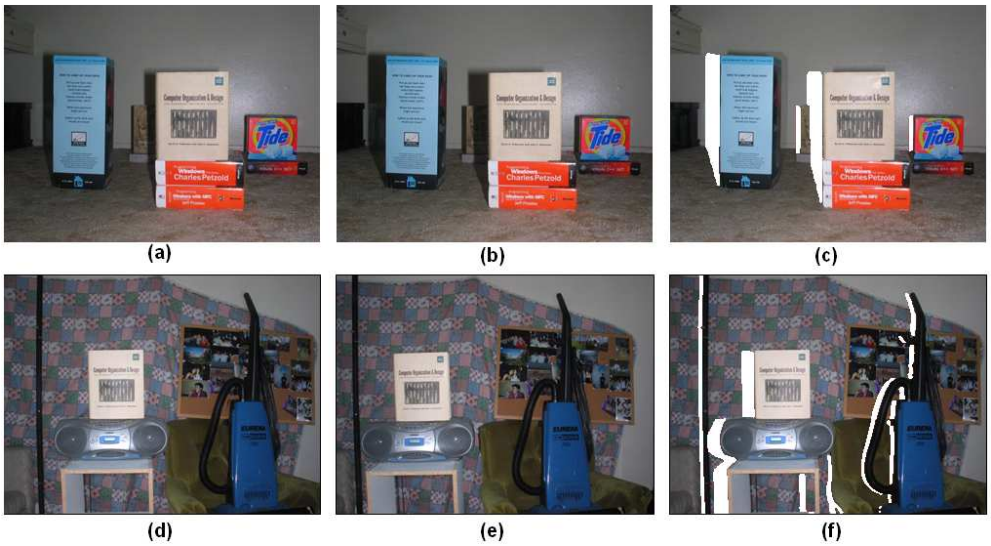


Figure 4.7: *Detection of binocular half-occlusions in both textured and textureless regions. (a)-(b) Images taken with light sources surrounding the other camera. (c) Our occlusion detection result marked as white pixels.* 0.65% *of false positives and* 0.12% *of false negatives were reported. (d) Left view. (e) Right view. (f) Occlusion detection (white pixels).*

vice swapping needs precise calibration and would be difficult to implement as a self-contained device.

We used two Canon G3 cameras with light sources arranged as Figure 4.6 to test our half-occlusion detection algorithm. Figure 4.7 demonstrates the reliable performance of our method. The images contain occlusion points in both textured and textureless regions, which is a challenging problem for passive algorithms that rely on pixel correspondence. For quantitative evaluation, we selected a piecewise planar scene (Figure 4.7a-c), since it is easier to obtain the occlusion ground truth (computed from the known disparity map). For this scene, our method reports 0.65% of false positives and 0.12% of false negatives. For very large depth differences our method may not give a precise estimation (for non-planar shadowed regions, due to larger bounded regions) and it might fail due to detached shadows with thin objects.

4.3 Enhanced Stereo Matching

In this section, we use our feature maps as prior information to enhance stereo matching algorithms. We start by demonstrating an enhanced window-based, local stereo method that takes advantage of depth edges and occlusions to produce disparity maps with very few computations and much more accuracy than traditional correlation-based methods. Then, we show how to incorporate our feature maps into global stereo methods based on Markov random field optimization. We also analyse different stereo implementation setups and scenes with specular reflections. Finally, we discuss limitations of our technique and compare with previous 3D reconstruction approaches.

4.3.1 Enhanced Local Stereo

A major challenge in local stereo is to produce accurate results near depth discontinuities. In such regions, the main assumption of local methods is violated: the same window (aggregation support) contains pixels that significantly differ in disparity, often causing serious errors in the matching process, due to perspective distortions. In addition, windows that include half-occluded points near depth discontinuities are another source of error, since they do not have correspondence in the other view.

The central problem of local methods is to determine the optimal size, shape, and weight distribution of the aggregation support for each pixel. There is a trade-off in choosing the window size: if the window is too small, a wrong match might be found due to ambiguities and noise. If the window is too large, problems due to foreshortening and depth discontinuities occur, with the result of lost detail and blurring of object boundaries. Previous solutions to this problem include the use of adaptive windows

[52] and shiftable windows [53], but producing clean results around depth discontinuities still remains a challenge.

Varying Window Size and Shape

We adopt a sliding window which varies in shape and size, according to depth edges and occlusion, to perform local correlation. Given the quality of the detection of depth edges and half-occluded points, results are significantly improved.

In order to determine the size and shape of the window for each pixel, we determine the set of pixels that has aproximatelly the same disparity as the center pixel of the window. This is achieved by a region growing algorithm (starting at the center pixel) which uses depth edges and half-occluded points as boundaries.

Only this set of pixels is then used for matching in the other view. The other pixels in the window are disconsidered, since they correspond to a different disparity.

Experiments

We first demonstrate the usefulness of depth edges in local stereo using the 640x480 Tsukuba stereo pair of the MiddleBury dataset (http://www.middlebury.edu/stereo). Figure 4.8a shows one of the stereo input images. The disparity ground truth for each pixel is shown in Figure 4.8b and the depth edge map computed from the ground truth is shown in Figure 4.8c. The results using a traditional correlation-based algorithm are shown in Figure 4.8d for a window size of 9x9 pixels and Figure 4.8e for a window size of 31x31 pixels. The trade-off in choosing the window size is clearly shown from these images: a smaller 9x9 window causes noisy results, while a larger 31x31 window causes significant errors near depth discontinuities. In order to verify the importance of

depth edges in local stereo, we used our algorithm considering as input the stereo pair and the depth edge map computed from the disparity ground truth. Figures 4.8f and 4.8g show our results for 9x9 and 31x31 window sizes, respectively. Clearly, the disparity map results are significantly improved near depth discontinuities. Note that this is a synthetic example to illustrate the effect of depth discontinuities in stereo, since we are assuming we have as input the depth edge map, which is difficult to obtain without active illumination.

Now we evaluate our method in a real scenario, using multi-flash imaging to compute depth edges and occlusions. We used a horizontal slide bar for acquiring stereo images with a multi-flash camera. Occlusions were estimated by moving the flashes properly to the shooting camera positions.

Figure 4.9a shows one of the views of a difficult scene we used as input. The image contains textureless regions, ambiguous patterns (e.g., the background close to the book), a geometrically complex object and thin structures. The resolution of the images is 640x480. We rectified them so that epipolar lines are aligned with horizontal scanlines. We adopted a small baseline between the cameras (maximum disparity equals 10), so that we can obtain a hand-labeled disparity ground truth (Figure 4.9b).

Figure 4.9c shows our computed depth edges and half-occluded points. Note that some edges do not appear in the ground truth (due to range resolution) and we also have some gaps in the edges due to noise. This data was considered to test our algorithms under noisy conditions.

Traditional local-correlation approaches perform very poorly in this scene, as we show in Figures 4.9d and 4.9e, using windows of size 9x9 and 31x31. In addition to noise, there are major problems at depth discontinuities - corners tend to become

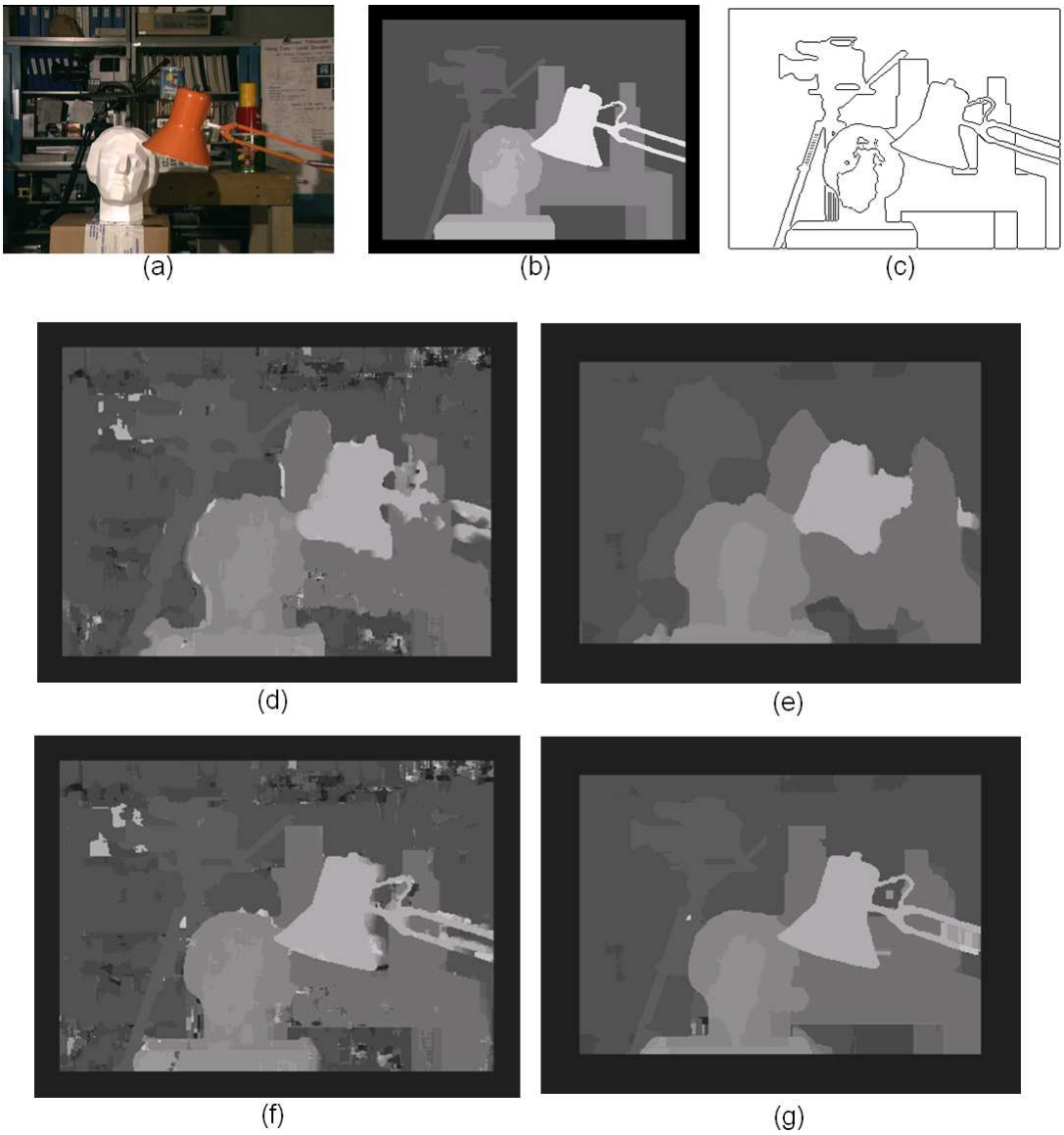


Figure 4.8: *(a) One image of the stereo pair. (b). Disparity map ground truth. (c) Depth edge map computed from the ground truth. (d) Local correlation result with a 9x9 window. (e) Local correlation result with a 31x31 window. (f) Our enhanced local stereo result with a 9x9 window. (g) Our enhanced local stereo result with a 31x31 window*.

