EFFICIENT TECHNIQUES FOR HIGH RESOLUTION STEREO

Yilin Wang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2016

Approved by:

Jan-Michael Frahm

Enrique Dunn

Henry Fuchs

Marc Niethammer

Philippos Mordohai

ABSTRACT

Yilin Wang: EFFICIENT TECHNIQUES FOR HIGH RESOLUTION STEREO
(Under the direction of Jan-Michael Frahm and Enrique Dunn)

The purpose of stereo is extracting 3-dimensional (3D) information from 2-dimensional (2D) images, which is a fundamental problem in computer vision. In general, given a known imaging geometry the position of any 3D point observed by two or more different views can be recovered by triangulation, so 3D reconstruction task relies on figuring out the pixel's correspondence between the reference and matching images. In general computational complexity of stereo algorithms is proportional to the image resolution (the total number of pixels) and the search space (the number of depth candidates). Hence, high resolution stereo tasks are not tractable for many existing stereo algorithms whose computational costs (including the processing time and the storage space) increase drastically with higher image resolution. The aim of this dissertation is to explore techniques aimed at improving the efficiency of high resolution stereo without any accuracy loss.

The efficiency of stereo is the first focus of this dissertation. We utilize the implicit smoothness property of the local image patches and propose a general framework to reduce the search space of stereo. The accumulated matching costs (measured by the pixel similarity) are investigated to estimate the representative depths of the local patch. Then, a statistical analysis model for the search space reduction based on sequential probability ratio test is provided, and an optimal sampling scheme is proposed to find a complete and compact candidate depth set according to the structure of local regions. By integrating our optimal sampling schemes as a pre-processing stage, the performance of most existing stereo algorithms can be significantly improved. The accuracy of stereo algorithms is the second focus. We present a plane-based approach for the local geometry estimation combining with a parallel structure propagation algorithm, which outperforms most state-of-the-art stereo algorithms. To obtain precise local structures, we also address the problem of utilizing surface normals, and provide a framework to integrate color and normal information for high quality scene reconstruction.

Dedicated to my parents.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Jan-Michael Frahm, for inspiring, guiding, and supporting me over the years. His admirable knowledge and insights made this dissertation possible. I would also like to thank my co-advisor Enrique Dunn, for his constant patience and his guidance and support not only on my research but also on my life. I am fortunate to have had Henry Fuchs, Marc Niethammer, and Philippos Mordohai on my committee. Their advice and suggestions help me gain new insights for this dissertation.

Thanks Enliang Zheng for being a great officemate and bringing me lots of interesting discussions in the past three years. Thanks Jared Heinly for his kind support and tolerance to my endless questions. I am also grateful to Ke Wang for his great help on my research and life, and to Dinghuang Ji for enjoyable memories of pingpong and basketball. Many thanks to other members in the vision group: Yi Xu, Joseph Tighe, Yunchao Gong, David Paul Perra, Junpyo Hong, Meng Tan, Hongsheng Yang, Sangwoo Cho and Hyo jin Kim, and it is a great pleasure to work with you all. Also I deeply appreciate the help and suggestions from former members in our group: Rahul Raguram, David Gallup, Pierre Georgel, and Shih-Ling Keng.

I would like to thank my friends in the computer science department: Chen-Rui Chou, Mingsong Dou, Cong Liu, Liang Shan, Peng Li, Hao Xu, Lei Wei, Richard Skarbez, and many others, for giving me memorable experiences during my PhD years.

Also, I am grateful to many members of UNC computer science faculty and staff, whose support enabled me to progress smoothly.

Finally, I would like to thank my parents for their love, encouragement and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

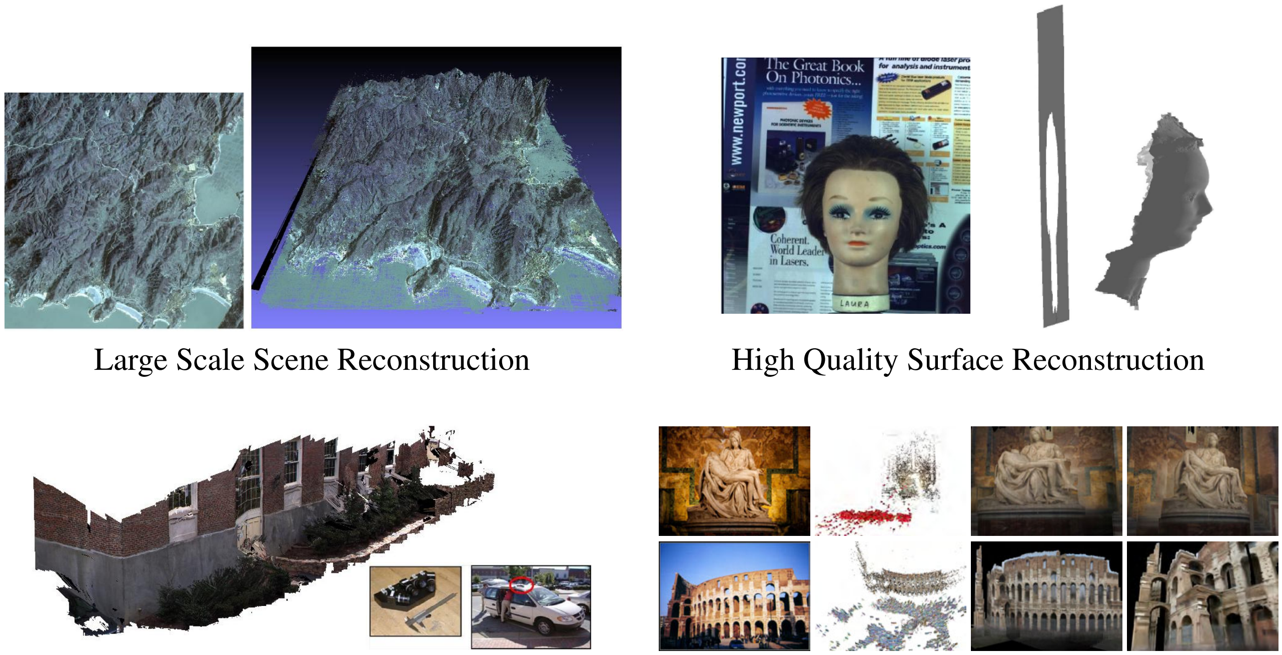
CHAPTER 1: INTRODUCTION

In computer vision, stereo is a fundamental research topic that focuses on capturing the shape of real objects. The research began in the 1970s (Hannah, 1974; Marr and Poggio, 1976, 1979), and numerous stereo algorithms (Kolmogorov and Zabih, 2001; Rusinkiewicz et al., 2002; Hirschmuller, 2005; Klaus et al., 2006; Yoon and Kweon, 2006; Yang et al., 2009; Rhemann et al., 2011; Mei et al., 2011) and outstanding hardware (Izadi et al., 2011; Smisek et al., 2013) have been proposed in last few decades. Generally, stereo methods can be classified as active approaches (interfering with objects) (Rusinkiewicz et al., 2002; Izadi et al., 2011) and passive approaches (not interfering with objects) (Kolmogorov and Zabih, 2001; Hirschmuller, 2005; Klaus et al., 2006; Yoon and Kweon, 2006; Yang et al., 2009; Rhemann et al., 2011; Mei et al., 2011) .

There are two main types of active stereo approaches: Structured Light (Scharstein and Szeliski, 2003a) and Time of Flight (Hansard et al., 2012), where the former one is the process of projecting a known pattern of pixels (often grids or horizontal bars) onto a scene, and the latter one is a range imaging camera system that resolves distance based on the known speed of light, measuring the time-of-flight of a light signal between the camera and the subject for each point of the image. Kinect [1] is a typical active stereo camera, which is able to perform real time 3d reconstruction but only works well within a limited depth range (usually $1m \sim 5m$). Furthermore, if multiple active stereo cameras are running at the same time, the devices may disturb each other's measurements.

Due to the limitation of active cameras, passive stereo approaches are still good choices in many applications (as shown in Figure 1.1), such as High-Quality Reconstruction (Beeler et al., 2010; Furukawa and Ponce, 2010; Matthies et al., 2007), Real-time 3D Urban/Terrain Modeling (Gallup et al., 2007; Matthies et al., 2007), and Modeling from Image Collections (Furukawa et al.,

[1] Kinect for Windows: http://www.microsoft.com/en-us/kinectforwindows/



Large Scale Scene Reconstruction

High Quality Surface Reconstruction

Real-time 3D Urban Modeling
(Gallup et al., 2007)

Modeling from Image Collection
(Frahm et al., 2010)

Figure 1.1: Applications of Passive Stereo.

2010; Frahm et al., 2010). In passive stereo, multiple color/intensity images covering the same scene are captured at different camera positions, and then the 3D information of one pixel can be estimated by finding its corresponding pixels on matching images based on perspective geometry (Hartley and Zisserman, 2003). In the Middlebury stereo benchmark [2], more than 150 different passive stereo algorithms have been proposed in the last ten years, however, most of them are only evaluated by images with Video Graphics Array (VGA) resolutions ($640 \times 480 \approx 0.3M$ pixels, Fig 1.2 left), which are much smaller than the resolution of images captured by state of the art mobile phones (around $8M$ pixels) and digital cameras ($20M$ pixels, Fig 1.2 middle). Furthermore, some applications like 3D map reconstruction (Hirschmuller et al., 2012; Dial and Grodecki, 2005) require to process even larger images ($150M$ pixels, Fig 1.2 right) captured by aircrafts or satellites. Since the computational complexity of stereo is proportional to the number of image pixels and the number of candidate depths for testing (called search space), the computational cost for high resolution stereo would be unaffordable for many existing stereo approaches. Performing high

[2] Middlebury Stereo Evaluation: http://vision.middlebury.edu/stereo/eval/

resolution stereo efficiently is still an open problem in computer vision. Since directly performing stereo algorithms designed for low resolution image in high resolution cases will lead to lots of redundant costs, reducing the computational requirements has recently received renewed attention (Sizintsev, 2008; Hawe et al., 2011; Min et al., 2011).

In this dissertation, we study the efficiency of stereo by investigating two critical components of the stereo algorithm: the search space and the aggregation structure. The former determines the number of candidate depths needed to be tested, while the latter one will heavily influence the accuracy of the estimated depths. By improving the performance of stereo algorithms in both aspects, i.e. reducing the search space and refining the aggregation structure, we can obtain depth maps with more efficiency and higher quality than existing approaches.

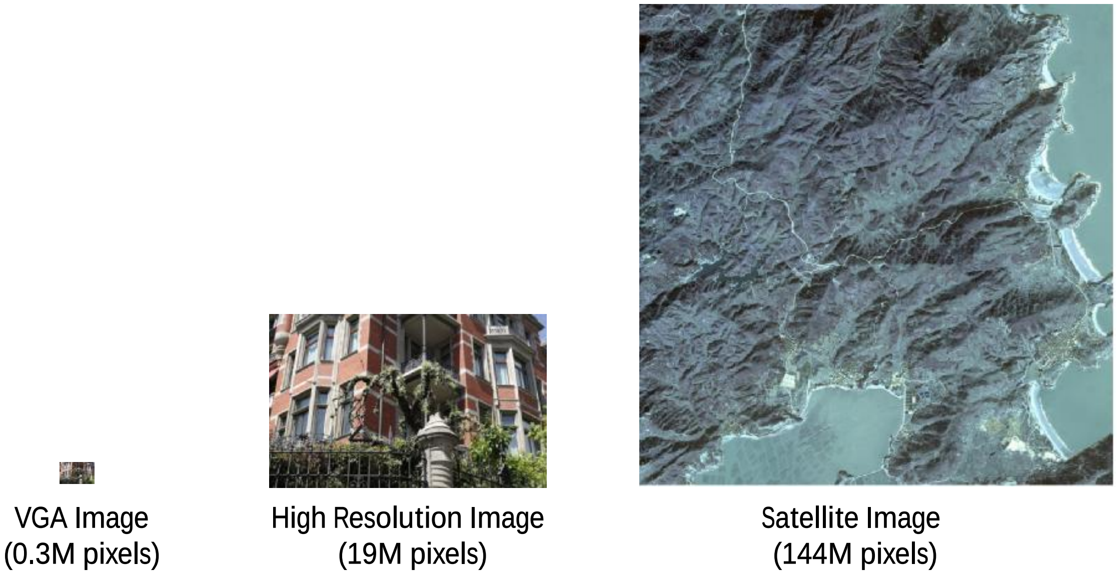


Figure 1.2: Image resolution comparison: VGA image, High Resolution image, and Satellite image.

Thesis Statement

The efficiency of stereo algorithms can be highly improved by reducing the search space under optimal sampling schemes, and high quality depth maps can be achieved by extracting and integrating local structures.

Contributions

My research makes the following contributions.

Search Space Reduction

Sparse Distributed Depth Sampling. We introduce a novel framework for efficient stereo disparity estimation leveraging the spatial smoothness typically assumed in stereo. The smoothness constraint presumes that a neighboring set of pixels shares the same disparity or the disparity varies smoothly. The key insight is that it hence suffices to evaluate any single one of those pixels at the correct disparity to identify a valid estimate for the entire set. We leverage this insight into the formulation of a complexity reducing mechanism, and distribute the exploration of the disparity search space among neighboring pixels, effectively reducing the set of disparity hypotheses evaluated at each individual pixel.

Sequential Optimal Sampling. We develop a sequential optimal sampling framework for stereo disparity estimation by adapting the Sequential Probability Ratio Test (SPRT) model. The proposed framework operates over local image neighborhoods by iteratively estimating single pixel disparity values until sufficient evidence has been gathered to either validate or contradict the current hypothesis regarding local scene structure. The output of the sampling within a given region is a set of sampled pixel positions along with a robust and compact estimate of the set of disparities contained within that region. The attainment of such disparity set enables the effective reduction of the disparity search space for all remaining non-sampled pixels. Accordingly, the proposed sampling framework is a general pre-processing mechanism aimed at reducing computational complexity of disparity search algorithms.

Local Structure Estimation

Surflet-based Stereo. We present a multi-modal surface reconstruction approach, which utilizes direct surface orientation measurements, along with luminance information, to obtain high quality 3D reconstructions. The proposed approach models local surface geometry (called Surflet) as a set of intersecting natural cubic splines estimated through least squares fitting of our input pixelwise surface normal measurements. We use this representation to detect discontinuities and divide the scene into disjoint continuous surfaces, which are constructed by the aggregation of connected local surface geometry elements. In order to obtain absolute depth estimates, we introduce the concept of multi-view surflet sweeping, where we search for the most photo-consistent patch displacement along a viewing ray. This approach improves on existing shape from normals methods by enabling absolute depth estimates for scenes with multiple objects. Furthermore, in contrast to existing multi-view stereo methods, this method is able to reconstruct textureless regions through the propagation of relative surface orientation measurements.

Stereo by Local Structure Propagation. We present a plane-based sampling scheme to robustly approximate the structure of a local patch, which leverages the pre-computed sampling positions and refined candidate depths. A parallel structure propagation mechanism is then proposed, which outperforms state-of-art propagation-based stereo algorithms with random initialization. Moreover, by integrating with sequential optimal sampling scheme instead of random sampling, the completeness and sufficiency of exploited local structures are further improved.

Organization

The remainder of the dissertation is organized as follows.

Chapter 2 provides a general framework of stereo algorithms, which consists of local matching cost computation, disparity computation, and disparity refinement. Classical global and local stereo approaches are reviewed, followed by a specific literature review of sampling-based stereo techniques.

Chapter 3 presents an efficient and accurate modification to the standard exhaustive search paradigm in local stereo. The approach reduces the search space by sampling matching costs for the local patch, and then using a voting scheme to recover the depth map from a sparse pattern of seeds. The joint use of these orthogonal performance-oriented optimization mechanisms enables a dramatic reduction in the computational burden of local stereo methods.

Chapter 4 proposes a statistical analysis framework for search space reduction based on the sequential ratio probability test from the sequential decision theory. The method avoids unnecessary evaluation of irrelevant disparities for pixels of an image, and can be combined with a large variety of existing stereo estimation methods.

Chapter 5 proposes a novel method to recover the absolute surfaces of a scene using surface normal and texture information as inputs. The approach detects surface discontinuities by modeling local surface topology as a parametric surface obtained through least square data fitting of a set of measured surface normals. Next, accurate depth estimates are obtained through multi-view surflet sweeping.

Chapter 6 describes a propagation-based stereo scheme integrated with the sequential optimal sampling scheme, which maintains the quality of the exhaustive disparity estimation at significantly lower computational costs.

Finally, Chapter 7 concludes with a summary and discussion of potential improvements and future work.

CHAPTER 2: BACKGROUND AND RELATED WORK

In the pinhole camera model (Hartley and Zisserman, 2003), where the camera aperture is described as a point and no lenses are used to focus light, given the calibration matrix (formed by the focal length and principal point) and extrinsic matrix (image rotation and translation), any 2D image can be placed into the 3D space with the corresponding camera pose (viewpoint). As illustrated in Figure 2.1, shooting a viewing ray (straight line) from the viewpoint through the corresponding 3D position of a pixel (called reference pixel) on the image plane, all possible 3D positions of this reference pixel are restricted along this viewing ray, with the actual 3D point position determined by one factor, *depth*. If we know the corresponding pixel on another image (called a matching image), which gives another viewing ray, ideally the intersection of these two viewing rays is the true 3D position of the pixel. Hence, the crucial problem becomes how to find the corresponding pixel on the matching image. The similarity of pixels is measured by photo-consistency, and those approaches are called photo-consistency-based stereo.

In general, the projection of the viewing ray onto one matching image is an epipolar line, and a special case is when images are rectified, the epipolar line will only pass through the pixels on the same row of the reference pixel. In this case, the depth is a one-to-one (non-linear) correspondence to the column difference (also called *disparity*) of the reference pixel and the matching pixel. When we discussed the correspondence between depths and pixel-level matching costs for rectified images, it would be an intuitive and convienent way to illustrate concepts in the scope of disparity.

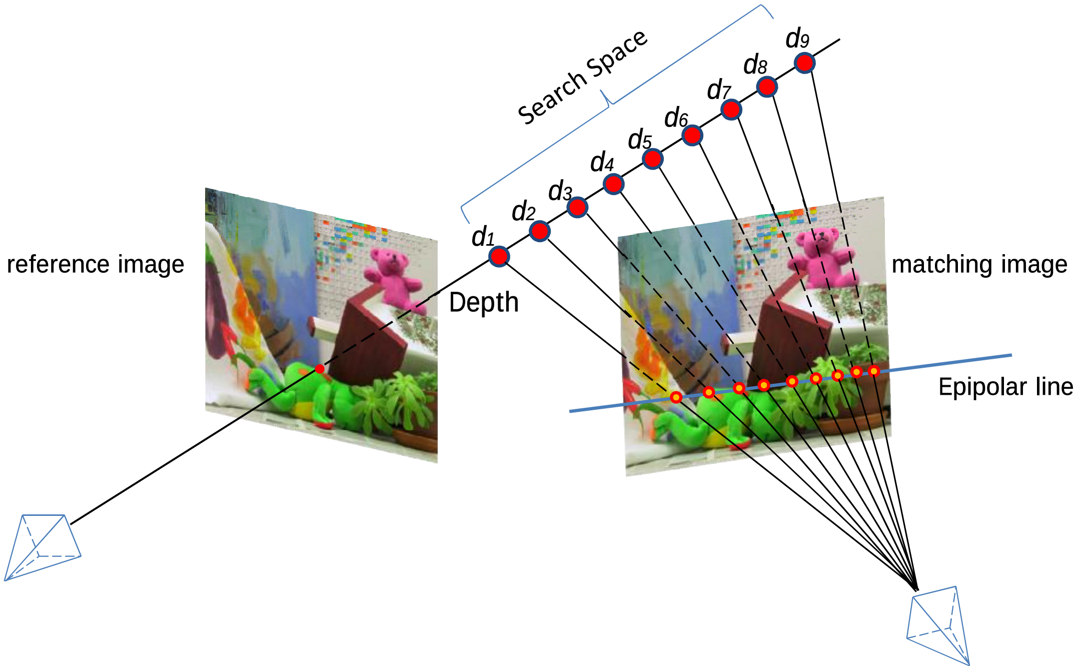


Figure 2.1: Depth Recovery from Stereo.

A General Framework of Stereo Algorithms

In the first part of this chapter, we introduce a general framework of stereo algorithms, which consists of two components: local matching cost computation and disparity (or depth in unrectified cases) computation.

Local Matching Cost Computation

The basic operation of stereo is to compute the matching cost for a given disparity, and there are two issues to be considered: 1) which metric is used for photo-consistency comparison, and 2) what spatial support region are used for local cost computation.

Metric for Photo-Consistency Comparison

As previously discussed, to recover the depth of the pixel we need to find its corresponding pixel on the matching image. Accordingly, there should be detectable cues to distinguish the correct

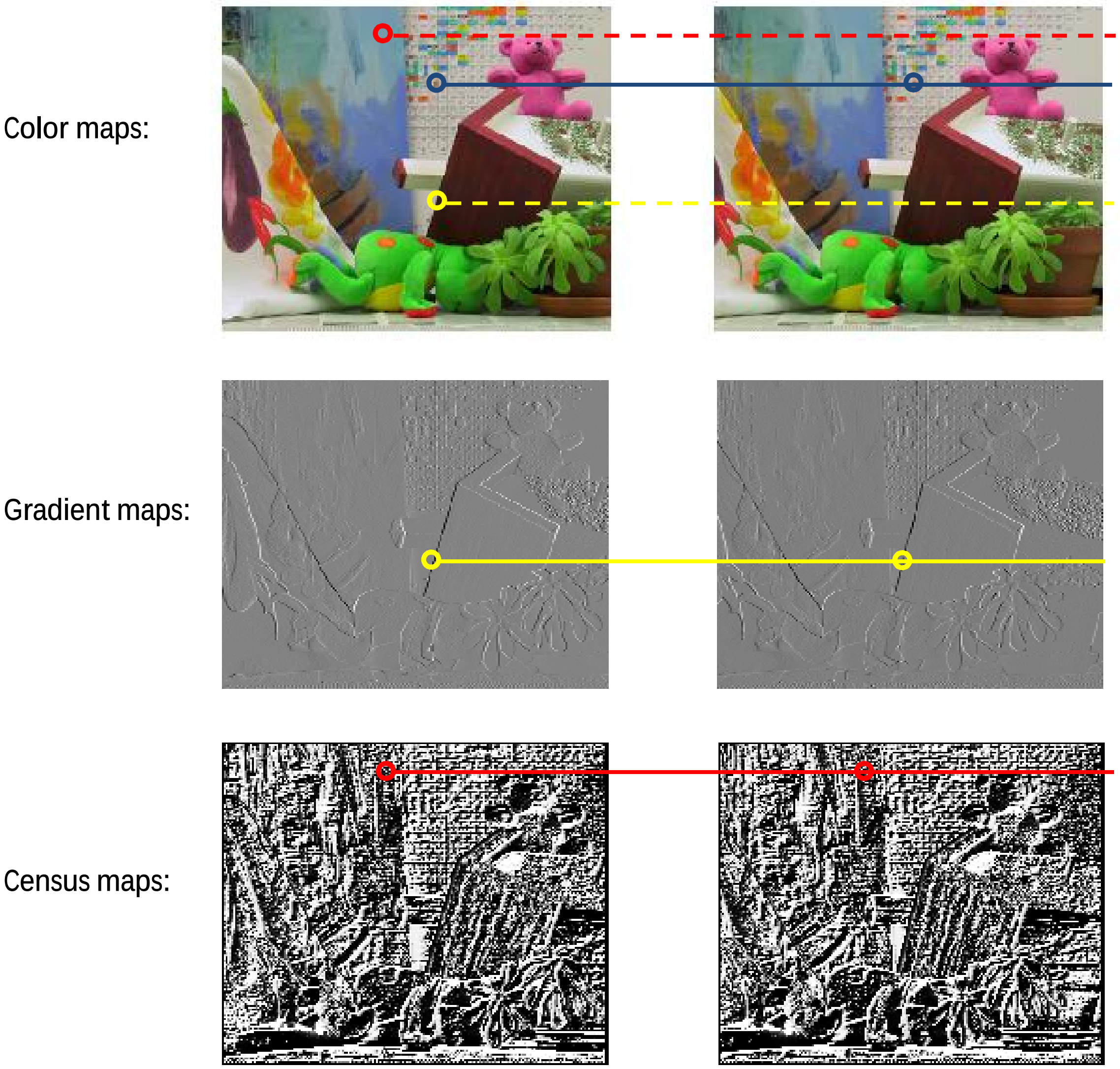
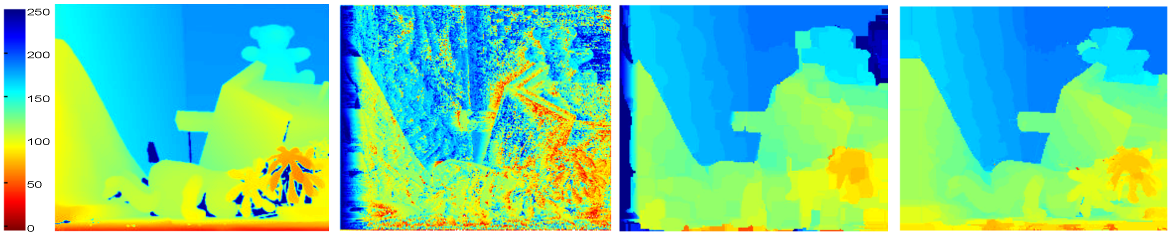