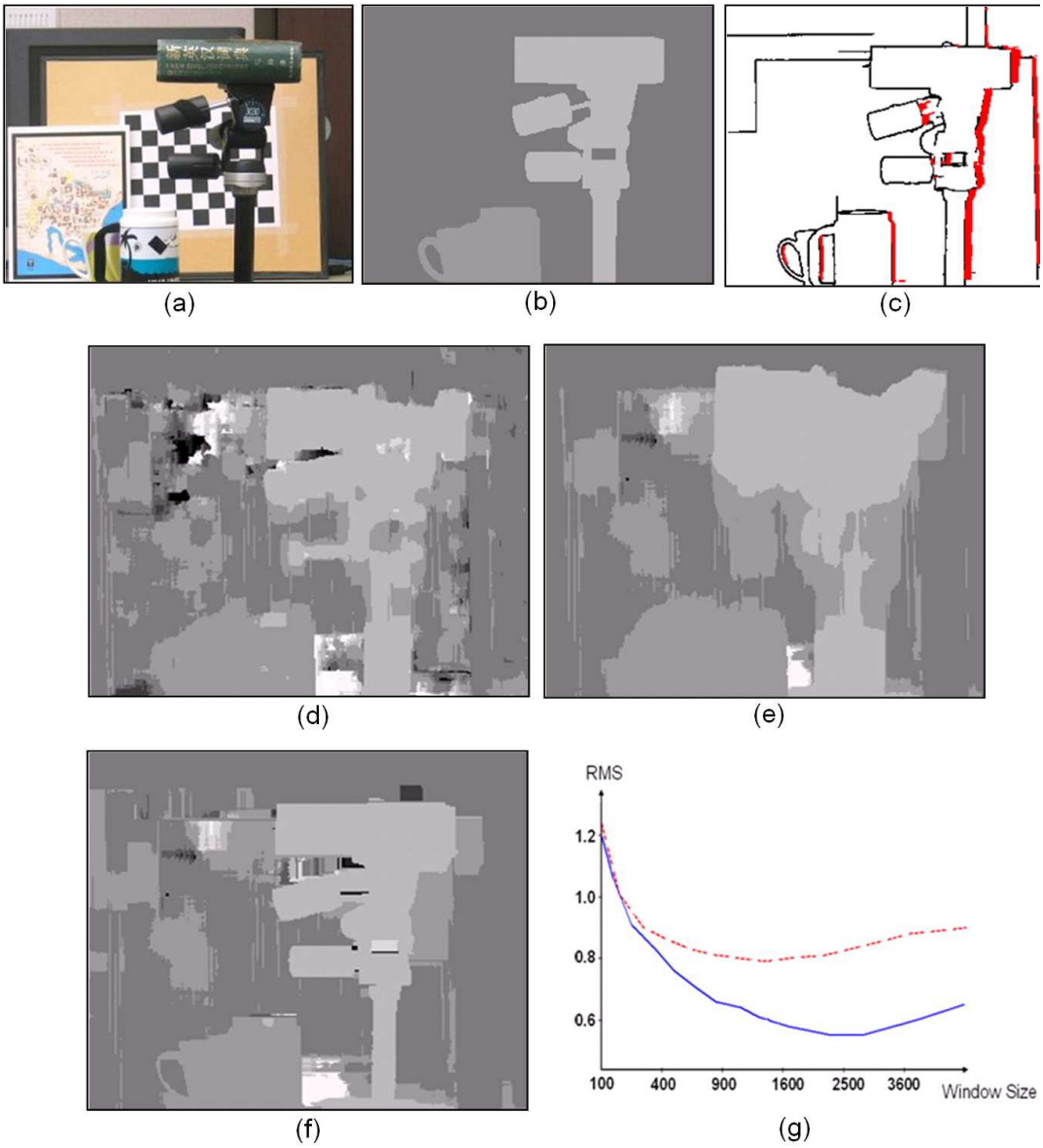


Figure 4.9: Enhanced Local Stereo *(a) Original image. (b) Hand-labeled ground truth. (c) Detection of depth edges and binocular half-occlusions. (d) Local correlation result with a 9x9 window. (e) Local correlation result with a 31x31 window. (f) Our multi-flash local stereo result with a 31x31 window. (g) Analysis of the root-mean-squared error with respect to window wize. The dashed line corresponds to traditional local correlation, while the solid line corresponds to our approach.*

rounded and thin structures often disappear or expand. In contrast, our method preserve discontinuities with large windows (Figure 4.9f). We show a quantitative analysis of the two methods with respect to the window size in Figure 4.9g. The axis of the graph correspond to the root-mean-squared error (RMS) and the window size in pixels. The error decreases significantly as the window grows for our method (solid line). At some point, it will start growing again with larger windows due to gaps in the depth edges. We could use our qualitative depth map here, but this would add an undesirable computational load, since local-based approaches are attractive because of their efficiency.

4.3.2 Enhanced Global Stereo

The best results achieved in stereo matching thus far are given by global stereo methods, particularly those based on belief propagation and graph cuts [56, 106]. These methods formulate the stereo matching problem as a maximum a posteriori Markov Random Field (MRF) problem. In this section, we will describe our enhanced global stereo method, which uses belief propagation for inference in the Markov network.

Some current approaches explicitly model occlusions and discontinuities in the disparity computation [5, 50], but they rely on intensity edges and junctions as cues for depth discontinuities. This poses a problem in low-contrast scenes and in images where object boundaries appear blurred. However, we want to suppress smoothness constraints only at occluding edges, not at texture or illumination edges. Our method makes use of the prior information to circumvent these problems, including the qualitative depth map and the automatically detected binocular half-occlusions described earlier.

Inference by Belief Propagation

The stereo matching problem can be formulated as a MRF with hidden variables $\{x_s\}$, corresponding to the disparity of each pixel, and observed variables $\{y_s\}$, corresponding to the matching cost (often based on intensity differences) at specific disparities. By denoting $X = \{x_s\}$ and $Y = \{y_s\}$, the posterior $P(X|Y)$ can be factorized as:

$$P(X|Y) \propto \prod_s \psi_s(x_s, y_s) \prod_s \prod_{t \in N(s)} \psi_{st}(x_s, x_t) \tag{4.4}$$

where $N(s)$ represents a neighborhood of s, ψ_{st} is called the compatibility matrix between nodes x_s and x_t (smoothness term), and $\psi_s(x_s, y_s)$ is called the local evidence for node x_s, which is the observation probability $p(y_s|x_s)$ (data term). The belief propagation algorithm gives an efficient approximate solution in this Markov network [106].

Qualitative Depth as Evidence

We can potentially use our computed depth edges to suppress smoothness constraints during optimization. However, the depth contours may have gaps. Fortunately, our qualitative depth image shows a desirable slope in intensity when gaps occur (as we will show in our experiments), and hence it is a good choice to set the compatibility matrix ψ_{st}. In addition, the qualitative depth map encodes the object relative distances via the shadow width information, and we use the map to encourage discontinuities at a certain disparity difference.

Let P be the qualitative depth scaled to match the set of possible disparities d_i, $i = 1..L$. We define $\psi_{st}(x_s, x_t) = C^{st}_{LxL}$, where C^{st}_{ij} is defined as:

$$C^{st}_{ij} = exp(-\frac{|d_i - d_j - \Delta P_{st}|}{F}) \tag{4.5}$$

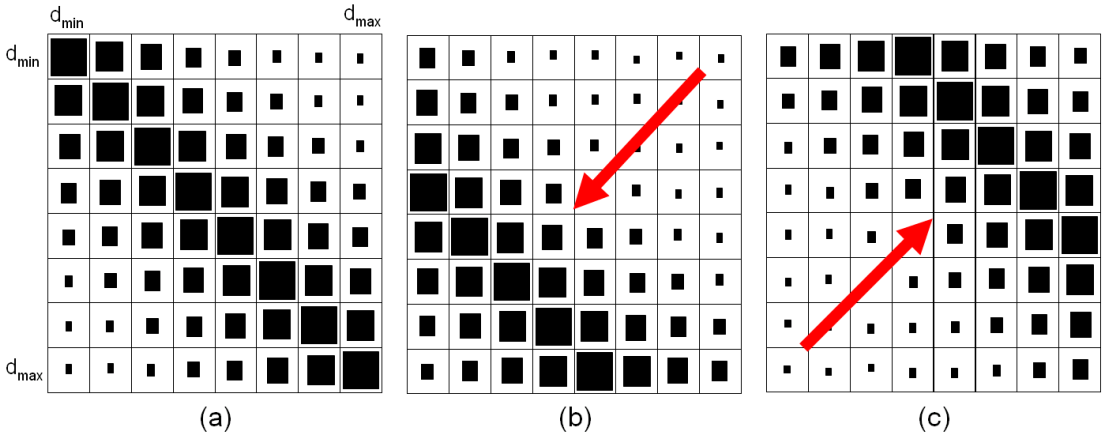


Figure 4.10: *(a) Compatibility matrix encouraging pixels to have the same disparity. Larger rectangles correspond to larger values. (b) Compatibility matrix encouraging neighboring pixels to have different disparities according to the qualitative depth map. (c) Same as (b), but considering a different sign of the depth edge so that the shift goes on the opposite direction.*

where ΔP_{st} is the intensity difference between pixels s and t in the qualitative map (which was scaled to match possible disparities) and F is a constant scaling factor. Intuitively, if $\Delta P_{st} = 0$, there is no sharp discontinuity for neighboring pixels s and t and the compatibility matrix will have larger values along its diagonal (see Figure 4.10a), encouraging neighboring pixels to have the same disparity. In contrast, if $\Delta P_{st} \neq 0$, the larger values will be shifted to the disparity encoded by ΔP_{st} (see Figures 4.10b-c). The direction of this shift depends on the sign of ΔP_{st}, which is the sign of the correspondent depth edge.

We have also included the half-occlusion information in our method. Nodes correspondent to pixels that have no match in the other view are eliminated, while a penalty is given for matching a given pixel with an occluded point in the other view.

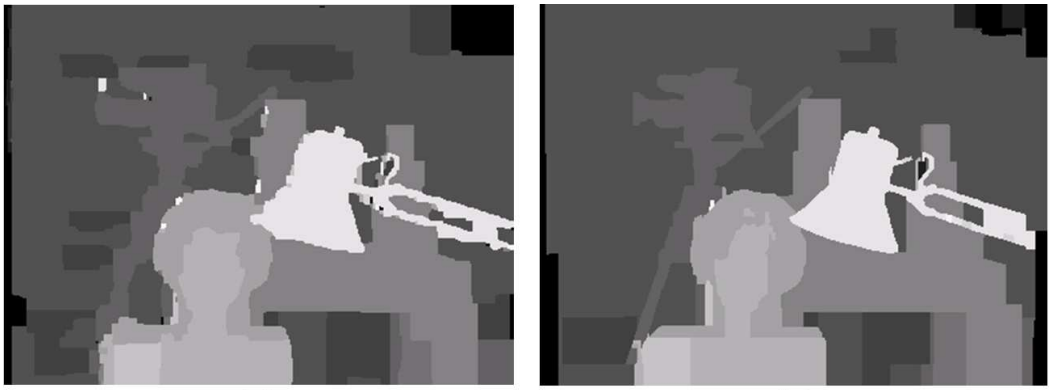


Figure 4.11: *(a) Standard belief propagation result. (b) Our enhanced global stereo method, given the knowledge of depth discontinuities*.

Experiments

Figure 4.11 shows a comparison of our algorithm with traditional global stereo based on belief propagation. As before, we used the input images from the Middlebury dataset with depth edges computed from the disparity map ground truth. For this example, we have not used the information from occlusions and qualitative depth; we just used depth edges to stop smoothness constraints in the energy function. As we can see, results are considerably improved near depth discontinuities.

The computed qualitative map in our challenging stereo example is shown in Figure 4.12a. The results for the standard belief propagation algorithm and our enhanced method are shown in Figures 4.12b and 4.12c, respectively. The passive method fails to preserve discontinuities due to matching ambiguities (we used the implementation available at http://cat.middlebury.edu/stereo/ with different weight and penalty parameters). Black pixels mean noisy values (zero disparity). Our results clearly show significant improvements with a RMS of 0.4590 compared to 0.9589 for this input. It is important to note that (although we do not show in this scene) our method handles

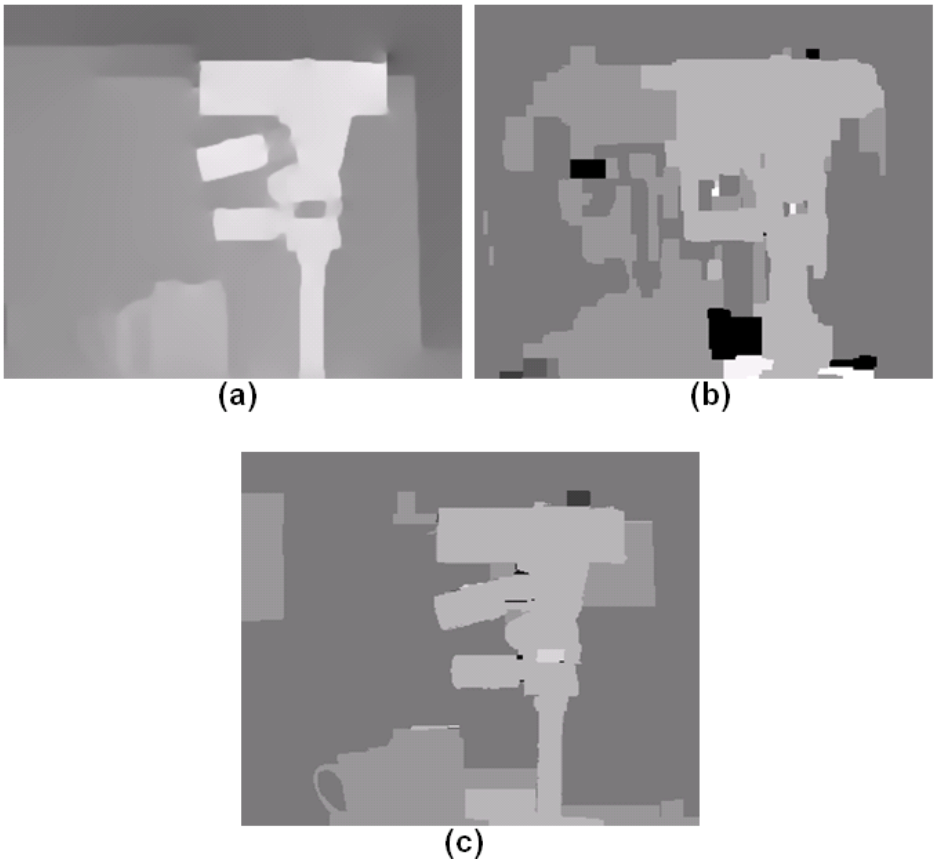


Figure 4.12: Enhanced Global Stereo *(a) Qualitative depth map. (b) Standard passive belief propagation result (RMS: 0.9589). (c) Our enhanced global stereo method (RMS: 0.4590).*

slanted surfaces exact in the same way as standard global methods. In other words, we do not sacrifice slanted surfaces to preserve discontinuities as opposed to [12].

4.3.3 Implementation Setups

As we mentioned before, we used a horizontal slide bar for acquiring stereo images with a multi-flash camera. Occlusions were estimated by moving the flashes properly to the shooting camera positions. This would be equivalent to using two multi-flash

cameras as shown in Figure 4.13a.

An alternative implementation setup is shown in Figure 4.13b. In this case, the flashes surround both cameras, and, for each flash, two images are captured simultaneously by the two cameras. This setup would be more appropriated to process dynamic scenes (using lights with different wavelength or triggered in a rapid cyclic sequence), which is not possible with our setup based on a slide bar. Compared to the setup showed in Figure 4.13a, it offers advantages in terms of acquisition time, while requiring only four light sources. On the other hand, occlusions can not be estimated reliably using our algorithm described in Section 4.2. Also, the top and bottom ratio image traversals for depth edge detection is not aligned with the pixel grid, since the top and bottom flashes are positioned on the upper and lower diagonals of the center of projection of the cameras.

Figure 4.13b shows an implementation setup that uses only one camera with a stereo adapter. With such adapter, it is possible to obtain the stereo image pair with a single shot, eliminating the need for camera synchronization. Experiments with this implementation setup are demonstrated next.

4.3.4 Specular Scenes

Specularities pose a problem for stereo matching, since they are viewpoint dependent and can cause large intensity differences at corresponding points. With multi-flash imaging, as we showed in Section 3.3, we can significantly reduce the effect of specular reflections in images, thus enhancing stereo correspondence near specular regions.

We used the setup shown in Figure 4.13c to capture four image pairs of a specular scene under different lighting conditions. Figures 4.14a and 4.14b show the stereo pair

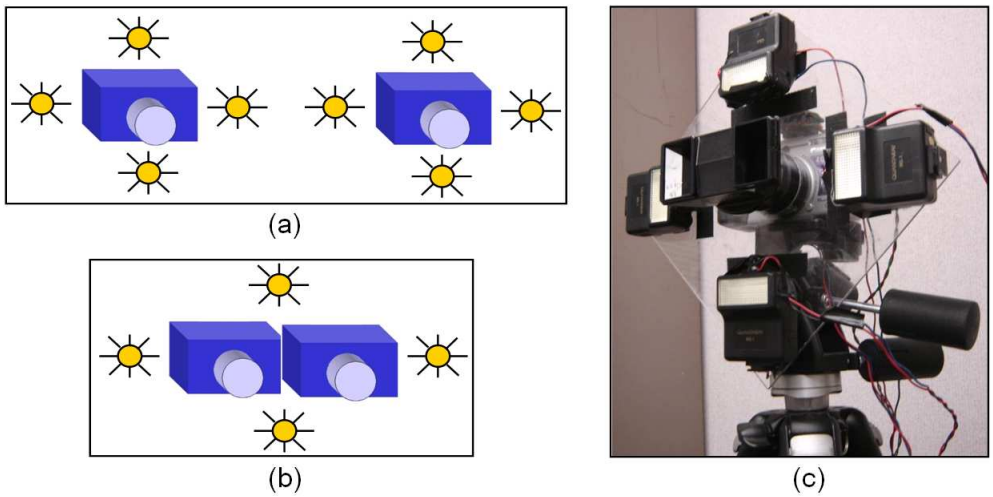


Figure 4.13: Different multi-flash stereo implementation setups. (a) Each camera with its own flashes. (b) Flashes surrounding both cameras. (c) Flashes surrounding only one camera with a Pentax stereo adapter.

(left view and right view, respectively), captured with one single shot, using the flash positioned to the right of the camera. Note how specularities are different in the two views.

Using the remaining flash images, we can attenuate the effect of specular reflections with our gradient-domain method described in Section 3.3. The specular-reduced image pair is shown in Figures 4.14c and 4.14d.

For stereo matching, we rectified the images and computed depth edges as preprocessing. Our enhanced local stereo matching was applied to both flash and specular-reduced image pairs, using a 31x31 search window. The disparity map results are shown for a region of interest in Figures 4.14e and 4.14f. Note that we are able to reduce artifacts due to specularities in the disparity map. The artifacts near the handle of the cup are due to partial occlusions, which were not detected and processed in this experiment.

Our method uses flash images to handle specularities. The detection and attenuation

of specular reflections in ambient (no-flash) images has been recently addressed by Agrawal et al. [4], using flash and no-flash image pairs. The advantage of using flash images is that they are less noisy and more appropriated for dark environments.

As we discussed in Section 3.3, when specular boundaries overlap in most images, we are not able to remove specularities. This is the reason why we still have specular artifacts in Figure 4.14f.

4.3.5 Efficiency

Our qualitative depth map takes about two seconds to compute on a Pentium IV 1.8 GHz for 640x480 resolution images. Our enhanced local-based stereo algorithm requires very few computations since depth edges can be computed extremely fast [85]. Our enhanced global method computation time is the sum of the time for the qualitative depth map computation plus the time for belief propagation procedure. We refer to [30] for an efficient implementation of the belief propagation algorithm.

4.4 Discussion

In addition to the proposed methods described in the previous section, other methods that take advantage of our feature maps (e.g., dynamic programming or segmentation-based stereo) could be explored. Signed depth edges could also be used as part of the matching cost computation. This would be very useful in low-contrast scenes, where occluding boundaries may not correspond to intensity edges. The disadvantage of matching depth edges is that problems may occur when a depth discontinuity in one view corresponds to a surface normal discontinuity in the other view.

Figure 4.14: (a) Left view of a flash image. (b) Right view of a flash image. (c) Left view of our specular-reduced image. (d) Right view of our specular-reduced image. (e) Disparity map for a region of interest using the flash image pair. (f) Disparity map using the specular-reduced image pair.

Small baseline multi-flash illumination could be used to enhance multiple view stereo algorithms for 3D object modeling [67, 96, 19]. We refer to the work of Crispell [20] along this direction, which shows the importance of depth edges and multi-flash photography to reconstruct objects with concavities. In our work, we applied our feature maps to aid the establishment of point correspondences between two images acquired with a pair of small baseline cameras.

We note that the shape of the background does not influence the detection of depth discontinuities. It does affect the qualitative depth map computation and occlusion detection (only along the shadowed region). In this case we get approximate heights and occluded areas, respectively, but this is not a problem since these feature maps are used as prior information in a Bayesian framework for stereo matching.

4.4.1 Comparison with other techniques

Table 4.1 shows a comparison of our multi-flash stereopsis approach with other stereo methods. Note that a small baseline flash setup means we do not need a laboratory setup as in photometric stereo and the cost and complexity of a flash attachment is very low. In addition, for non-intrusive applications, we can use readily available infra-red flash lighting, while projecting high frequency structured patterns requires an infra-red projector.

Below we give a more detailed discussion of the pros and cons of our method compared with stereo techniques:

	Recovered Information	Active / Passive	Handles Constant Albedo	Handles Depth Discontinuities	Compact, self-contained	Hardware Complexity / Cost
Structured Light	Depth	Active	Yes	Yes	Difficult	More
Photometric Stereo	Normals	Active	Yes	Limited	No	Less
Helmholtz Stereo	Depth+ Normals	Active	Yes	Limited	No	Less
Multi-Flash Stereo	Depth	Active	No	Yes	Yes	Less
Passive Stereo	Depth	Passive	No	Limited	Yes	Less

Table 4.1: *Comparison of our technique with other 3D reconstruction approaches.*

Passive Stereo

As we showed in the previous section, our method significantly enhances the establishment of point correspondences near depth discontinuities and specular highlights, when compared to passive stereo methods. Both techniques will fail in large textureless regions. Passive stereo methods are non-intrusive and more suitable for processing dynamic scenes. In outdoor scenarios, when sun light has more intensity than flash light, we can not enhance passive stereo matching.

Stereo Based on Structured Light

Active stereo techniques based on structured lighting produce more accurate correspondence maps than our approach. On the other hand, our method offers advantages in terms of low cost and portability. In addition, our feature maps could be used to enhance structured light techniques. Even state of the art 3D scanners may produce

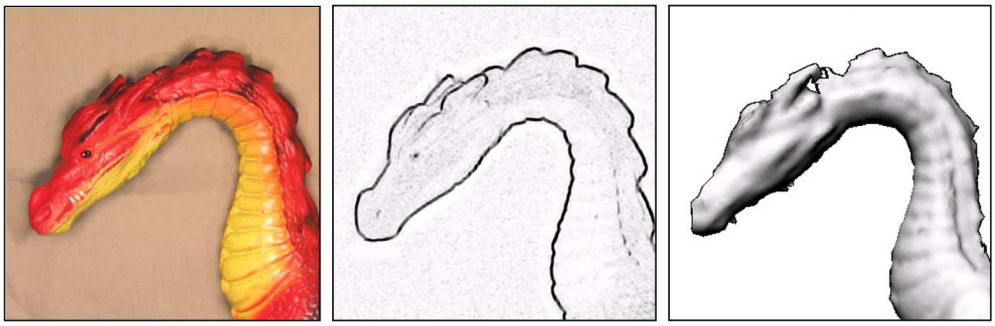


Figure 4.15: (a) Original Photo. (b) Our depth edge confidence map. (c) Depth map from active illumination 3Q scanner. Note the jagged edges.

jagged edges along depth discontinuities, as shown in Figure 4.15 [1].

Photometric Stereo

Photometric stereo techniques require a fixed lighting rig and thus are limited to industrial settings, contrasting with our method which can be built into a self-contained camera. They produce excellent results for smooth, Lambertian surfaces, but are unstable near depth discontinuities or rapid surface normal changes [91]. They offer the advantage of handling textureless regions and estimating surface reflectance properties.

Helmholtz Stereo

Helmholtz stereopsis has the ability to handle surfaces with arbitrary reflectance, in contrast to most previous methods that assume Lambertian reflectance. It also offers the advantage of estimating surface normals in textureless regions. In regions with texture, both depth and normals are estimated. Similarly to photometric stereo, light sources with large baseline are assumed to allow sufficient photometric variation across

[1] Thanks to Karhan Tan for providing the reconstruction using the 3Q scanner.

reciprocal image pairs, so that the normal field can be estimated. Hence, the setup is difficult to be built into a self-contained device. In addition, the camera and light source must be calibrated and moved in a precise and controlled fashion. Although the authors claim that shadows can be used in Helmholtz stereo as a cue for detection of partial occlusions, no experiments are reported for obtaining discontinuity preserving depth maps.

4.4.2 Limitations

Ou approach has the following limitations:

- Although we significantly enhance passive stereo matching near discontinuities and specularities, our method suffers from other well-known problems in passive matching, such as handling textureless regions, noise, and non-Lambertian surface reflectance. Some of these problems are addressed by active stereo approaches, as we mentioned in the last section. Our feature maps obtained with small baseline illumination could be used to enhance these active illumination stereo methods as well, as most of them are sensitive near depth discontinuities.

- Our method fails for outdoor scenarios when the sun light has more intensity than the flash lights. In this case, depth edges and occlusions can not be detected and used as prior information in stereo. For local stereo, our algorithm would be equivalent to traditional correlation-based approaches, since the window shape and size would keep constant along the image. For global stereo, we would have to use intensity edges in addition to the qualitative depth map to set smoothness constraints.

- Motion is another cause of failure in our approach. Without proper image registration, our feature maps can not be computed reliably. Possible solutions to this problem include the use of light sources with variable wavelength, as we discussed in section 3.4. However, reliably finding depth edges in motion in general scenes is still an open research problem.