Figure 2.2: Three kinds of information used for photo-consistency comparison: Color, Gradient, and Census. The yellow and red pixels are ambiguous for color-based matching, but can be distinguished by gradient (yellow) and census (red).



Ground Truth Pixel-wise Matching SAD Adaptive Weight

Figure 2.3: Depth maps recovered from different aggregation structures.

matching pixel from other incorrect pixels. The basic photo-consistency cue for stereo is the color similarity (or intensity similarity for gray images), including SAD (Sum of Absolute Differences) (Kanade et al., 1995), SSD (Sum of Squared Differences) (Hannah, 1974; Anandan, 1989), and NCC (Normalized Cross-Correlation) (Hannah, 1974; Bolles et al., 1993; Scharstein and Szeliski, 2002). Sometimes, color information is too ambiguous to find the correct matching pixel (as the yellow and red pixels shown in Figure 2.2), there are also two additional cues widely used in stereo: gradient (Seitz, 1989; Scharstein, 1994; Bleyer et al., 2011; Rhemann et al., 2011; Min et al., 2013) and census (Zabih and Woodfill, 1994; Humenberger et al., 2010; Mei et al., 2011). The gradient of the intensity image captures many details of the fine structures by amplifying the intensity variances in different directions and is useful to recover high quality depth maps; while census encodes local image structures with relative orderings of the pixel intensities instead of the intensity values themselves. Accordingly, census tolerates outliers due to radiometric changes and image noise, and is suitable for ambiguous regions.

Reliability of Matching Cost

To compute the matching cost for one pixel, the simplest way is to compare its color difference with the corresponding matching pixel. However, it is not a reliable criterion in most cases (as shown in Figure 2.3) because there would be many ambiguous pixels having too similar matching costs to distinguish the correct one. One way to remove the ambiguity is using the average matching cost of all pixels within a certain neighborhood (e.g. SAD), which is more reliable for pixels on a color smooth surface, but still fails when the neighborhood covers two or more discontinuous regions. A more robust aggregation structure to preserve edges is adaptive support weight (Yoon and Kweon, 2006) (also called Bilateral Filter in (Petschnigg et al., 2004; He et al., 2010)), where the matching cost for pixel p with disparity d is computed as:

$$cost(p,d) = \frac{\sum_{q \in N(p)} w(p,q) w(\bar{p},\bar{q}) e(q,\bar{q})}{\sum_{q \in N(p)} w(p,q) w(\bar{p},\bar{q})} \tag{2.1}$$

where $\bar{p} = p - d$ and $\bar{q} = q - d$ are the corresponding pixels of p and q on the matching image, while $e(q,\bar{q})$ is the pixelwise difference between q and $\bar{q}$ and $N(p)$ is the set of p's neighbors. The adaptive weight $w(p,q)$ for q with respect to the center p combines the similarity distance and the spatial distance:

$$w(p,q) = \exp\left(-\frac{\parallel (I_p - I_q) \parallel}{\lambda_{color}} - \frac{\parallel (p-q) \parallel}{\lambda_{spatial}}\right) \tag{2.2}$$

here λ_{color} and $\lambda_{spatial}$ are constants used to adjust the influences of color and spatial differences. Figure 2.4 shows an example of adaptive weights for the local patch $N(p)$, where we can see high weights are assigned to neighbors with the similar color as the center pixel. For neighbors with the same color (e.g. red pixels on the roof), the weights of pixels far from the patch center are smaller than the weights of pixels closer to the center.

The accuracy of the aggregated cost is also influenced by the aggregation structure. In order to aggregate matching costs from neighboring pixels, we need to assume their relative depths with respect to the center pixel. The basic assumption of the aggregation structure is the fronto-parallel plane, where all the neighbors have the same depths as the center pixel. However, such assumption would fail to get correct aggregated costs when the local patch contains non-fronto-parallel structures. In the example shown in Figure 2.5, the bottom region can only be correctly recovered by using oriented planes instead of fronto-parallel planes.

Let $S_p = \{s_{q \in N(p)}(d)\}$ be the aggregation structure of pixel p, where $s_q(d)$ is the disparity of neighboring pixel q given d as the depth of the center pixel p. Then $\tilde{q} = q - s_q(d)$ is the corresponding pixel of q on the matching image, and we modify Equation 2.1 to be a general

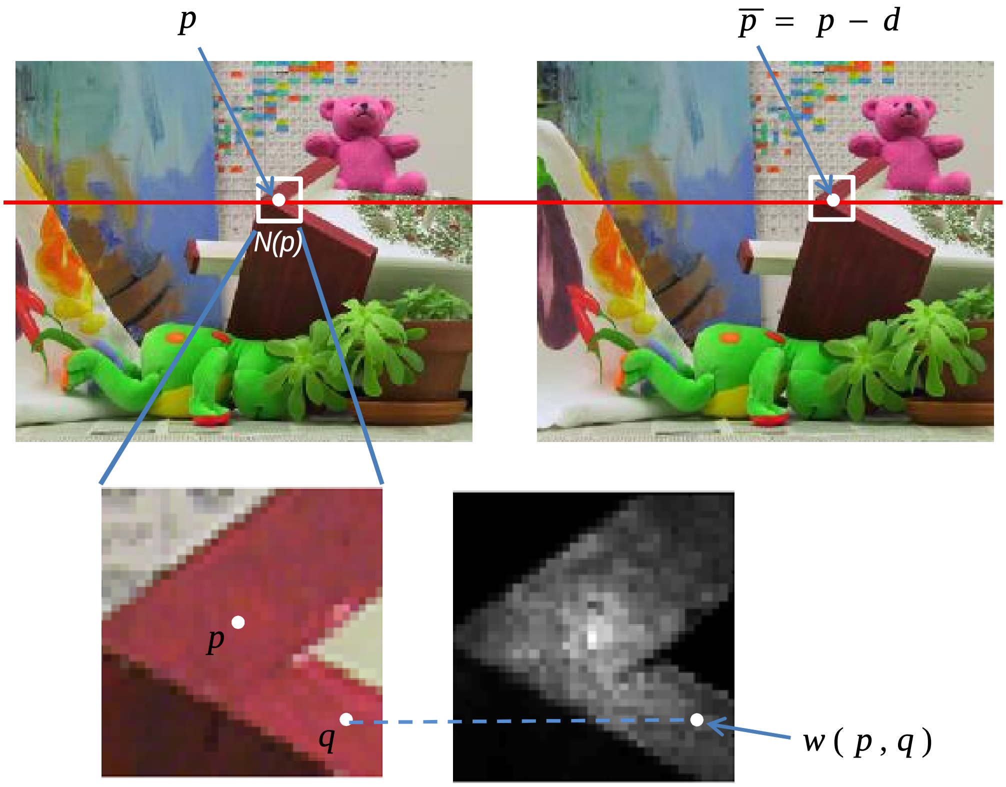


Figure 2.4: Adaptive weights for neighboring pixels.

formulation of the local cost computation:

$$cost(p, d, S_p) = \frac{\sum_{q \in N(p)} w(p, q) w(\tilde{p}, \tilde{q}) e(q, \tilde{q})}{\sum_{q \in N(p)} w(p, q) w(\tilde{p}, \tilde{q})} \tag{2.3}$$

Equation 2.3 can be simplified to fronto-parallel aggregation by setting $s_q(d) \equiv d$, and it further reduces to the uniform cost aggregation when $w(p, q) \equiv 1$ and even the pixel-wise comparison when $N(p) = \{p\}$. The reliability of a neighborhood depends on the structures of local patches, and usually needs to cover a large region, e.g.35×35 pixels in Yoon and Kweon (2006); Bleyer et al. (2011). Thus, there is a tradeoff between the cost reliability and computation complexity, and it can be improved by caching or sparse sampling techniques (Wang et al., 2012).

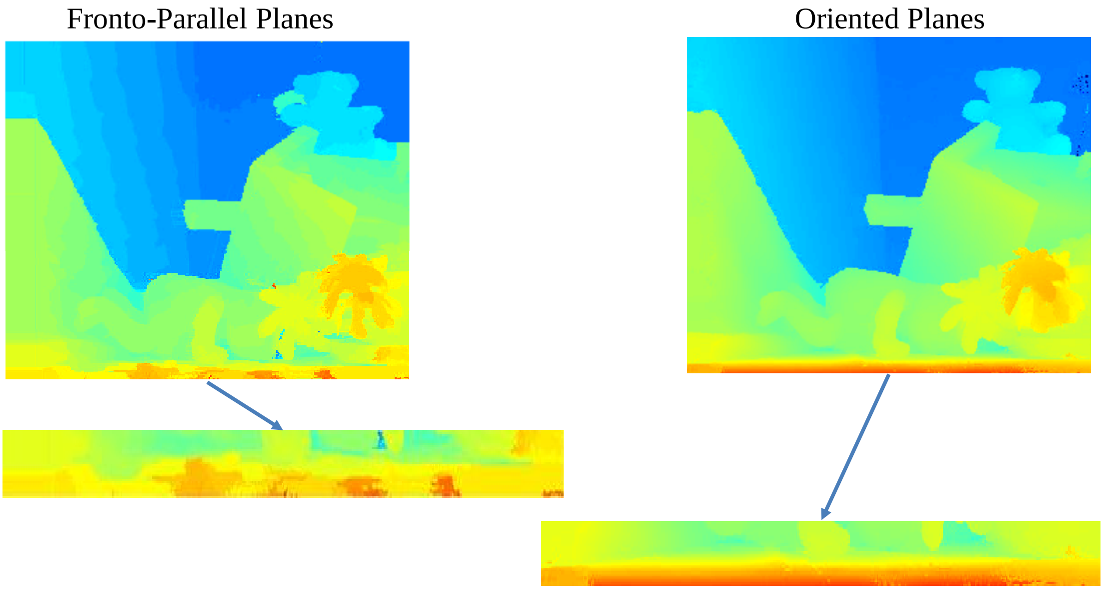


Figure 2.5: Comparison of tow aggregation structures: Fronto-parallel planes vs. Oriented planes.