Chapter 5

Comprehensible and Artistic Rendering

Traditional digital cameras are excellent for capturing the realism in a scene. However, the captured images may be insufficient for many applications where the goal is not physical realism. For instance, it is useful to synthesize images that intentionally look different from photographs for technical illustrations of geometrically complex scenes, such as mechanical parts and anatomical structures. In this case, the goal is to emphasize important features, while suppressing unnecessary detail, to produce comprehensible, easy to understand images. Another important application of non-photorealistic rendering (NPR) is to create stylized/artistic images, which may look like paintings or pencil portraits.

Creating comprehensible and stylized renderings from images, rather than 3D geometric models, has recently received a great deal of attention [25, 116]. Most of previous techniques involve processing a single image as the input, applying morpho-

logical operations, image segmentation and enhanced filters. Interactive techniques, such as rotoscoping, have been used as well, but our focus is to automate tasks where meticulous manual operation was previously required.

In this chapter, we show that depth discontinuities play an important role in non-photorealistic rendering, showing the application of our techniques in comprehensible rendering, medical imaging, and human facial illustrations. The idea of using depth edges for NPR started with Raskar et al. [85, 86] and the medical imaging analysis was carried out by Tan et al. [109]. Our contributions are mostly concerned with the sections related to tunable abstraction, integration with mean-shift segmentation edges, and human facial illustrations with a large camera-flash baseline.

5.1 Comprehensible Rendering

We now show that depth edges may be very useful for comprehensible rendering of low-contrast and geometrically complex scenes, such as mechanical parts, organic and anatomical structures. Real-world scenes are full of colors, texture, shadows and specularities. For technical illustrations, we aim to emphasize important features such as shape boundaries and reduce unimportant detail associated with clutter, less important colors, and texture. We provide a set of tools to accomplish this task based on depth edges.

A simple, yet quite effective way of creating easy to understand images is to simply increase the brightness of the input image and superimpose the captured depth edges. The resultant images obtained from this process, including anatomical, organic and mechanical examples, are showed in Figure 5.1. Notice the four spark plugs and the

dip-stick which are now clearly visible in the picture showing an engine of a car. Also, the bones of the skeleton and the leaves of the plant are clearly depicted. No intensity edge detector would capture these shape features in such low-contrast scenes.

5.1.1 Tunable Abstraction

Another way to reduce visual clutter in an image and emphasize object shape is to remove or simplify details not associated with the shape boundaries (depth edges) of the scene, such as texture and illumination variations. We provide a gradient domain tool for accomplishing this task. Our goal is to create large flat colored regions separated by strokes denoting important shape boundaries. Relevant details may be lost as the image is simplified, so tunable abstraction is needed The basic idea is to modify the gradient field of the input image by attenuating gradients not associated with depth edges. We define a mask image $M(x, y)$, where $M(x, y) = 1.0$ if (x, y) is a depth edge pixel and $M(x, y) = a$, otherwise, where $a, 0 \leq a \leq 1$, is a parameter that controls tunable abstraction. Image simplification is obtained through the following algorithm:

- Create a mask image $M(x, y)$
- Compute intensity gradient $\nabla I(x, y)$
- Modify masked gradients $G(x, y) = \nabla I(x, y) \times M(x, y)$
- Reconstruct image I' to minimize $|\nabla I' - G|^2$
- Normalize $I'(x, y)$ colors to closely match $I(x, y)$

The estimate of the intensity function I can be obtained by solving a Poisson differential equation, as we showed in our methods for specular reflection reduction and qualitative depth map computation. Figure 5.2 illustrates the tunable abstraction pro-

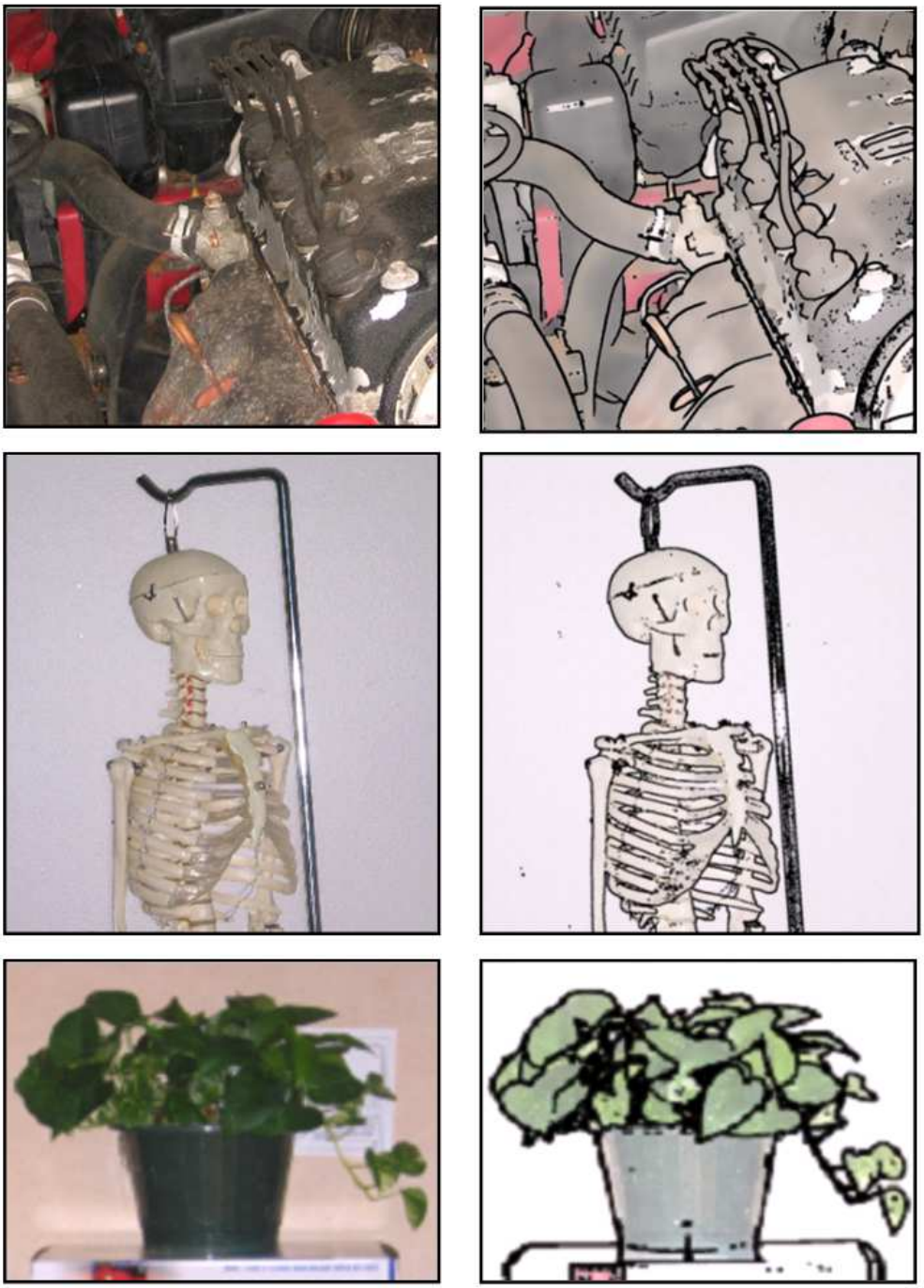


Figure 5.1: *(Left) Input images of geometrically complex scenes: mechanical, anatomical and organic examples. (Right) Comprehensible rendering using our technique.*

cess, where we render images using $a = 1$ (no attenuation), $a = 0.5$ and $a = 0$ (details not associated with depth edges are removed). After unnecessary detail is suppressed through our method, depth edges are superimposed in the resultant image.

5.1.2 Combining with Segmentation Edges

Despite the effectiveness of our texture de-emphasis algorithm for removing detail, we need to consider some issues that may arise depending on the scene. Depth edges may not form closed contours, thus leading to undesirable color bleeding of regions during image integration. In addition, important edges not related to depth edges (such as creases and relevant texture edges) are attenuated or removed with the present algorithm.

We provide a new tool, combining depth edges with mean-shift segmentation, to tackle these problems. Mean-shift segmentation [18] has been recently used for abstraction and stylization of photographs and video [25, 116]. However, general segmentation is still a problem far from being solved. Incorrect segmentation boundaries might cause undesirable edges to be enhanced in the image. Moreover, the level of detail is dependent on parameter tuning: a coarse segmentation may lose important fine detail, while a fine scale segmentation might lead to enhancement of unnecessary detail, also increasing the chances of incorrect segmentation boundaries (over-segmentation). DeCarlo and Santella [25] approached this problem using eye tracking information, to select the level of detail in different regions of the image.

Our method involves the following steps:

- Create edge map S by applying a coarse scale mean-shift segmentation on the

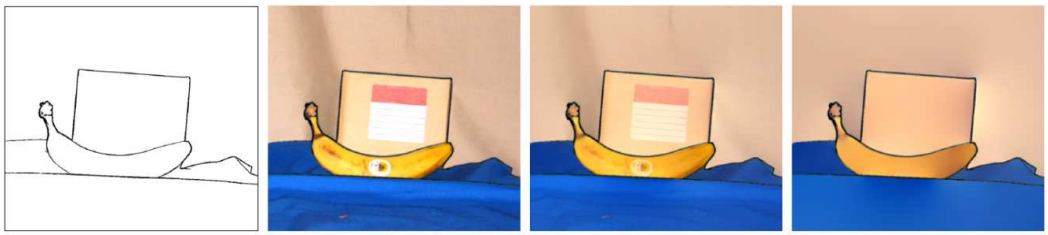


Figure 5.2: *Tunable abstraction. From left to right: depth edges and renderings with control parameter* $a = 1$, $a = 0.5$ *and* $a = 0$.

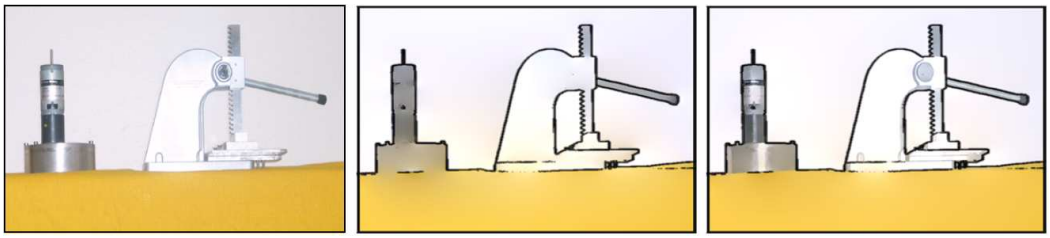


Figure 5.3: *From left to right: input image, texture de-emphasis based only on depth edges (notice the color bleeding) and our algorithm which combines depth edges with mean-shift segmentation.*

input image.

- Compute depth edge map D using our multi-flash technique.
- Let $A = S \cup D$, where A is an edge map containing both segmentation and depth edges.
- Attenuate gradients in the input image I not associated with edges in A and reconstruct I by solving a Poisson equation.
- Superimpose only depth edges D on simplified image I.

Notice that we use segmentation information only for helping color assignment, to avoid color bleeding. Figure 5.3 illustrates the result of our algorithm, comparing it with texture de-emphasis based only on depth edges.

5.2 Medical Imaging

In many medical applications like minimally invasive surgery with endoscopes, it is often difficult for the surgeon to visualize the 3D shape of the organs and tissues being examined. Our multi-flash imaging method captures additional shape information compared with traditional cameras and therefore has the potential to enhance visualization and documentation in surgery and pathology.

Application of enhanced shadow information to augment surgical perception has not been exploited previously. Shadows normally provide clues about shape, but with the circumferential ringlight illumination provided by most laparoscopes, this information is diminished. Similarly, the intense multi-source lighting used for open procedures tends to reduce strong shadow effects. Loss of shadow information may make it difficult to appreciate the shapes and boundaries of structures and thus more difficult to estimate their extent and size.

Our approach [109] is suitable to enhance traditional endoscopes, due to the fact that the light sources can be placed near to the camera. This allows compact designs that can be used in tightly constrained spaces, unlike many traditional 3D shape recovery methods where the imaging apparatus must be placed at large distances apart. Figure 5.4 illustrates a laryngeal endoscope enhanced with two lights which are turned on and off to detect depth edges.

Depth edges in motion can not be handled with our current setup. Also, a full evaluation with doctors assessing its usefulness compared to conventional endoscopic images is still needed.

Figure 5.4: *(Left) An enhanced endoscope with two lights. (Right) input image and image with depth edges superimposed.*

5.3 Human Facial Illustrations

We now turn to the problem of creating artistic images, similar to human drawings, with multi-flash imaging. In particular, we consider the problem of creating automatic human facial illustrations. Current methods in general are based on intensity edge detection techniques and thus are dependent on parameter settings, often requiring manual assistance to achieve the desired drawing [40].

Using our small baseline multi-flash illumination technique, we are not able to produce compelling human facial drawings due to the fact that important discontinuities not associated with depth discontinuities are not captured. Combining our method with segmentation or edge detection operators would also fail to capture important details that may not be associated with intensity variation (such as details along the hair), while possibly including unwanted edges (e.g., due to skin texture).

Ou approach consists in using a larger camera-flash baseline. Instead of looking for depth edges on the ratio images, we apply a Sobel edge detector operator (which keeps only negative transitions) on each ratio image and combine the outputs to obtain an

Figure 5.5: *Fully automatic human facial illustrations with a large camera-flash baseline setup.*

artistic illustration. Figure 5.5 (Top) shows an example. Due to the large camera-flash baseline, we obtain more shading variation and fine details that would not be captured with small baseline illumination. For instance, due to the complex hair reflectance, the ratio of large baseline images allows the detection of important hair edges, as showed in the example. Motion during image capture also contributes to detect edges due to material changes.

Our technique may also be combined with image deformation to construct caricatures, as shown in Figure 5.5 (Bottom). In this example, we detected edges using a large-baseline setup and superimposed the edges on the input image (with increased brightness). We have not seen any previous method that can achieve results of this

quality in a fully automatic way. For proper caricature drawings, facial features should be analysed in order to determine the appropriated deformation to be applied.

We imagine our method would be useful in low-bandwidth videoconferencing systems. Figure 5.6 shows black and white human facial illustrations obtained with our technique (Figure 5.6b), with a Sobel Operator (Figure 5.6c), and with Canny edge detection (Figure 5.6d). Clearly, our method captures more important details and facilitates recognition. Two-tone images consume significantly less storage, thus being suitable for rapid transmission over a network, for telecommunication.

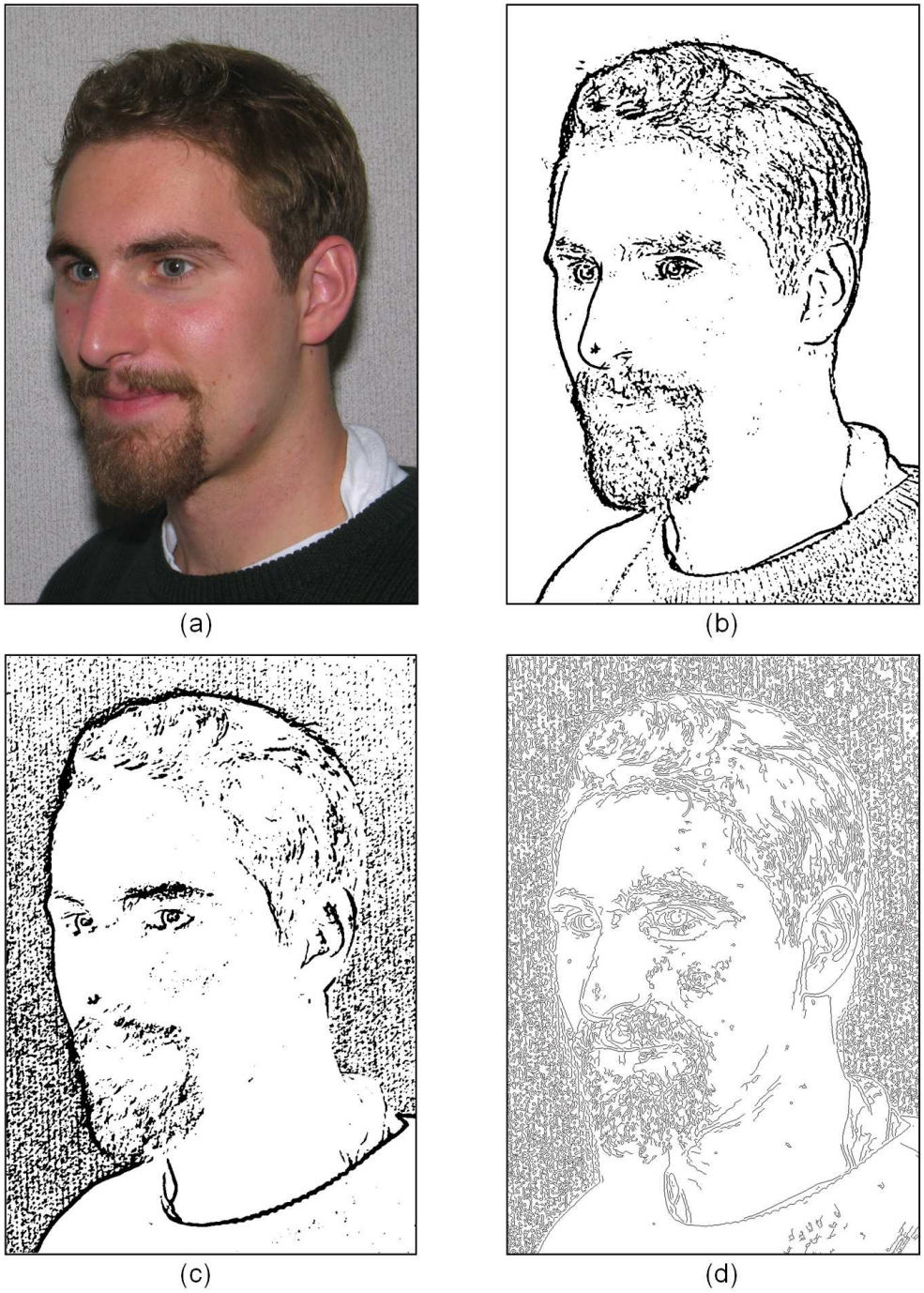


Figure 5.6: *Black and white illustrations. (a) Original photograph. (b) Our result with large-baseline multi-flash imaging. (c) The thresholded output of a Sobel operator. (d) Canny edge detection.*

Chapter 6

Exploiting Depth Discontinuities for Visual Recognition

In this chapter, we show the usefulness of depth discontinuities in visual recognition. In particular, we present a novel method for automatic fingerspelling recognition which is able to discriminate complex hand configurations with high amounts of finger occlusions. Such a scenario, while common in most fingerspelling alphabets, presents a challenge for vision methods due to the low intensity variation along important shape edges in the hand image. We demonstrate great improvement over methods that rely on features acquired by traditional edge detection and segmentation algorithms.

6.1 Vision-Based Fingerspelling Recognition

Sign language is the primary communication mode used by most deaf people. It consists of two major components: 1) word level sign vocabulary, where gestures are

used to communicate the most common words and 2) fingerspelling, where the fingers on a single hand are used to spell out more obscure words and proper nouns, letter by letter. Facial expressions can also be employed to distinguish statements, questions and directives.

Over the past decade, great effort has been made to develop systems capable of translating sign language into speech or text, aiming to facilitate the interaction between deaf and hearing people. Extensive research has been done in both word level and fingerspelling components.

Previous approaches to word level sign recognition rely heavily on statistical models such as Hidden Markov Models (HMMs) [102, 115, 17]. Excellent recognition rates were obtained for small word lexicons, but scalability is still an issue for glove-free sign recognition. For fingerspelling recognition, most successful approaches are based on instrumented gloves, which provide information about finger positions. Lamar and Bhuiyant [60] achieved letter recognition rates ranging from 70% to 93%, using colored gloves and neural networks. More recently, Rebollar et al. [88] used a more sophisticated glove to classify 21 out of 26 letters with 100% accuracy. The worst case, letter 'U', achieved 78% accuracy.

In general, non-intrusive vision-based methods, while useful for recognizing a small subset of convenient hand configurations [57, 8], are limited to discriminate configurations with high amounts of finger occlusions - a common scenario in most fingerspelling alphabets. In such cases, traditional edge detectors or segmentation algorithms fail to detect important internal edges along the hand shape (due to the low intensity variation in skin-color), while keeping edges due to nails and wrinkles, which may confound scene structure and the recognition process (see Figure 6.1b). Also, some

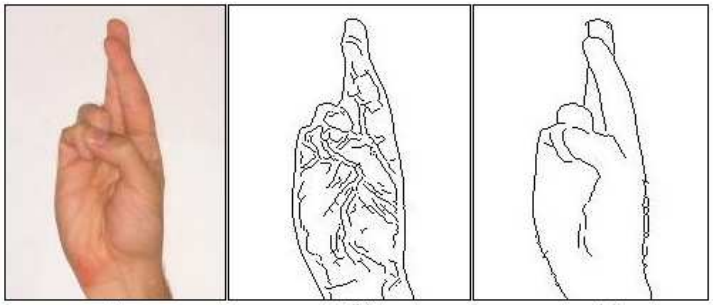


Figure 6.1: *(a) Letter 'R' in ASL alphabet. (b) Canny edges. Note that important internal edges are missing, while edges due to wrinkles and nails confound scene structure. (c) Depth edges obtained with our multi-flash technique.*

signs might look very similar to each other, with small differences on finger positions, thus posing a problem for appearance-based approaches [57].

We show that depth discontinuities may be used as a signature to reliably discriminate among complex hand configurations in the ASL alphabet (see Figure 6.1c), which would not be possible with current glove-free vision methods. For classification, we have used a shape descriptor similar in spirit to shape context matching [10], which is invariant with respect to image translation and scaling.

In Section 3.1 we demonstrated a reliable method for depth edge detection based on multi-flash imaging. Next we describe our shape descriptor and classification method.

6.2 Shape Descriptor and Classification

In this section, we present a shape descriptor for depth edges which is invariant with respect to image translation and scale. Our approach is simple and yet very effective. It has been recently evaluated on a large dataset for content-based image retrieval [71].

The basic idea is illustrated on Figure 6.2. For each edge pixel of interest, we first

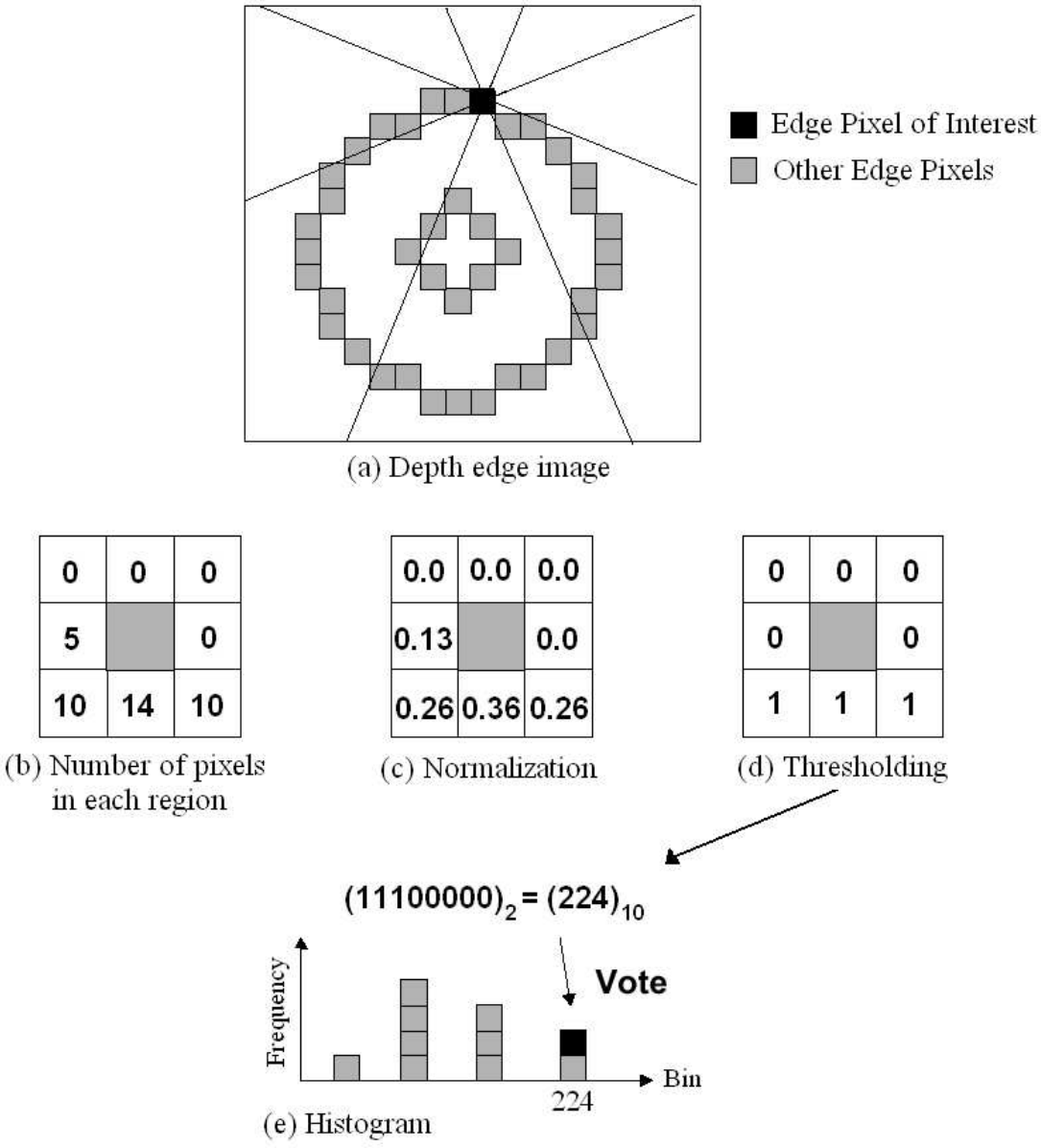


Figure 6.2: *Shape descriptor used for classification.*

analyze its context by counting the number of other edge pixels in eight neighboring regions, as shown in Figure 6.2(a). This gives us a vector of eight elements $C_i, 1 \leq i \leq 8$ (Figure 6.2b). We then normalize each element for scale invariance (Figure 6.2c) by denoting $S_i = C_i/C$, where $C = \sum_i^8 C_i$. Finally, thresholding is applied (Figure 6.2d), so that each element encodes the information of either high or low density of edge pixels along a specific direction of the pixel of interest. The threshold value 0.15 is obtained empirically.