Disparity Computation

The above metrics are used to evaluate all possible disparities, and in this section we will discuss how to select the right disparity for each pixel. The final disparity of the pixel is the one that has the best disparity cost. Besides the matching cost, the disparity cost may also consider the information passed from its neighboring pixels, and keep changing until it converges to a stable value. The disparity cost could be initialized by the pixel-wise matching cost (Tappen and Freeman, 2003), the local matching cost by adaptive support aggregation (Yoon and Kweon, 2006), or some predefined costs (Min et al., 2011). Then the disparity cost will be modified iteratively according to specific updating strategies, such as Graph-cut (Kolmogorov and Zabih, 2001, 2002), Belief-Propagation (Tappen and Freeman, 2003; Felzenszwalb and Huttenlocher, 2004; Klaus et al., 2006; Yang et al., 2010), Semi-Global Matching (Hirschmuller, 2008; Humenberger et al., 2010; Michael et al., 2013) and Non-Local Filter (Yang, 2012). A special case is the exhaustive local stereo such as SAD and SSD, which does not require any updating step and the initial local matching cost is just the final disparity cost. Not every disparity cost needs to be updated for each iteration, for propagation-based stereo like PatchMatch (Bleyer et al., 2011; Besse et al., 2012) and voting-based stereo HistogramAggregation (Min et al., 2011), only a small subset of disparities updates their costs in each iteration. Popular updating structures include grid (Graph-cut and Belief-Propagation), tree (Non-Local Filter), and straight lines (Semi-Global Matching and PatchMatch).

Stereo Algorithms

Accuracy and *Efficiency* are two major foci in stereo research. Commonly, high computational cost is required to recover accurate details, such as clear edges, by using global methods (Tappen and Freeman, 2003) or adaptive support aggregation windows (Yoon and Kweon, 2006).

To obtain a depth map with accurate details, global methods such as Graph Cuts (Kolmogorov and Zabih, 2001) and Belief Propagation (Tappen and Freeman, 2003) treat the disparity computation as an energy minimization problem, and the objective is to find an assignment of disparities d that minimize the global energy:

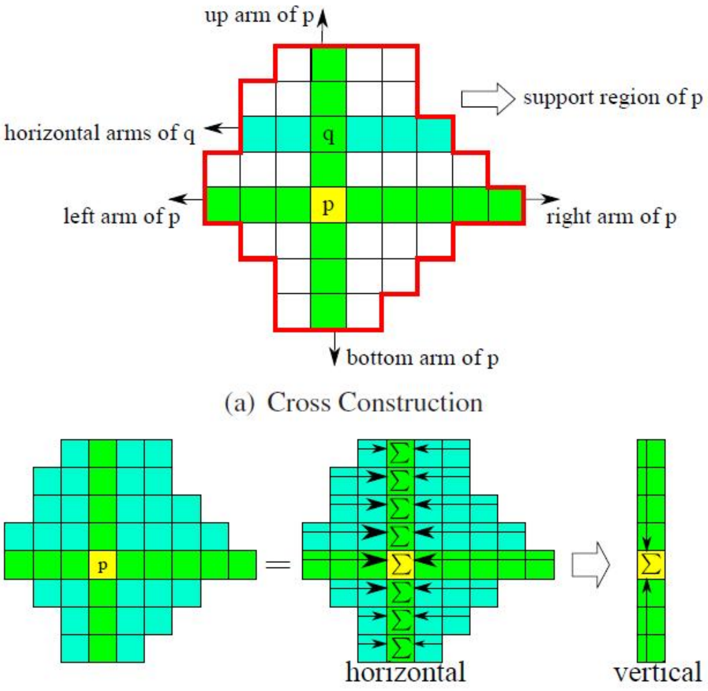
$$E(d) = E_{data}(d) + E_{smooth}(d) \tag{2.4}$$

where the data term $E_{data}(d)$ corresponds the local matching costs and smooth term $E_{smooth}(d)$ penalizes the discontinuities between neighboring pixels. There are two main benefits of global methods. First, for a homogenous region (assume the area is much larger than the size of the aggregation window), usually the matching costs of multiple candidates disparities are ambiguous. In this case, the global method is able to aggregate costs crossing the entire region so that the correct disparities can be more precisely determined by pixels along the boundaries. The second benefit is better edge preserving quality compared against original local methods, which assume pixels within the neighborhood belongs to the same surface and aggregate the matching costs equally. However, state of the art local stereo approaches are also able to preserve edges by using the adaptive support weight aggregation (Yoon and Kweon, 2006) (will be introduced in Section 2.1.1.2). A shortcoming of global methods is that their complexity is usually proportional to the square of the size of the search space, which makes it difficult to be applied in high resolution stereo.

The recent explosion in image resolution has brought efficiency to the forefront of requirements for high quality stereo. Algorithms for reducing the burden of matching cost aggregation while retaining matching accuracy include non-local aggregation (Yang, 2012), cross-based aggregation (Mei et al., 2011), and fast cost-volume filters (Rhemann et al., 2011).

Yang (2012) provided a non-local cost aggregation method, where the matching cost values are aggregated adaptively based on pixel similarity on a minimum spanning tree derived from the stereo image pair to preserve depth edges. It is a two step cost aggregation: each node will aggregate the original matching cost from leaf nodes towards the root node and store it as a temporal local cost. then the root will propagate the aggregated cost towards leaf nodes, and each node updates its local cost based on the propagated cost. The method has an extremely low computational complexity, and outperforms many local cost aggregation methods. The entire image is treated as a minimum spanning tree, and a shortcoming is that the method is difficult to be parallelized when dealing with high resolution images. Also since pixels only have few supports in the tree structure (only one father and several sons), the quality of boundary regions is slightly worse than the methods (Bleyer et al., 2011; Besse et al., 2012) using adaptive support weight aggregation.

Mei et al. (2011) proposed a cross-based aggregation to achieve comparable quality as the adaptive support weight aggregation with much less computation time, which is much more efficient for GPU implementation. The algorithm consists of two step aggregation: first an upright cross is constructed for each pixel. As illustrated in Figure 2.6, the local pixel p will extend arms in four directions, and each expansion will stop when hitting a pixel whose color is very different from the local pixel p, or the length of arm exceed a preset maximum length. The support region of the local pixel is modeled by merging the horizontal arms of the pixels lying on the vertical arms of the local pixel. In the second step, the cost in the support region is aggregated within two passes along the horizontal and vertical directions. In contrast to assigning different weights for non-homogenous pixels in adaptive support weight, the costs are aggregated equally from pixels within the support region, which still works well because the cross is formed by pixels with similar colors, and in some sense the aggregated region can be roughly treated as a continuous homogenous segment.



(a) Cross Construction

(b) Cost Aggregation

Figure 2.6: Cross-based aggregation (Mei et al., 2011). The algorithm consists of two step aggregation: (a) Cross construction and (b) Cost aggregation.

Another approximation of adaptive support weight aggregation, or called Joint Bilateral Filter (Petschnigg et al., 2004), is Guided Filter (He et al., 2010) and then applied in stereo by Rhemann et al. (2011). The spatial distance is discarded and the Equation (2.1) is represented by a linear filter $cost(q, d) = a_p e(q, \bar{q}) + b_p, \forall q \in N(p)$, where a_p and b_p are some linear coefficients assumed to be constant in the neighborhood $N(p)$. However, a pixel q is involved in all the windows $N(p)$ that contain q, so the value of cost is not the same when it is computed in different windows. A simple strategy is to average all the possible values of costs. The standard bilateral filtering process is a translation-variant convolution, whose computational cost increases drastically when the kernel becomes larger. Instead of directly performing the convolution, the guided filter computes the filter output by aggregating the linear coefficients a_p and b_p so that it can be computed in $O(N)$ time.

Sampling-based Stereo

Perpendicular to various cost aggregation approaches, search space reduction for stereo offers a complexity reducing framework to avoid the exhaustive evaluation of the cost volume (i.e. reducing the number of matching cost computations). Our proposed statical analysis sampling framework falls into this category. And in this section we discuss related sampling-based stereo methods.

Hierarchical stereo (Koschan et al., 1996; Van Meerbergen et al., 2002; Sizintsev, 2008) is a multi-resolution approach for deterministic search space reduction, but lacks adaptability to fine structures. Veksler (2006) compared the effect of using the disparities obtained by hierarchical stereo, dynamic programming and local stereo for limiting the disparity range, and concluded that reduction by local stereo resulted in almost no loss in accuracy with a significant efficiency improvement over hierarchical stereo and dynamic programming.

Wang et al. (2008) proposed a search space reduction method for Markov Random Fields (MRF) stereo (Kolmogorov and Zabih, 2001; Tappen and Freeman, 2003) based on estimating a putative disparity map from pixel-wise photo-consistency. Estimation reliability is verified through left-right consistency and reliable pixel estimates are propagated to the entire image. The disparity variability in the local neighborhood is then used to determine a candidate depth range for each pixel. However, this method requires a rough disparity map obtained by local stereo and is only meaningful when combined with global stereo methods.

Hawe et al. (2011) employ a compressed sensing formulation to reconstruct a high quality disparity map through spatial sub-sampling of the images to reduce computation. The disparity map is estimated for as little as 5% of the original image's pixels and the depth map is densified through optimization based sparse signal reconstruction techniques. In addition to the computational burden of the reconstruction process, the sufficiency and accuracy of the underlying seed generation process and their effect on the final estimation are not addressed by the authors.

Histogram Aggregation (HA) (Min et al., 2011) reduces the computational complexity of the the disparity estimation by combining a pixel-wise likelihood histogram aggregation scheme with

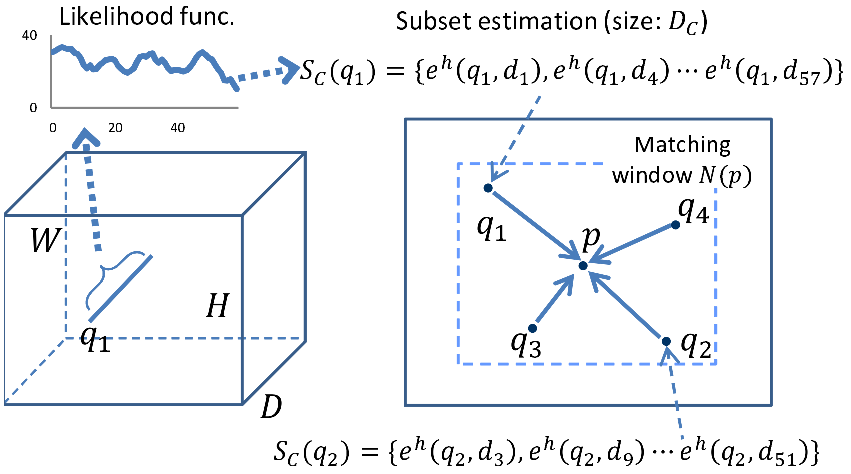

Likelihood func.
40
0
0
20
40
Subset estimation (size: D_C)
$S_C(q_1) = \{e^h(q_1, d_1), e^h(q_1, d_4) \cdots e^h(q_1, d_{57})\}$
Matching window $N(p)$
W
H
D
q_1
q_1
p
q_4
q_3
q_2
$S_C(q_2) = \{e^h(q_2, d_3), e^h(q_2, d_9) \cdots e^h(q_2, d_{51})\}$