Inspired by the concept of Local Binary Patterns [72] in the field of texture analysis, the values ”0”s and ”1”s are arranged counter-clockwise from a reference region (in our example, the bottom-right region) to express an 8-bit binary number. The correspondent decimal number $d, 0 \leq d \leq 255$ is used to vote for the respective bin in the histogram shown in Figure 6.2e. A 256-dimensional feature vector is then obtained by applying the above mentioned process to all edge pixels in the depth edge image.

Since the descriptor is based on the relative position of edge pixels, it is clear that it is invariant with respect to image translation. Scale invariance is obtained in the normalization step. The descriptor can also be made rotation invariant [71]. However, this may not be appropriated for some fingerspelling alphabets (e.g., Japanese Sign Language), which might have letters that are rotated versions of the others.

We have used a nearest-neighbor technique for classification. Initially, supervised learning is carried out by acquiring a set of images for each letter in the fingerspelling alphabet. Depth edges are then extracted and the shape descriptor technique is applied, so that a training database comprised of labeled 256-dimensional feature vectors is formed. Given a test image, features are extracted and the class of the best match training sample according to Euclidean distance is reported.

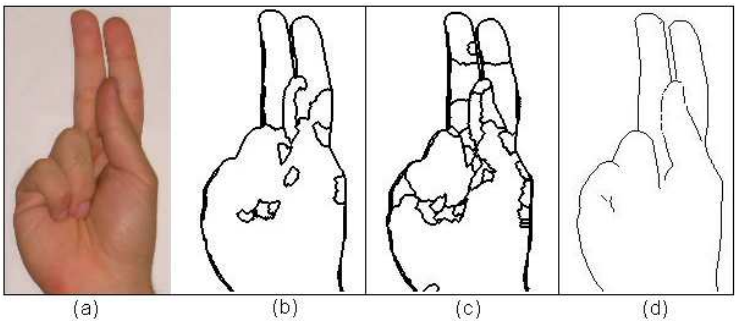


Figure 6.3: *(a) Letter 'K' of ASL alphabet. (b),(c) Mean-shift segmentation algorithm with different parameter settings. (d) Output of our method.*

6.3 Experiments

We compared the hand contours obtained using our technique with the output of a traditional Canny edge detector [16] and a state-of-the-art mean-shift segmentation algorithm [18]. We refer to Figure 6.1 for a comparison of our method with Canny edges. Changing parameter settings in the Canny algorithm could reduce the amount of clutter, but important edges along the hand shape would still not be detected. Figure 6.3 shows a comparison with mean-shift algorithm. Clearly, due to the low intensity skin-color variation in the inner hand region, the segmentation method is not able to detect important boundaries along depth discontinuities. Our method accurately locates depth edges and also offers the advantage that no parameter settings are required.

We realized that depth edges are good features to discriminate among signs of fingerspelling alphabets. Even when the signs look very similar (e.g., letters 'E','S' and 'O' in ASL alphabet), the depth edge signature is quite discriminative (see Figure 6.4). This poses an advantage over vision methods that rely on appearance or edge-based representations. Note that our method does not detect edges in finger boundaries with no depth discontinuity. It turns out that this is helpful to provide more unique signatures

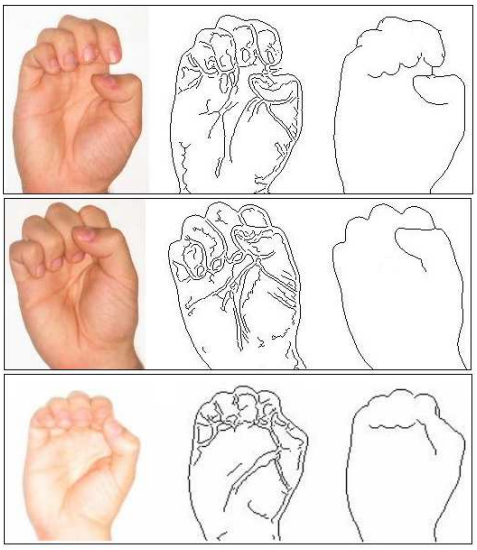


Figure 6.4: *From left to right: input image, Canny edges and depth edges. Note that our method misses finger boundaries due to the absence of depth discontinuities. This turns out to be helpful to provide unique signatures for each letter.*

for each letter.

In order to quantitatively evaluate the advantages of using depth edges as features for fingerspelling recognition, we considered an experiment with the complete ASL alphabet, except letters 'J' and 'Z', which require motion analysis to be discriminated. We collected a small set of 72 images using our multi-flash camera (three images per letter, taken at different times, with resolution 640x480). The images showed variations in scale, translation and slight variations in rotation. The background was plain, with no clutter, since our main objective is to show the importance of obtaining clean edges in the interior of the hand. It is worth mentioning that textured but flat/smooth backgrounds would not affect our method, but would make an edge detection approach (used for comparison) much more difficult.

For each image, features were extracted as described in sections 3.1 and 6.2. For sake of comparison, we also considered shape descriptors based on Canny edges.

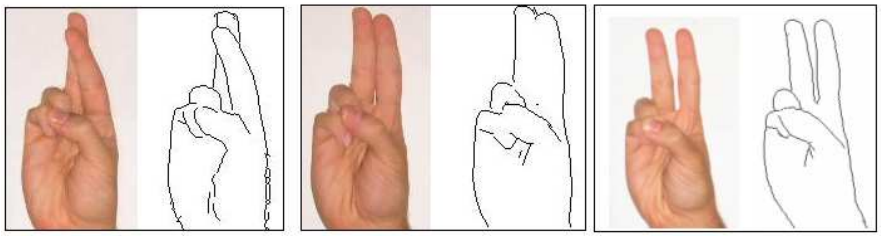


Figure 6.5: *Letters 'R', 'U' and 'V', the worst cases reported in [88]. Note that the use of a depth edge signature can easily discriminate them.*

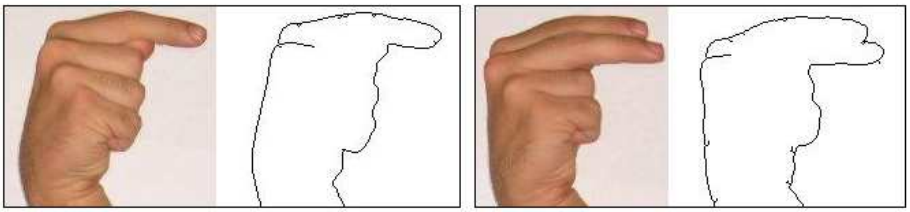


Figure 6.6: *A difficult case for traditional algorithms (letters 'G' and 'H'), where our method may also fail.*

Recognition rate was obtained using a leave-one-out scheme in the collected dataset. Our approach achieved 96% of correct matches, compared with 88% when using Canny edges.

Rebollar [88] mentioned in his work that letters 'R', 'U' and 'V' represented the worst cases, as their class distributions overlap significantly. Figure 6.5 shows these letters and their corresponding depth edge signatures. Note that they are easily discriminated with our technique. In the experiment described above, the method based on Canny edges fails to discriminate them.

Figure 6.6 shows a difficult case for traditional methods, where our method also fails to discriminate between letters 'G' and 'H'. In this particular case, we could make use of additional information, such as the intensity variation that happens between the index and the middle finger in letter 'H' and not 'G'.

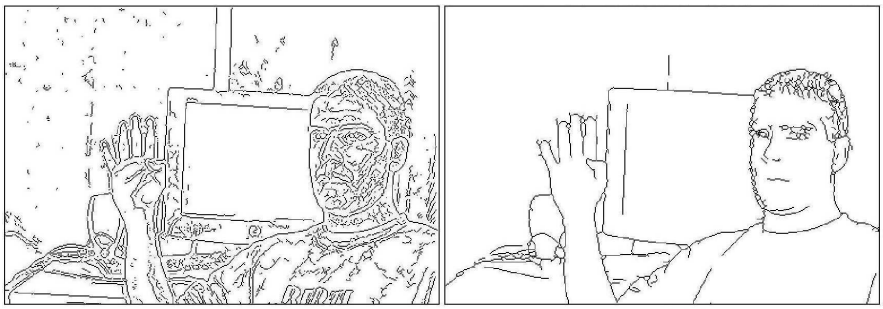


Figure 6.7: *(a) Canny edges (b) Depth edges. Note that our method considerably reduces the amount of clutter, while keeping important detail in the hand shape.*

All the images in our experiment were collected from the same person. A more complete evaluation would include a database with different signers. We believe that our method will better scale in this case, due to the fact that texture edges (e.g., wrinkles, freckles, veins) vary from person to person and are eliminated in our approach. Also, shape context descriptors [10] have proven useful for handling hand shape variation from different people. For cluttered scenes, our method would also offer the advantage of eliminating all texture edges, thus considerably reducing clutter (see Figure 6.7)

For segmented hand images with resolution 96x180, the computational time required to detect depth edges is 4ms on a Pentium IV 3GHz. The shape descriptor computation requires on average 16ms. Thus, our method is suitable for real-time processing. For improving hand segmentation, depth edges could be computed in the entire image. In this case, the processing time for 640x480 images is 77ms.

Our method could also be adapted for continuous sign recognition in video. We are currently exploiting a frequency division multiplexing scheme (Section 3.4), where flashes with different colors (wavelength) are triggered simultaneously. We hope this

will allow for efficient on-line tracking of depth edges in sign language analysis.

6.4 Discussion

The main difference of our shape descriptor from the one proposed by Belongie et al. [10] lies in the application of local binary patterns and the voting scheme. Hence we do not use the graph matching step (as in shape context matching), which makes the method simpler to implement and faster.

Shadows have already been exploited for gesture recognition and interactive applications. Segen and Kumar [95] describes a system which uses shadow information to track the user's hand in 3D. Leibe et al. [64] presented the concept of a *perceptive workbench*, where shadows caused by infrared lighting are exploited to estimate 3D hand position and pointing direction. Applications of their method include augmented reality gaming and terrain navigation.

These approaches consider light sources far away from the camera center of projection and casted shadows are separated from the objects. In contrast, our approach consider light sources with small baseline distance from the camera, allowing them to be built in a self-contained device, no larger than existing digital cameras.

We have not seen any previous technique that is able to precisely acquire depth discontinuities in complex hand configurations. In fact, most stereo methods for 3D reconstruction would fail in such scenarios, due to the textureless skin-color regions as well as low intensity variation along occluding edges.

Many exemplar-based [8] and model-based [65] approaches rely on edge features for hand analysis. We believe that the use of depth edges would lead to significant im-

provements in these methods. Word level sign language recognition could also benefit from our technique, due to the high amounts of occlusions involved. Flashes in our setup could be replaced by infrared lighting for user interactive applications.

We noticed that depth edges might appear or disappear with small changes in viewpoint (rotations in depth). This was in fact explored in the graphics community with the concept of *suggestive contours* [24]. We believe this may be a valuable cue for hand pose estimation [8].

A common thread in recent research on pose estimation involves using a 3D model to create a large set of exemplars undergoing variation in pose, as training data [97, 8]. Pose estimation is formulated as an image retrieval problem in this dataset. We could use a similar approach to handle out-of-plane hand rotations. In this case, a 3D hand model would be used to store a large set of depth edge signatures of hand configurations under different views. Another member of our lab is currently working on this problem.