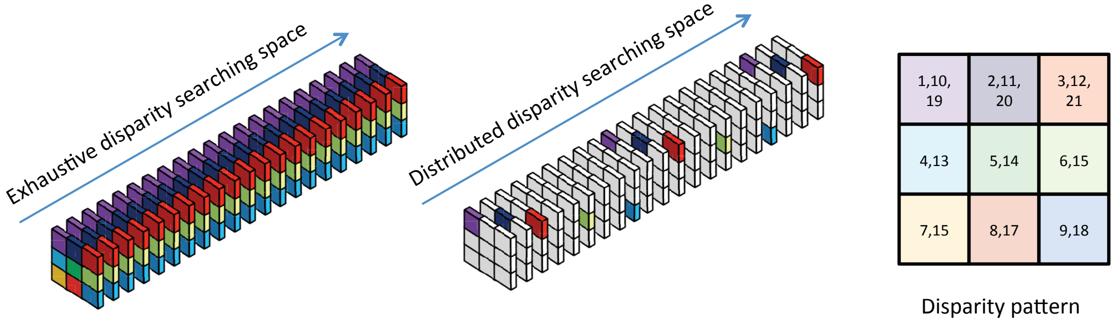
Figure 2.7: Histogram Aggregation (Min et al., 2011). Suppose the image size is $W \times H$ and the length of the search space is D. The likelihood function is computed for sampled pixel p_i, and then the depths with high likelihoods form a subset of weighted depths for p_i, where each survived depth is assigned a weight $e^h(q_i, d_j)$. Then each pixel q will aggregate all the weighted depths within its matching window, and its final depth is the one with the highest aggregated likelihood.

sparse image sampling. They discovered that their cost aggregation scheme has no monotonic correlation between estimation accuracy and sampling density. To be robust against noisy matching cost, the pixel-wise depth candidates for each seed in the sampling grid are attained by selecting a fixed number of local extrema in the seed cost function. The resulting set of extrema is used in a spatial voting framework to propagate the depths to the entire image. In some sense, the method could be treated as dense local stereo (pixelwise) on low resolution images and then voting for the original resolution image, which also has the problem of missing fine details as hierarchical stereo approaches.

PatchMatch (PM) (Bleyer et al., 2011) is a fixed propagation scheme with random depth initialization. With assumed sufficient sampling of the local structure (i.e. depths or oriented planes), the spatial propagation and pairwise hypothesis comparison message passing framework effectively converges in a few iterations. PM treats each pixel as a seed, which propagates its best structure to other pixels, and the received structure could be re-propagated from the local pixel to neighboring pixels only when the new structure has better aggregated matching costs than the current one; otherwise, the current local structure will be propagated. The propagation starts from the top-left of the image to the bottom-right and then backwards, and most pixels will obtain stable results after just a few iterations. An implicit assumption of PM is that the correct disparities among the random initialized seeds can reach corresponding pixels during the propagation, which may be not true for some isolated regions. To overcome such a defect of pixel-to-pixel propagation, Lu et al. (2013) proposed a generic and fast computational framework called PatchMatch Filter (PMF) by decomposing an image into compact superpixels.

CHAPTER 3: SPARSE DISTRIBUTED DEPTH SAMPLING

The accuracy and efficiency of depth estimates is contingent upon a variety of factors such as the photo-consistency *measure* (Scharstein and Szeliski, 2002) used to compare pixel similarity, the scope and form of the *aggregation mechanisms* (Scharstein and Szeliski, 2002; Yoon and Kweon, 2006) used to robustify individual pixel similarity measurements, as well as the *search strategy* (if any) used to explore the space of depth hypotheses. In order to develop an efficient depth *search strategy*, we leverage the implicit smoothness assumption used in most fronto-parallel photo-consistency measures, where pixels in a local neighborhood are assumed to belong to the same fronto-parallel surface and have the same depth. Clearly, such a smoothness assumption suggests that there is significant redundancy in an exhaustive search of the depth search space for each pixel. Hence, our first contribution is to reduce the number of evaluated depth hypothesis by spatially distributing (within a local neighborhood) the sampling of the depth hypothesis space. We introduce this structured approximation as *Distributed Depth Sampling* (DDS). Our second contribution in this Chapter is to incorporate within this new framework the recently introduced concept of spatial sampling (Min et al., 2011), we developed generalization of DDS we denote as


Exhaustive disparity searching space
Distributed disparity searching space
Disparity pattern

1,10, 19	2,11, 20	3,12, 21
4,13	5,14	6,15
7,15	8,17	9,18

Figure 3.1: Relationship between a traditional exhaustive search disparity strategy and our proposed DDS strategy. At left, every pixel within a local neighborhood is evaluated at each possible disparity. At middle, our proposed scheme where each pixel sparsely samples the disparity space, relying on neighboring pixels to infer missing data.

Sparse Distributed Depth Sampling (SDDS). The concepts of image plane and depth (or disparity) sparsity are orthogonal in the sense that they may be used independently or in a combination. In this chapter we propose a stereo depth approach that conjugates these two separate sparsity concepts in a manner consistent with the concept of smoothness in depth estimation. We note that spatial smoothness is an ubiquitous and implicit assumption throughout stereo methods, whose relevance for algorithm design has been hitherto neglected. Moreover, even though both sparsity mechanisms represent efficiency driven approximations, we combine them into a reduced complexity depth estimation framework, yielding comparable results to exhaustive search.

One work closely related to ours is Histogram Aggregation (HA) proposed by Min et al. (2011) for efficient cost aggregation, which combines both a new pixel-wise likelihood histogram aggregation scheme along with sparse image sampling in order to drastically reduce computational complexity. The authors explored the robustness of their cost aggregation scheme across different levels of pixel sampling sparsity and discovered that for their proposal there is not a monotonic correlation between estimation accuracy and the sampling sparseness. In contrast to (Min et al., 2011), we favor robustness by deploying full template variable cost aggregation instead of single pixel intensity photo-consistency. To overcome the related performance challenges we couple our template variable cost aggregation with a depth sub-sampling framework, which achieves both lower computational complexity and higher processing speed, while simultaneously improving accuracy.

Distributed Depth Sampling

The smoothness constraint in stereo estimation confers a strong correlation between the spatial proximity of neighboring image pixels and the spatial proximity of their corresponding 3D points in the observed scene. Accordingly, having each pixel evaluate all possible depths using a photo-consistency measure that implicitly enforces such smoothness constraints, leads unequivocally to the realization that there is significant redundancy in the computation of local stereo methods.

Key Insight. *If we assume a neighboring set of pixels share the same depth, it suffices to evaluate any single one of those pixels at the correct depth to identify a valid estimate for the entire set.*

In this section, we propose to modify the traditional (exhaustive) *search strategy* for depth estimation into a structured and sparsely spatially distributed search scheme. Namely, lets assume, without loss of generality, a neighborhood of pixels $\aleph = \{p_i | i \in [1, \ldots, N]\}$, which satisfies the smoothness constraint. Let us further assign to each p_i a depth offset value $o_i \in [1, \ldots, N]$, such that $\{o_i \neq o_j | \forall i \neq j\}$. In order for the neighborhood of pixels $\aleph$ to *collectively* explore an entire consecutive set of depths $\mathcal{D}= \{d_k | k \in [1, \ldots, D]\}$, where $D \geq N$, it suffices to assign to each p_i a subset of depths $\{d^i_j = d_k | k = j * N + i \ , \ j \geq 0 \ , \ k \leq D\}$, where the union of the subsets for pixels in $\aleph$ equals the entire depth range. In this way, each pixel will search a sparse set of depths where the number of total depths is bounded by the ratio D/N and the depth offset between consecutive samples is N.

We denote this new depth sampling scheme *Distributed Depth Sampling* (DDS). In order to extend DDS across the entire image we may simply tile the image with this sampling pattern, which ensures that in each local stereo window the set of sampled depths equals $\mathcal{D}$. In this way, depth assignment is based on the traditional Winner-Take-All (WTA) selection within a vicinity centered on a given pixel. Note that for pixels near the periphery of their sampling pattern, their WTA vicinity encompasses pixels belonging to neighboring sampling patterns. Accordingly, the distribution of offsets within the search patterns, may introduce bias into our estimation. We have found that randomly determining a fixed pattern to be repeated across the entire image generally provides robust results and minimizes the bias. The reduction in computational complexity afforded by DDS is proportional to the size of the depth distribution neighborhood. Hence, we can define a square neighborhood of side length $M = 10$ for an effective quadratic reduction in computational complexity of two orders of magnitude.

Figure 3.2 gives the raw depth maps of DDS for Midlebury benchmark (Scharstein and Szeliski, 2002). We can see discontinuous regions (e.g. edges) become very jagged when $N > 5^2$. One

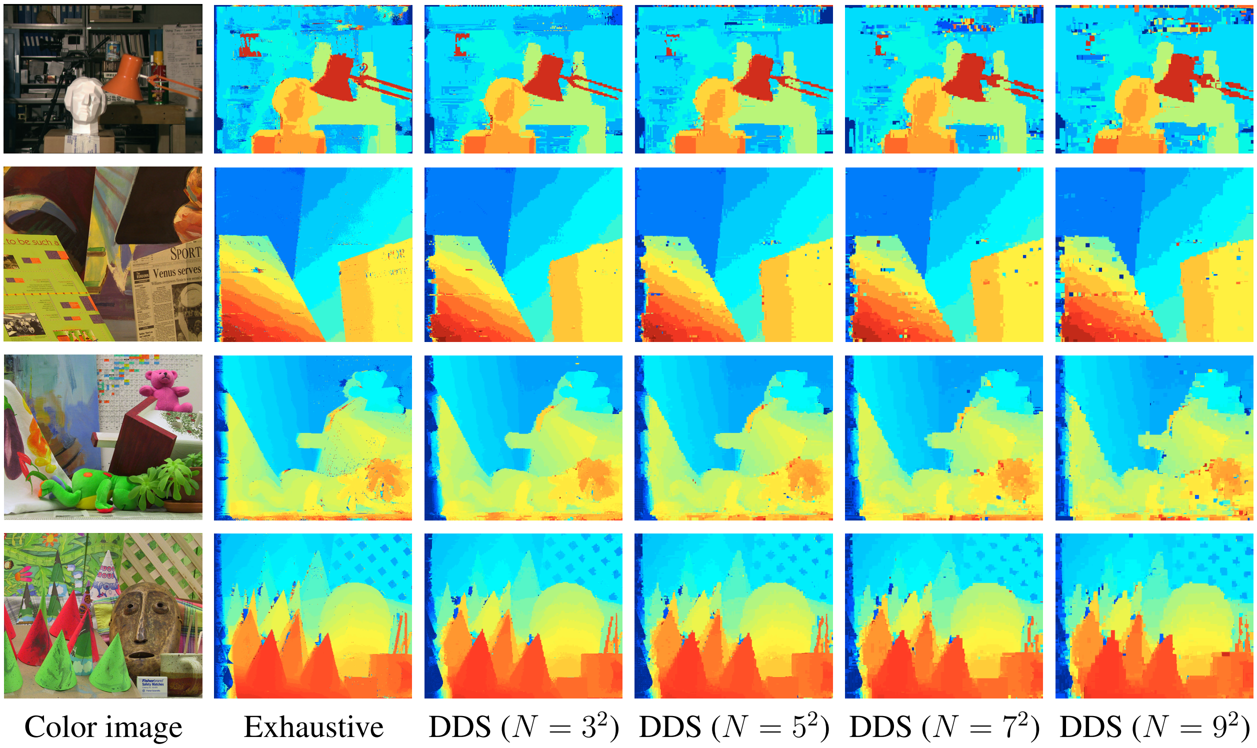


Figure 3.2: Comparison for raw disparity maps for DDS. From left to right: color images, raw disparity maps for Exhaustive Search and DDS with neighborhood 3^2, 5^2, 7^2, and 9^2.

reason is that DDS uses the fixed depth distribution pattern for all patches. And another reason is that DDS assumes that pixels in the neighborhood share the same depth, which is too strong to be satisfied in large neighborhoods.