Chapter 7

Conclusions

In this dissertation, we have addressed the problem of detection and modeling of depth discontinuities in computer vision. We showed that depth contours can be estimated reliably from real-world scenes and used as a positive source of information for image analysis algorithms, including stereo matching and visual recognition. Most previous methods have treated depth discontinuities as a source of noise in full 3D scene reconstruction.

7.1 Synopsis

Our research extended initial work on multi-flash imaging for depth edge detection [85] in different ways. First, we showed that by varying illumination parameters, such as the number, spatial position, type, and wavelength of light sources, we can detect depth discontinuities in a wider range of imaging conditions. Second, we showed that by combining viewpoint variation with multi-flash illumination, we can significantly

improve dense stereo matching near depth discontinuities. Finally, we demonstrated that depth edges can be useful in different computer vision and graphics applications, including non-photorealistic rendering, medical imaging, and visual recognition.

7.1.1 Varying Illumination Parameters

A **multi-baseline approach** was proposed to detect depth edges in different scales. By placing light sources at different baselines, and using algorithms to combine the set of images, we were able to detect depth edges associated with small and large changes in depth. For thin objects, false edges may be detected with large baseline light sources, due to detached shadows. We have investigated algorithms and novel setups to handle this problem. In particular, we compared the use of linear light sources with multiple point light sources, discussing their pros and cons.

We showed that specularities pose a problem for depth edge detection and proposed a gradient-domain method for **specular reflection reduction** in multi-flash imaging. Our approach consists in taking the median of gradients of the input images and then integrating the gradient field to obtain a specular-reduced image. We analyzed different cases according to the movement of specularities and showed that our method can be used to significantly reduce spurious edges in depth edge detection, when compared to the traditional algorithm based on the maximum composite of the input images.

Using **lights with diffent wavelength**, we proposed a method to detect depth edges in dynamic scenes. Our method works by triggering the light sources at the same time, while analyzing the color of shadows to detect depth edges. This is useful for specific applications (such as hand gesture interpretation), but the method is limited to handle general scenes, as the color of shadows depend on the albedo of objects in the scene.

For general scenes, we used a technique based on a reference image captured with white lights. Although this technique does not solve the motion problem, it reduces the acquisition time, allowing motion compensation algorithms to work better.

7.1.2 Varying Viewpoint

We combined viewpoint variation with small baseline multi-flash illumination to produce accurate stereo correspondence maps near depth discontinuities. Using a single multi-flash camera, we formulated a **qualitative depth map**, which is based on two important quantities: the sign of each depth edge pixel, which indicates which side of the edge is the foreground and which is the background, and the shadow width information, which encodes object relative distances.

In a multi-view setup, an **occlusion map** was proposed to label partial occluded regions in stereo, using the length of shadows created by the flashes. We then demonstrated **enhanced local and global stereo algorithms** that use these rich feature maps (qualitative depth and occlusion) as prior information. Compared with passive techniques, our method shows significant superior results in regions near depth discontinuities. Compared to previous active illumination approaches, our method offers advantages in terms of low cost and portability. Our feature maps could also be used to complement active lighting approaches, which may produce jagged depth edges.

7.1.3 Applications

Many applications can benefit from our proposed methods. The qualitative depth map can be used for **image segmentation**, while establishing object depth-order re-

lations. Following the work of Raskar [85], we have worked on **non-photorealistic rendering techniques**, including tunable abstraction, integration of depth edges with mean-shift segmentation edges, and human facial illustrations. We also showed depth edge detection in **medical imaging**, based on the work of Tan et al. [109]. Finally, we have demonstrated the importance of depth edges in **visual recognition**. In the problem of fingerspelling recognition, we showed a system that obtains significant higher recognition rate when compared to a system based on intensity edges.

7.1.4 Remarks

Many techniques presented in this dissertation (e.g., qualitative depth, enhanced stereo, fingerspelling recognition) are independent of the multi-flash camera setup. Signed depth edges could be computed using other techniques (independent of multi-flash imaging) and used as input for our methods.

There are cases where we need to know scene properties before applying our approach. For example, we need to know whether the scene contains or not a background in order to use our flash/no-flash technique to detect depth contours in a non-background scenario. Also, the choice for linear or point light sources (or even for using our multibaseline approach) is dependent on the task and scene complexity.

Except for the qualitative depth map computation and the global stereo matching approach, our techniques could be implemented in real-time. The basic multi-flash method for depth edge detection takes 77ms on a Pentium IV 3GHz, for images with resolution 640x480.

Finally, a more comprehensive evaluation, including more quantitative results, is necessary for assessing the performance of our methods. This is difficult and subjective

for some of our techniques (e.g., for non-photorealistic rendering results).

7.2 Future Work

There are a number of potential areas for future work involving discontinuity detection and modeling. Within our framework of varying lighting and viewpoint parameters, **novel imaging models** for depth edge detection and 3D photography could be exploited. For example, lights placed out of the camera plane could be used for faster acquisition in depth edge detection. The imaging geometry of two lights positioned in front and behind the camera along the optical axis (using a beamsplitter) would allow detection of depth edges with two shots.

Another example would be to consider setups with a light source surrounded by cameras, rather than a camera surrounded by light sources. This could facilitate the detection of depth edges in motion by keeping the light always on. The drawback is the need to solve the correspondence problem, but shadows would still be a useful cue for improving the matching through the detection of partial occlusions.

Outdoor scenes could be handled by using different imaging methods (such as radar or sonar). Another direction is to take advantage of the high number of images available on the web for the same location, in the same spirit of the *Photo Tourism* project [99].

As we mentioned in section 3.4, infrared lighting could be useful for detection of depth edges in dynamic scenes. Research on passive stereo techniques that are designed to detect discontinuities [12], rather than (or in addition to) the full disparity map, would allow the detection of depth edges in outdoor scenes, which is not possible with our approach.

Beyond illumination and viewpoint, the variation of other imaging parameters could be applied for detection and modeling of depth discontinuities. Variable focus has been recently exploited for silhouette extraction in image matting [69]. Coded exposure photography in conjunction with strobed lighting flashes could be used to analyze depth edges in motion, as noted by Raskar et al. [83].

Beyond depth edges, a new direction is to use imaging techniques to detect other physical discontinuities, including discontinuities in surface normal, illumination, motion and albedo. For example, similarly to depth edge detection, can we detect discontinuities in surface normal or motion without the full dense field estimation? How can we integrate different discontinuity detection modules for image analysis? The answer to these questions could lead to a framework for general scene understanding based on discontinuities.

Bibliography

[1] T. Adelson and J. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, 1992.

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[3] A. Agrawal, R. Raskar, and R. Chellappa. Edge suppression by gradient field transformation using cross-projection tensors. In *Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006.

[4] A. Agrawal, R. Raskar, S. Nayar, and Y. Li. Removing photography artifacts using gradient projection and flash-exposure sampling. *SIGGRAPH 2005 / ACM Transactions on Graphics*, 2005.

[5] M. Agrawal and L. Davis. Window-based, discontinuity preserving stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR'04)*, Washington, DC, 2004.

[6] D. Akers, F. Losasso, J. Klingner, M. Agrawala, J. Rick, and P. Hanrahan. Conveying shape and features with image-based relighting. In *IEEE Visualization*, 2003.

[7] N. Apostoloff and A. Fitzgibbon. Learning spatiotemporal t-junctions for occlusion detection. In *Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, California, 2005.

[8] V. Athitsos and Stan Sclaroff. Estimating 3D hand pose from a cluttered image. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, Madison, USA, 2003.

[9] P. Belhumeur and D. Mumford. A Bayesian treatment of the stereo correspondence problem using half-occluded regions. In *Conference on Computer Vision and Pattern Recognition (CVPR'92)*, pages 506–512, Champaign, Illinois, 1992.

[10] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[11] S. Birchfield. *Depth and Motion Discontinuities*. PhD thesis, Stanford University, 1999.

[12] S. Birchfield and C. Tomasi. Depth discontinuities by pixel-to-pixel stereo. *International Journal of Computer Vision*, 35(3):269–293, 1999.

[13] M. Black and D. Fleet. Probabilistic detection and tracking of motion discontinuities. In *International Conference on Computer Vision (ICCV'99)*, pages 551–558, Corfu, Greece, 1999.

[14] J. Bouguet and P. Perona. 3D photography on your desk. In *International Conference on Computer Vision (ICCV'98)*, Bombay, India, 1998.

[15] T. Boult and L. Wolff. Physically-based edge labeling. In *Conference on Computer Vision and Pattern Recognition (CVPR'91)*, 1991.

[16] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[17] Y. Chen, W. Gao, G. Fang, C. Yang, and Z. Wang. CSLDS: Chinese sign language dialog system. In *International Workshop on Analysis and Modeling of Faces and Gestures*, Nice, France, 2003.

[18] C. Christoudias, B. Georgescu, and Peter Meer. Synergism in low level vision. In *International Conference on Pattern Recognition*, Quebec City, Canada, 2002.

[19] R. Cipolla and P. Giblin. *Visual Motion of Curves and Surfaces*. Cambridge University Press, 2000.

[20] D. Crispell, D. Lanman, P. Sibley, Y. Zhao, and G. Taubin. Beyond sillhouettes: Surface reconstruction using multi-flash photography. In *International Symposium on 3D Data Processing, Visualization and Transmission*, 2006.

[21] M. Daum and G. Dudek. On 3-D Surface Reconstruction using Shape from Shadows. In *Conference on Computer Vision and Pattern Recognition (CVPR'98)*, pages 461–468, June 1998.

[22] J. Davis, D. Nehab, R. Ramamoothi, and S. Rusinkiewicz. Spacetime stereo: a unifying framework for depth from triangulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2), 2005.

[23] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of SIGGRAPH 1997*, August, 1997.

[24] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, and A. Santella. Suggestive contours for conveying shape. *SIGGRAPH 2003 / ACM Transactions on Graphics*, 2003.

[25] D. DeCarlo and A. Santella. Stylization and abstraction of photographs. *SIGGRAPH 2002 / ACM Transactions on Graphics*, 2002.

[26] G. Egnal and R. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1127–1133, 2002.

[27] E. Eisemann and F. Durand. Flash photography enhancement via intrinsic relighting. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[28] R. Fattal, D. Lischinski, and M. Werman. Gradient domain high dynamic range compression. In *SIGGRAPH 2002 / ACM Transactions on Graphics*, 2002.

[29] P. Favaro and S. Soatto. A geometric approach to shape from defocus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):406–417, 2005.

[30] P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision. In *Conference on Computer Vision and Pattern Recognition (CVPR'04)*, 2004.

[31] R. Feris, M. Turk, R. Raskar, K. Tan, and G. Ohashi. Exploiting depth discontinuities for vision-based fingerspelling recognition. In *IEEE Workshop on Real-time Vision for Human-Computer Interaction (in conjunction with CVPR'04)*, Washington DC, USA, 2004.

[32] G. Finlayson, M. Drew, and B. Funt. Spectral sharpening: Sensor transformations for improved color constancy. *Journal of the Optical Society of America*, 11(5):1553–1562, 1994.

[33] G. Finlayson, S. Hordley, C. Lu, and M. Drew. On the removal of shadows from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):59–68, 2006.

[34] R. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):439–451, 1988.

[35] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

[36] W. Freeman and H. Zhang. Shape-time photography. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, Madison, USA, 2003.

[37] E. Gamble, D. Geiger, T. Poggio, and D. Weinshall. Integration of vision modules and labeling of surface discontinuities. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1576–1581, 1989.

[38] T. Gevers. Reflectance-based classification of color edges. In *International Conference on Computer Vision (ICCV'03)*, Nice, France, 2003.

[39] S. Gokturk and C. Tomasi. 3D head tracking based on recognition and interpolation using a time-of-flight depth sensor. In *Conference on Computer Vision and Pattern Recognition (CVPR'04)*, Washington, DC, 2004.

[40] B. Gooch, E. Reinhard, and A. Gooch. Human facial illustrations: Creation and psychophysical evaluation. *ACM Transactions on Graphics*, 23(1):24–44, 2004.

[41] S. Hasinoff and K. Kutulakos. Confocal stereo. In *European Conference on Computer Vision (ECCV'06)*, pages 620–634, 2006.

[42] M. Hatzitheodorou and J. Kender. An optimal algorithm for the derivation of shape from shadows. In *Conference on Computer Vision and Pattern Recognition (CVPR'88)*, pages 486–491, Ann Harbor, Michigan, 1988.

[43] A. Hertzmann and S. Seitz. Shape and materials by example: A photometric stereo approach. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, pages 533–540, Madison,Wisconsin, 2003.