The proposed DDS scheme will provide reliable depth estimates only when the neighborhood used for depth exploration covers a single fronto-parallel surface. Moreover, while DDS indeed provides a remarkable performance to cost ratio, up to this point we have only considered the case where the depth range is larger than the number of pixels in the depth distribution neighborhood $\aleph$ (i.e. $D \geq N$). In order to improve upon the efficiency of DDS we have explored the opposite scenario where $D < N$. A straightforward solution is to apply redundant depth sampling within the DDS neighborhood. Instead, we have incorporated spatial sparsity into our approach and in doing so we have developed a more efficient and accurate generalization of DDS, which is described in the following section.

Sparse Distributed Depth Sampling

In this section we extend our DDS method to be able to leverage the spatial sparsity concept proposed by Min et al. (2011). The resulting sparse distributed depth sampling (SDDS) approach can be summarized as follows.

1. We define a set of neighborhoods (with possible overlap) that cover the entire image. For each of these neighborhoods:
   (a) We randomly select individual pixels to be evaluated each at a single specific depth hypothesis until the entire depth range $\mathcal{D}$ is sampled without redundancy, i. e. each depth is sampled once.
   (b) Step 1 (a) is performed k times for the neighborhood to obtain a consensus on a reduced set of representative depths for it.
2. Then we spread a regular sparse pattern of seeds across the entire image and for each seed we evaluate the joint set of representative depths of all the neighborhoods to which the seed belongs.
3. The depth estimate for each of the seed pixels is the one with minimum cost among those evaluated in Step 2.
4. The depth estimate for non-seed pixels is obtained through proximity and photo-consistency weighted voting among all seeds in their vicinity.

Our cost aggregation deploys robust symmetric weight aggregation during the windowed matching (steps 1(a) and step 2). The remainder of this section discusses in greater detail the mechanisms and design decisions enabling our SDDS approach.

Adaptive Weight Cost Aggregation

We rely on adaptive weight cost aggregation (AW) in a similar manner to (Yoon and Kweon, 2006), as it has been shown to be a highly accurate and discriminative local photo-consistency aggregation framework. We mitigate the computational burden of using AW by effectively reducing the number of times such template vs. template evaluations need to be made. Note that this in no way precludes the use of constant time weighted aggregation approaches such as the ones discussed in Chapter 2. Recall that for adaptive weight based stereo, the matching cost for pixel p with depth d is computed as:

$$cost(p, d) = \frac{\sum_{q \in N(p)} w(p, q) w(\bar{p}, \bar{q}) e(q, \bar{q})}{\sum_{q \in N(p)} w(p, q) w(\bar{p}, \bar{q})} \tag{3.1}$$

where $\bar{p} = p - d$ and $\bar{q} = q - d$ are the corresponding pixels of p and q on the matching image, while $N(p)$ is the set of p's neighbors. The pixel similarity measure $e(q, \bar{q})$ used in this work is the AD-census cost (Mei et al., 2011). The adaptive weight $w(p, q)$ for q with respect to the center p, combines the color distance (in CIELAB space) and the spatial distance as defined by Equation (2.2).

The two most time consuming operations are the pixel-wise comparison $e(q, \bar{q})$ and the weighting $w(p, q)$. Even though very efficient implementations of the AD-Census similarity measure can be achieved (Mei et al., 2011), we note in general that pixel-wise weight computation $w(p, q)$ is more efficient than similarity computation $e(q, \bar{q})$, due to the overhead associated with the census transform estimation in addition to Census and SAD aggregation. Other operations like depth sorting could be ignored with respect to these two operations. To analyze the time complexity of AW we define Q to be the number of pixels in the image, D the number of tested depth hypotheses, and W the side length of a square matching window. Then the complexity of the exhaustive adaptive weight stereo is $QDW^2(2 * u_w + u_m)$, where u_m and u_w are basic complexity for pixel-wise comparison and weighting. Here, we reduce the window size W by sparse spatial sampling. Let $s \in [1, \ldots, W/2]$ be the sampling step indicating the distance between samples along each image direction, then sub-sampling reduces the total complexity by a factor of $1/s^2$. Note that, such spatial sub-sampling within the photo-consistency measure assumes smoothness between pixels separated by $L = s/2$ pixels. This formulation implies that by performing photo-consistency using every consecutive pixel (i.e. $s = 1$) we are in fact assuming a smoothness level of $L = 0.5$ pixels (i.e. the strong spatial correlations only extend from the center of the pixel to each edge along both directions). Depending on the smoothness level assumed for a given image scene, the value of s can be reasonably controlled. We empirically determined a value of $s = 4$ represents a good trade-off between accuracy and efficiency.

Representative Depths of a Neighborhood

For each local patch, we randomly select D (equal to the size of the depth set) pixels, assign them different test depths and compute their costs according to Eqn. (3.1). Next, we sort all the estimated costs and assign to each depth a score inversely proportional to its order. By repeating this sampling and aggregating the scores for each depth, we obtain a reliable profile of scores, which approximates the profile of the likelihoods of all depths within the local patch structure. For example (as illustrated in Figure 3.3), if we only find one peak in the profile, since weights are computed from randomly selected pixels, it is highly probable that all the pixels in the local patch share the same depth. Similarly, if there are two peaks, the local patch may be on the boundary of two objects, the absence of distinguishable peaks suggests the complex scene, which may not follow the smoothness assumption.

Let c_i be the cost for the ith depth after a single sampling, and o_i is the position of the ith depth in the sorted sequence of the cost of the depths, then each depth is scored as $1/o_i$. After sampling k times, depths with total scores greater than T_s (pre-defined threshold, e.g. $T_s = 1.2$ when sampling 4 times) will be treated as the *representative depths* for the current local patch. Given that sampled pixels are selected randomly, there is in fact a possibility of failing to find any correct depths. However, in practice, *SDDS* works well for patches containing three or less structures.

It can be seen that if all all the sampled pixels within a local patch belong to a single fronto-parallel surface (i.e. they all have the same depth), the probability of identifying the correct depth is $\texttt{P}= 1$ given that all depths of $\mathcal{D}$ are sampled. Accordingly, after the kth sampling, the final score for the correct depth would be k. Hence for patches from a single surface, we will always find the

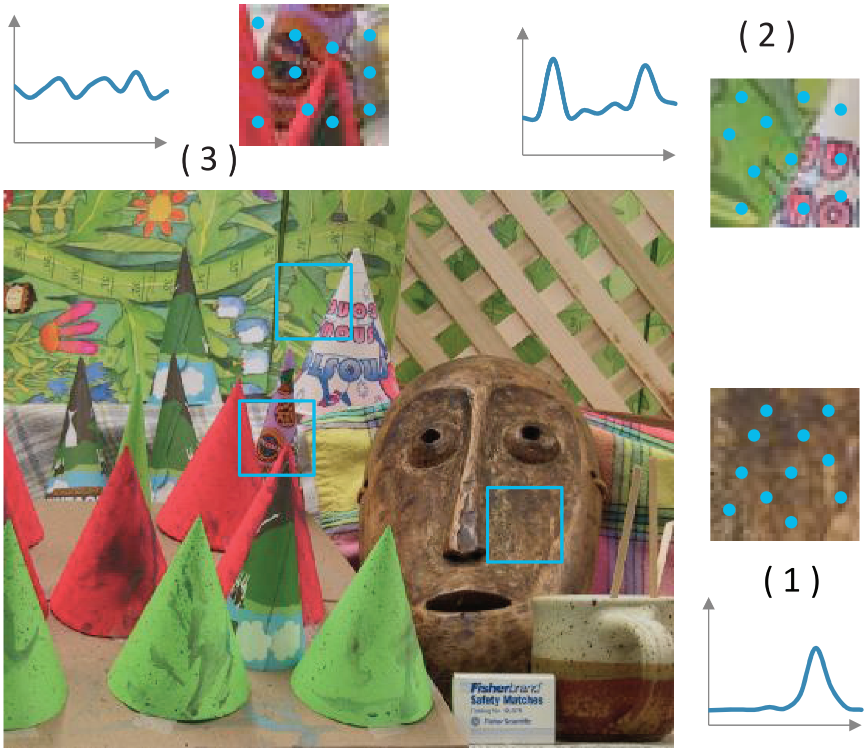


Figure 3.3: Disparity sampling analysis. Case 1: only one peak of the score profile for continuous surfaces; case 2: two peaks for simple discontinuous region (only two different surfaces); case 3: no distinguish peaks for patches with complex structures.

correct depth (i.e. one of the samples will be correct and we are certain this sample will have the highest score). This argumentation is predicated on the typical assumptions made in stereo that for a given pixel, the correct depth will get the smallest cost, or alternatively, that this minimum cost corresponds to the best possible depth estimate. This implies that when a pixel is evaluated at its correct depth hypothesis, it will be within the depths with the smallest cost, hence o_i will be small after sorting the results of all depth values within the patch. It is further assumed that the positions o_i of the other (incorrect) depth estimates within the patch are randomly distributed, which is true for all regions with a well defined global cost minimum. In all other cases the stereo decision is ambiguous. Hence we exclude them from our analysis, although in practice even those cases work robustly. For patches containing multiple surfaces we chose the threshold T_s to ensure a false positive rate (wrong depths selected into the set of interest depths) of about 2%. Please, note that any wrong depth being part of the interest depths does only slightly influence computational performance but has no quality implications.

Sparse Seed Evaluation and Propagation

The distributed attainment of a *representative depth set* for each neighborhood can be seen as an efficient sampling in depth space. In the next stage of our pipeline we also perform sampling on the 2D image space to reduce computation even further. Given that neighboring pixels generally have similar matching costs for a given depth, it is reasonable for the majority of pixels to approximate their corresponding costs from a set of pre-computed cost measurements in their vicinity. This reasoning leads to the sparse sampling of the image by means of pixel ”seeds”. Seed selection may be based on color similarity or spatial proximity. However, a good color-based sampling usually needs additional processing like clustering or segmentation, which may be prohibitive in cost for local stereo methods. Accordingly, we uniformly distribute m seeds in the image, and for each depth d belonging to the joint set of *representative depths* of the seed, we compute their adaptive weight costs $c_{s_1,d}, ..., c_{s_m,d}$. For the ith remaining pixels, their costs $c_{i,d}$ are computed by weighted

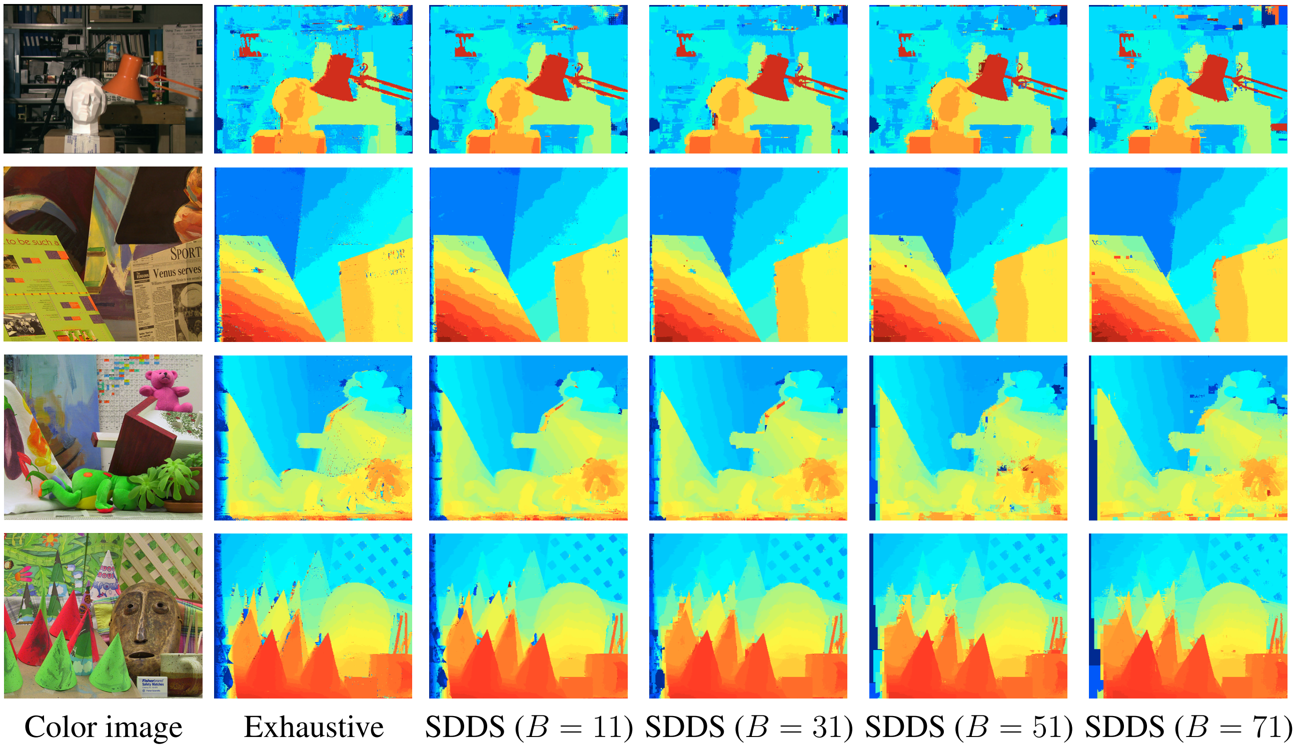


Figure 3.4: Comparison for raw depth maps for SDDS. From left to right: color images, raw depth maps for Exhaustive Search and SDDS with block size 11, 31, 51, and 71, and the default sampling time is 4.

aggregation of neighboring seeds

$$c_{i,d} = \frac{\sum_{s_j \in N(i)} w(i, s_j) c_{s_j,d}}{\sum_{s_j \in N(i)} w_(i, s_j)} \tag{3.2}$$

where $N(i)$ is the set of seeds within the ith pixel's proximity and w_{i,s_j} is the adaptive weight between the ith and s_jth pixels. After cost aggregation, the candidate depth with minimum cost will be chosen as the final depth.

Figure 3.4 compares the raw depth maps of SDDS against the exhaustive search. In contrast to the results of DDS (Figure 3.2), discontinuous regions (edges) are preserved by SDDS with almost the same quality as the exhaustive search, which means the selected seeds are able to maintain both main structure and fine details of the local patches.

Computational cost analysis

In this section, we analyze the total computational cost of our SDDS approach. As discussed in Section 3.2.1, the complexity for exhaustive adaptive weight based stereo and corresponding sub-sampled aggregation stereo are $C_{EX} = QDW^2(2u_w + u_m)$ and $C_{SP} = QDW^2(2u_w + u_m)/s^2$, where Q is the total number of pixels, D is the size of the depth set, W is the size (width and height) of matching window, s is the spatial sampling ratio inside the matching window, u is the unit computation cost, u_w and u_m are complexity for pixel-wise weighting ($u_w \approx 2u$) and pixel-wise matching cost ($u_m \approx 5u$). In our algorithm, when applying sampled aggregation, the computational cost C_1 to select the representative depths, the cost C_2 for aggregating the seed costs, and the computational cost C_3 for aggregating the remaining pixels' costs are:

$$\begin{aligned} C_1 &= kQDW^2(2u_w + u_m)/(\mu s^2 B^2) && (3.3) \\ C_2 &= mQD_cW^2(2u_w + u_m)/(\mu s^2 B^2) && (3.4) \\ C_3 &= nQD_c u_w/\mu. && (3.5) \end{aligned}$$

where B is the size of local patch, D_c is the average size of selected candidate depths (≈ 3), $1 - \mu$ is the overlapping ratio of neighboring patches, k is the number of randomly sampled times, m is the number of seeds per local patch, and n is the number of neighboring seeds for cost aggregation. Accordingly, the total complexity is $C_{SDDS} = C_1 + C_2 + C_3$. Substituting the default values used in our experiments (see Table 3.1), and assuming the size of the original depth set $D = 60$, block size $B = 50$, and sampling $k = 4$ times, we find ratio of computational costs to be $\frac{C_{EX}}{C_{SDDS}} = 1096$ and $\frac{C_{SP}}{C_{SDDS}} = 69$. In fact, even when omitting the use of sub-sampling within the AW matching (i.e. $s = 1$), we obtain a computational speed-up of $\frac{C_{EX}}{C_{SDDS}} = 131$. We can see our depth sampling approach reduces the computational cost dramatically. The main reason is that our method reduces the cardinality of the tested depths per pixel ($\approx$ 3). Additionally, our technique replaces the costly and redundant matching cost computation (patch-wise comparison) by aggregating costs from neighboring seeds, which has only a cost of $2u_w + u_m$ to u_w. Comparing with another fast stereo approach HistoAggr (Min et al., 2011), whose computational cost could be approximated by $C_{HA} = QDu_m/(s_{HA}^2) + QD_{fixed}W^2u_w/(s_{HA}^2)$, where $s_{HA} = 3$ is the default spatial ratio and $D_{fixed} = D/10$ is a fixed subset size of depths. Our experiments demonstrate that our method outperforms (Min et al., 2011) by a factor of $2.8 = \left(\frac{C_{HA}}{C_{SDDS}}\right)$.

Depth Refinement

To overcome the typical minor mishaps of local stereo estimation caused for example by occlusions, we propose a voting-based refinement method inspired by the work of (Min et al., 2011). Firstly, pixels P_i^r with reliable depths d_i^r are found by left-right cross validation. Since the cost c_i^r for each reliable depth d_i^r is known after the depth computation, we define the likelihood for d_i^r as $l_i^r = 1 - c_i^r$. Notice that each reliable pixel only needs to keep the likelihood value for its final depth. When estimating the depth for an unreliable pixel P_j^u, first we check the ratio of reliable pixels within its neighborhood, if the ratio is greater than some pre-defined threshold T_N, then we build a voting list for all the reliable depths occurring in its neighborhood, and the voting score is computed by summing weighted likelihoods of the reliable neighbors:

$$v_{j,d_i}^u = \sum_{P_i^r \in N_r(P_j^u)} l_i^r w(P_j^u, P_i^r), \tag{3.6}$$

where $N_r(P_j^u)$ is the set of reliable pixels within P_j^u's neighborhood N_r, and the weights $w(P_j^u, P_i^r)$ are computed by Equation 2.2. Also, for each candidate depth d_i, its confidence f_{j,d_i} is determined by the maximum value of corresponding weight $w(P_j^u, P_i^r)$. Then the depth with maximum confidence v_{j,d_i}^u greater than threshold T_F (initially $T_F = 0.5$) is chosen as the final depth d_j^r. If no candidate depth has good confidence, the minimum depth will be set as the final depth (similar to filling holes with the background). Finally, P_j^u becomes a new reliable pixel P_j^r, and its new likelihood is computed as

$$l_j^r = \frac{v_{j,d_j}^u}{\sum_{d_i = d_j} w(P_j^r, P_i^r)} \tag{3.7}$$

We repeat the above procedure for all unreliable pixels, and gradually decrease the threshold T_F if all the unreliable pixels cannot find enough reliable neighbors. The detailed algorithm is shown in Algorithm 1. Notice that the goal of this paper is to more efficiently perform the local depth estimation without loss of accuracy. We utilized similar post processing to that in (Min et al., 2011) in order to better contextualize our results with respect to the Middlebury benchmark and make both approaches comparable. Improved refinement procedures would elevate the ranking of our implementation but that was not our emphasis.

Algorithm 1 Voting Refinement

```
Parameters:
$\{P^r\}$ and $\{P^u\}$: set of reliable and unreliable pixels
$d^r_i$: depth for reliable pixels $P^r_i \in \{P^r\}$
$l^r_i$: likelihood for reliable pixels $P^r_i$ with depth $d^r_i$
$N_r(P^u_j)$: set of neighboring reliable pixels for $P^u_j$
$T_N$: Threshold of size of neighboring reliable pixels
$T_F$: Threshold of confidence

Algorithm:
while $\{P^u\} \neq \emptyset$ do
    for $P^u_j \in \{P^u\}$ do
        if $|N_r(P^u_j)| < T_N$ then
            continue
        Initialize voting list $v^u$ for candidate set $\{d\}$
        for $d_i \in \{d\}$ do
            $v^u_{j,d_i} = \sum_{P^r_i \in N_r(P^u_j)} w(P^u_j, P^r_i) l^r_i$
            $f_{j,d_i} = \max_{P^r_i \in N_r(P^u_j)}(\{w(P^u_j, P^r_i)\})$
        if $\max(\{f_{j,d_i}\}) < T_F$ then
            $d_j = \min(\{d\})$
        else
            $d_j = \text{argmax}_{d_i}(\{v^u_{j,d_i}\})$
            $l^r_j = v^u_{j,d_j} / \sum_{d_i = d_j} w(P^u_j, P^r_i)$
            $\{P^u\} = \{P^u\} - P^u_j$
            $\{P^r\} = \{P^r\} \cup P^u_j$
    if $|\{P^u\}|$ not decrease then
        $T_N = T_N / 2$
```

Experiments

We evaluate our methods on the stereo images from the Middlebury benchmark. Table 3.1 lists the default parameters used in all experiments, whose corresponding descriptions may be found in Section 3.2.4.

Table 3.1: Default values for experiment parameters.

Parameter	W	s	μ	m	n
default value	31	4	50%	100	20

In the first set of experiments, we investigate the quality of raw depth images. As an evaluation baseline, we use the results and performance of an exhaustive depth search with variable cost aggregation. Moreover, we omitted performing our depth and spatial sub-sampling, yielding a similar approach to that of (Rhemann et al., 2011) but using AD-Census as a photo-consistency measure. Results for DDS and SDDS, with varying cardinality of the depth exploration neighborhood $\aleph$, are compared relatively to the baseline, with all methods being executed on the same machine. For accuracy comparison, the baseline is the number of pixels with correct depths (according to the Midlebury benchmark (Scharstein and Szeliski, 2002)) generated by the baseline. From this value, the **hit ratio** is computed as the number of correct pixels estimated by our methods divided by the same corresponding quantity for the baseline method. Similarly, the **relative processing time** compares the processing times of the exhaustive baseline method and our proposals.

First, we investigate how the performance of DDS changes when the neighborhood size N increases form 3^2 to 9^2, see Figure 3.5. We find that for images with small depth set ($D <$ 20, e.g. Tsukuba and Venus), the hit ratio and processing time do not decrease significantly when $N \geq 5^2$, this is because when $D \leq N$, each pixel only tests a single depth (the sparsest case for DDS), and no further computation savings can be achieved. For large depth sets (Teddy and Cones), the relative processing time is about 0.0016 when $N = 9^2$ (approximately 625 times faster than the baseline), while DDS is still able to obtain more than 90% hit ratio. The considerable speed up for DSS is a consequence of both the quadratic reduction in depth sampling and the sparse spatial sampling used for variable cost aggregation.