[44] B. Horn and M. Brooks. *Shape from Shading*. MIT Press, 1989.

[45] E. Horn and N. Kiryati. Toward optimal structured light patterns. *Image and Vision Computing*, 17(2):87–97, 1999.

[46] P. Huggins, H. Chen, P. Belhumeur, and S. Zucker. Finding folds: On the appearance and identification of occlusion. In *Conference on Computer Vision and Pattern Recognition (CVPR'01)*, volume 2, pages 718–725, December 2001.

[47] S. Intille and A. Bobick. Disparity-space images and large occlusion stereo. In *European Conference on Computer Vision (ECCV'94)*, pages 179–186, 1994.

[48] R. Irvin and D. McKeown. Methods for exploiting the relationship between buildings and their shadows in aerial imagery. *IEEE Systems, Man, and Cybernetics*, 19(6):1564–1575, 1989.

[49] A. Isaksen, L. McMillan, and S. Gortler. Dynamically reparameterized light fields. *Proceedings of SIGGRAPH 2000*, 2000.

[50] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *European Conference on Computer Vision (ECCV'98)*, June 1998.

[51] J. Jia, J. Sun, C. Tang, and H. Shum. Bayesian correction of image intensity with spatial consideration. In *European Conference on Computer Vision (ECCV'04)*, pages 342–354, 2004.

[52] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932, 1994.

[53] S. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multi-view stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR'01)*, volume 1, pages 102–110, 2001.

[54] J. Kender and E. Smith. Shape from darkness: deriving surface information from dynamic shadows. In *International Conference on Computer Vision (ICCV'87)*, pages 539–546, London, UK, 1987.

[55] G. Klinker, S. Shafer, and T. Kanade. The measurement of highlights in color images. *International Journal of Computer Vision*, 2:7–32, 1988.

[56] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *International Conference on Computer Vision (ICCV'01)*, Vancouver, Canada, 2001.

[57] M. Kolsch and M. Turk. Robust hand detection. In *International Conference on Automatic Face and Gesture Recognition*, Seoul, Korea, 2004.

[58] D. Kriegman and P. Belhumeur. What shadows reveal about object structure. *Journal of the Optical Society of America*, pages 1804–1813, 2001.

[59] Kutulakos and Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[60] M. Lamar and M. Bhuiyant. Hand alphabet recognition using morphological PCA and neural networks. In *International Joint Conference on Neural Networks*, pages 2839–2844, Washington, USA, 1999.

[61] E. Land and J. McCann. Lightness and retinex theory. *Journal of the Optical Society of America*, 61:1–11, 1971.

[62] D. Lee. Coping with discontinuities in computer vision: Their detection, classification, and measurement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(4):321–344, 1990.

[63] S. Lee. *Understanding of Surface Reflections in Computer Vision by Color and Multiple Views*. PhD thesis, University of Pennsylvania, 1991.

[64] B. Leibe, T. Starner, W. Ribarsky, Z. Wartell, D. Krum, J. Weeks, B. Singletary, and L. Hodges. The perceptive workbench: Toward spontaneous and natural interaction in semi-immersive virtual environments. *IEEE Computer Graphics and Applications*, 20(6):54–65, 2000.

[65] S. Lu, D. Metaxas, D. Samaras, and J. Oliensis. Using multiple cues for hand tracking and model refinement. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, Madison, USA, 2003.

[66] D. Marr. *Vision: a Computational Investigation into the Human Representation and Processing of Visual Information*. MIT Press, 1982.

[67] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *Proceedings of SIGGRAPH 2000*, pages 369–374, New Orleans, LA, 2000.

[68] M.Bell and W. Freeman. Learning local evidence for shading and reflectance. In *International Conference on Computer Vision (ICCV'01)*, volume 1, pages 670–677, 2001.

[69] M. McGuire, W. Matusik, H. Pfister, J. Hughes, and F. Durand. Defocus video matting. *SIGGRAPH 2005 / ACM Transactions on Graphics*, 2005.

[70] S. Nayar, X. Fang, and T. Boult. Removal of specularities using color and polarization. In *Conference on Computer Vision and Pattern Recognition (CVPR'93)*, pages 583–590, New York City, USA, 1993.

[71] G. Ohashi and Y. Shimodaira. Edge-based feature extraction method and its application to image retrieval. In *7th World Multi-conference on Systemics, Cybernetics and Informatics*, Florida, USA, 2003.

[72] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[73] M. Oren and S. Nayar. A theory of specular surface geometry. *International Journal of Computer Vision*, 2:105–124, 1997.

[74] M. Osadchy, D. Jacobs, and R. Ramamoorthi. Using specularities for recognition. In *International Conference on Computer Vision (ICCV'03)*, Nice, France, 2003.

[75] L. Parida and D. Geiger. Junctions: detection, classification and reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(7):687–698, 1998.

[76] J. Paterson, D. Claus, and A. Fitzgibbon. BRDF and geometry capture from extended inhomogeneous samples using flash photography. *Computer Graphics Forum (Special Eurographics Issue)*, 24(3):383–391, 2005.

[77] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[78] M. Petrov, A. Talapov, T. Robertson, A. Lebedev, A. Zhilyaev, and L. Polonskiy. Optical 3d digitizers: Bringing life to the virtual world. *IEEE Computer Graphics and Applications*, 18(3):28–37, 1998.

[79] N. Petrovic, I. Cohen, B. Frey, R. Koetter, and T. Huang. Enforcing integrability for surface reconstruction algorithms using belief propagation in graphical models. In *Conference on Computer Vision and Pattern Recognition (CVPR'01)*, volume 1, pages 743–748, 2001.

[80] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, and K. Toyama. Digital photography with flash and no-flash image pairs. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[81] J. Posdamer and M. Altschuler. Surface measurement by space encoded projected beam systems. *Computer Graphics and Image Processing*, 18(1):1–17, 1982.

[82] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C: The Art of Scientific Computing* . Pearson Education, 1992.

[83] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. *SIGGRAPH 2006 / ACM Transactions on Graphics*, 2006.

[84] R. Raskar, A. Ilie, and J. Yu. Image fusion for context enhancement and video surrealism. In *International Symposium on Non-Photorealistic Animation and Rendering*, pages 85–152, Annecy, France, 2004.

[85] R. Raskar, K. Tan, R. Feris, J. Yu, and M. Turk. A non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[86] R. Raskar, J. Yu, and A. Illie. A Non-photorealistic camera: detecting silhouettes with multi-flash. In *SIGGRAPH Technical Sketch*, San Diego, California, 2003.

[87] D. Raviv, Y.H. Pao, and K. A. Loparo. Reconstruction of three-dimensional surfaces from two-dimensional binary images. In *IEEE Transactions on Robotics and Automation*, volume 5(5), pages 701–710, Oct 1989.

[88] J. Rebollar, R. Lindeman, and N. Kyriakopoulos. A multi-class pattern recognition system for practical fingerspelling translation. In *International Conference on Multimodal Interfaces*, Pittsburgh, USA, 2002.

[89] A. Sa, P. Carvalho, and L. Velho. (b, s)-bcsl : Structured light color boundary coding for 3D photography. In *International Fall Workshop on Vision, Modeling, and Visualization*, 2002.

[90] J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.

[91] I. Sato, Y. Sato, and K. Ikeuchi. Stability issues in recovering illumination distribution from brightness in shadows. In *Conference on Computer Vision and Pattern Recognition (CVPR'01)*, pages 400–407, 2001.

[92] S. Savarese, H. Rushmeier, F. Bernardini, and P. Perona. Shadow carving. In *International Conference on Computer Vision (ICCV'01)*, Vancouver, Canada, 2001.

[93] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *International Journal of Computer Vision*, volume 47(1), pages 7–42, 2002.

[94] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, pages 195–202, Madison,Wisconsin, 2003.

[95] J. Segen and S. Kumar. Shadow gestures: 3D hand pose estimation using a single camera. In *Conference on Computer Vision and Pattern Recognition (CVPR'99)*, pages 479–485, Fort Collins, USA, 1999.

[96] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Conference on Computer Vision and Pattern Recognition (CVPR'06)*, New York, NY, 2006.

[97] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *International Conference on Computer Vision (ICCV'03)*, Nice, France, 2003.

[98] E. Shechtman, Y. Caspi, and M. Irani. Space-time super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):531–545, 2005.

[99] N. Snavely, S. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. *SIGGRAPH 2006 / ACM Transactions on Graphics*, 2006.

[100] F. Solomon and K. Ikeuchi. Extracting the shape and roughness of specular lobe objects using four light photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):449–454, 1996.

[101] A. Spoerri and S. Ullman. The early detection of motion boundaries. In *International Conference on Computer Vision (ICCV'87)*, pages 209–218, London, UK, 1987.

[102] T. Starner, J. Weaver, and A. Pentland. Real-time american sign language recognition using desk- and wearable computer-based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, 1998.

[103] A. Strat and M. Oliveira. A point-and-shoot color 3D camera. In *International Conference on 3-D Digital Imaging and Modeling*, pages 483–492, 2003.

[104] J. Sun, S. Kang, and H. Shum. Symetric stereo matching for occlusion handling. In *Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, California, 2005.

[105] J. Sun, Y. Li, S. Kang, and H. Shum. Flash matting. *SIGGRAPH 2006 / ACM Transactions on Graphics*, 2006.

[106] J. Sun, N. Zheng, and H. Shum. Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(07):787–800, 2003.

[107] R. Szeliski and H. Shum. Creating full view panoramic image mosaics and environment maps. In *Proceedings of SIGGRAPH 1997*, August, 1997.

[108] J. Tajima and M. Iwakawa. 3D data acquisition by rainbow range finder. In *International Conference on Pattern Recognition*, pages 309–313, 1990.

[109] K. Tan, J. Kobler, P. Dietz, R. Feris, and R. Raskar. Shape-enhanced surgical visualizations and medical illustrations with multi-flash imaging. In *International Conference on Medical Imaging Computing and Computer Assisted Intervention (MICCAI'04)*, France, 2004.

[110] P. Tan, S. Lin, L. Quan, and H. Shum. Highlight removal by illumination-constrained inpainting. In *International Conference on Computer Vision (ICCV'03)*, pages 164–169, 2003.

[111] M. Tappen and W. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters. In *International Conference on Computer Vision (ICCV'03)*, Nice, France, 2003.

[112] M. Tappen, W. Freeman, and E. Adelson. Recovering intrinsic images from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(9):1459–1472, 2005.

[113] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(4):413–424, 1986.

[114] W. Thompson, K. Mutch, and V. Berzins. Dynamic occlusion analysis in optical flow fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(4):374–383, 1985.

[115] C. Vogler and D. Metaxas. ASL recognition based on a coupling between HMMs and 3D motion analysis. In *International Conference on Computer Vision (ICCV'98)*, pages 363–369, Mumbai, India, 1998.

[116] J. Wang, Y. Xu, H. Shum, and M. Cohen. Video tooning. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[117] Y. Weiss. Deriving intrinsic images from image sequences. In *Proceedings of International Conference on Computer Vision (ICCV'01)*, pages 68–75, 2001.

[118] B. Wilburn, N. Joshi, V. Vaish, E. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *SIGGRAPH 2005 / ACM Transactions on Graphics*, 2005.

[119] L. Wixson. Detecting occluding edges without computing dense correspondence. In *DARPA Image Understanding Workshop*, 1993.

[120] L. Wolff and T. Boult. Constraining object features using a polarization reflectance model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):635–657, 1991.

[121] R. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

[122] D. Yang. *Shape from darkness under error*. PhD thesis, Columbia University, 1996.

[123] Y. Yu and J. Chang. Shadow graphs and surface reconstruction. In *European Conference on Computer Vision (ECCV'02)*, 2002.

[124] L. Zhang, B. Curless, and S. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *International Symposium on 3D Data Processing Visualization and Transmission*, pages 24–26, Padova, Italy, 2002.

[125] L. Zhang and S. Nayar. Projection defocus analysis for scene capture and image display. *SIGGRAPH 2006 / ACM Transactions on Graphics*, 2006.

[126] L. Zhang, N. Snavely, B. Curless, and S. Seitz. Spacetime faces: High-resolution capture for modeling and animation. *SIGGRAPH 2004 / ACM Transactions on Graphics*, 2004.

[127] T. Zickler, P. Belhumeur, and D. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. In *European Conference on Computer Vision (ECCV'02)*, 2002.

[128] T. Zickler, J. Ho, D. Kriegman, J. Ponce, and P. Belhumeur. Binocular helmholtz stereopsis. In *Conference on Computer Vision and Pattern Recognition (CVPR'03)*, pages 1411–1417, Madison,Wisconsin, 2003.