Similar comparisons for SDDS are shown in Figure 3.6 and Figure 3.7 for various block sizes $11 \leq B \leq 91$ and sampling iterations $2 \leq k \leq 10$. From the hit ratio plot, we can see when $B \leq 51$, SDDS hits more than 95% percent of reliable pixels by sampling 2 times. Even for large block sizes (71 and 91), the hit ratio is greater than 90% after using four samples to build the representative depth sets. Particularly, the hit ratio for Venus images is greater than 1, which means our method could find more reliable pixels than the exhaustive method did. From the processing time comparison, we can see when the block size is larger than 51 and sampling times less than 5, the processing time is improved by 1000 times. Considering that the sparse aggregation scheme speeds up our algorithm by 16 times, the pure benefit of our depth sampling scheme is more than 60 times.

Comparing DDS and SDDS on the Cones dataset illustrates the relative processing time for DDS ($N = 9^2$) and SDDS ($B = 51$ and $k = 4$) are 0.00164 (≈ 610 times) and 0.00108 (≈ 926 times) respectively, while the corresponding hit ratios are 0.94 and 1.00. Similar results have been found for other datasets, indicating that SDDS is both faster and more accurate than DDS.

For the second set of experiments we benchmarked our performance against three recently proposed stereo algorithms aimed at improving efficiency: HistoAggr (Min et al., 2011) (pixel-wise cost aggregation), ESAW (Yu et al., 2009) (exponential step cost aggregation), and CSBP (Yang et al., 2010) (constant-space belief propagation). Our C++ with OpenMP (4 threads) implementation of SDDS was executed on a Quad-core Intel Xeon W3540 @2.93GHz with default parameters shown in Section 4.4. We used our own implementation of HistoAggr in C++ with OpenMP (4 threads). We used the author provided code for CSBP and ESAW. For ESAW we set the number of iterations to six in order to achieve comparable results (longer executions tend to improve quality at the cost of efficiency). Table 3.2 shows the Middlebury benchmark results for *raw depth map* comparison for depth error (all pixels) and processing time. Notice that our SDDS implementation is CPU based, and is the fastest among all reported CPU methods. Specific comparison with the

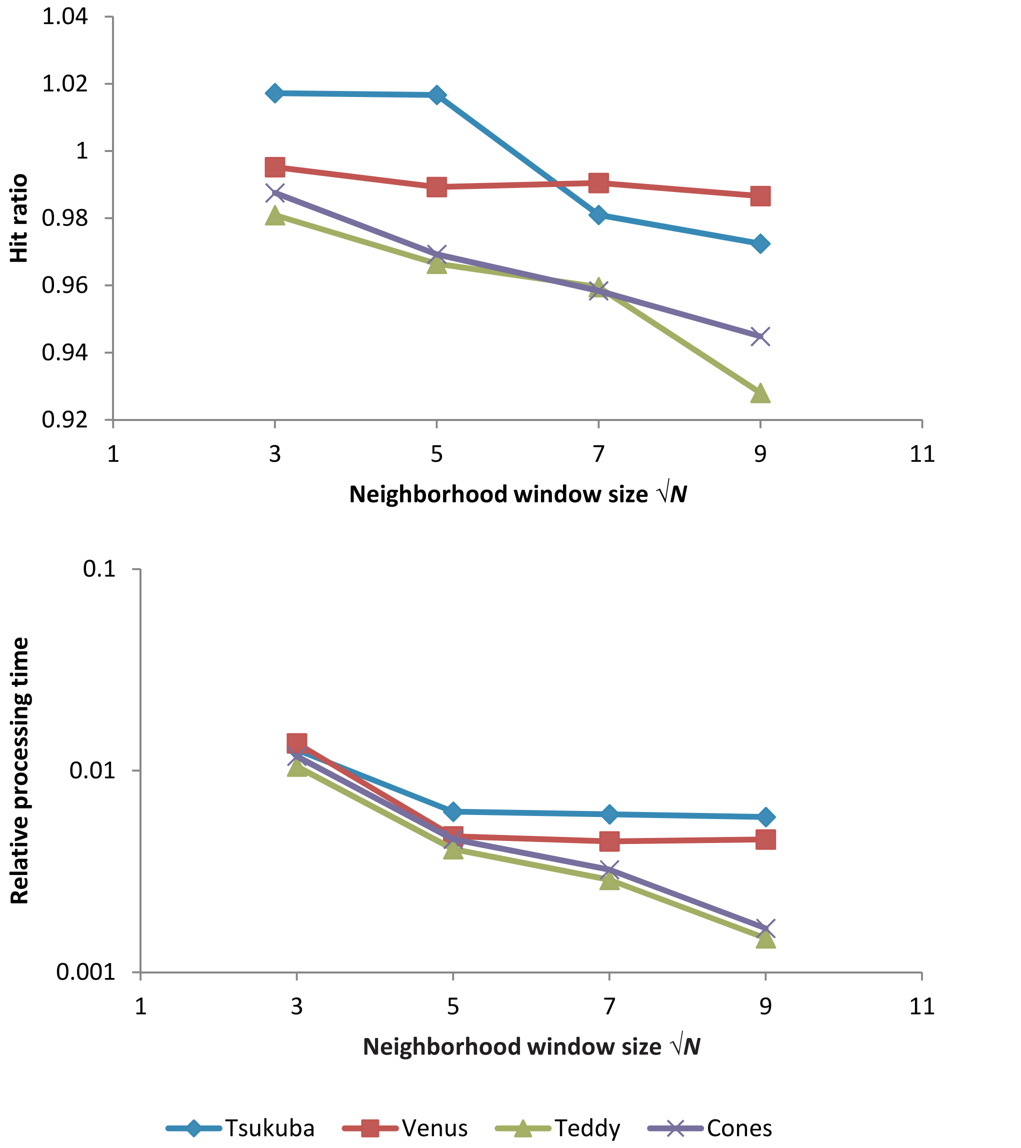


Figure 3.5: Raw depth map evaluation for DDS : Hit ratio and relative processing time

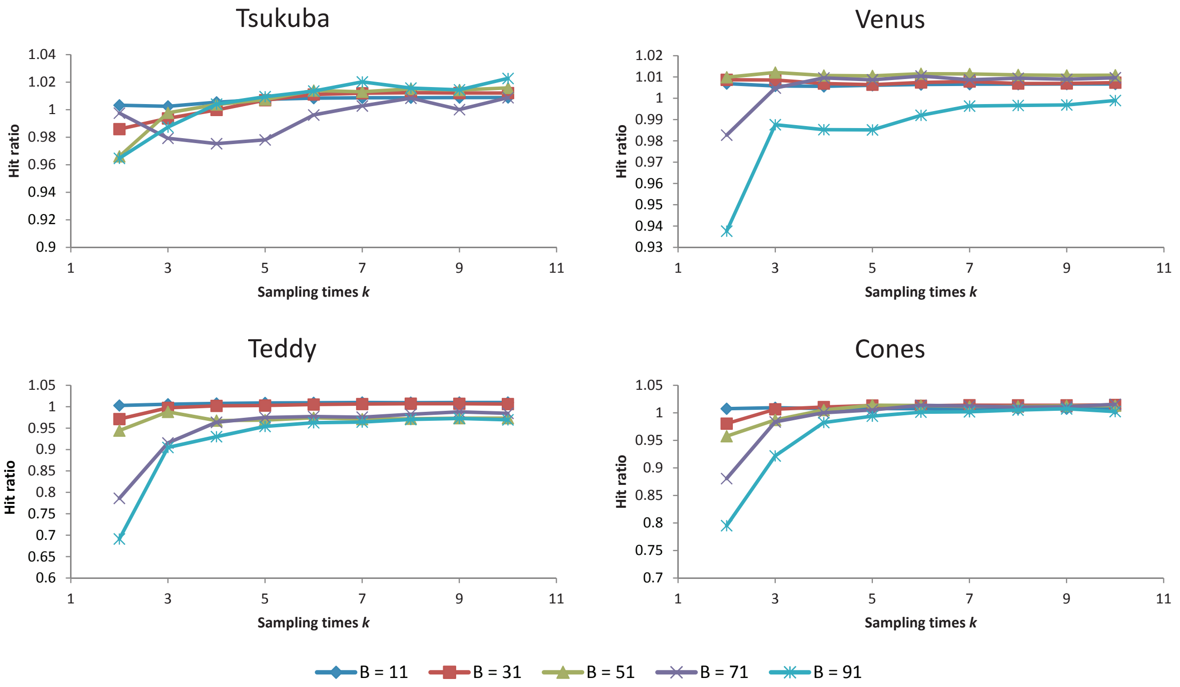


Figure 3.6: Raw depth map evaluation for SDDS: Hit ratio for various block size B and sampling times k.

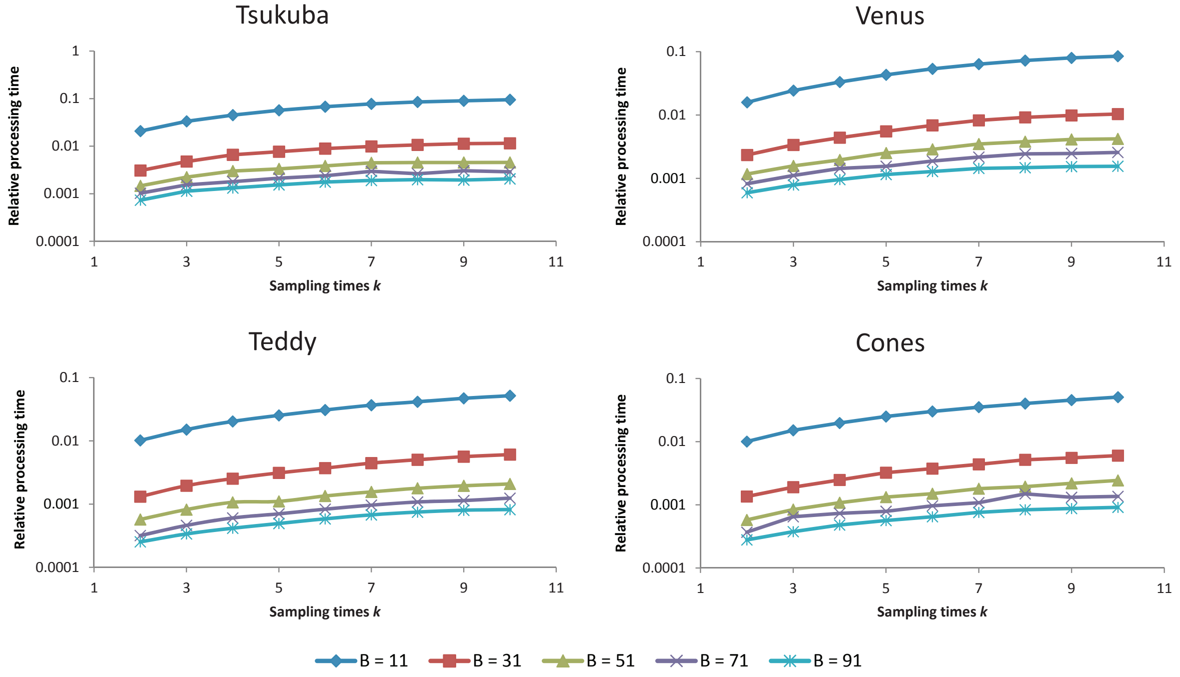


Figure 3.7: Raw depth map evaluation for SDDS: relative processing time for various block size B and sampling times k.

Table 3.2: Raw Depth map comparison for depth error (all regions) and processing time.

Algorithm	Tsukuba		Venus		Teddy		Cones	
	error	time	error	time	error	time	error	time
SDDS	6.25	0.137s	2.00	0.196s	17.4	0.339s	13.1	0.327s
HistoAggr (Min et al., 2011)	6.63	0.256s	2.15	0.392s	17.9	0.507s	13.1	0.490s
ESAW (Yu et al., 2009)	2.85	0.314s	3.75	1.097s	17.1	2.237s	17.4	2.285s
CSBP (Yang et al., 2010)	4.17	0.304s	3.11	0.406s	20.2	0.662s	16.5	0.677s

computational times reported in (Min et al., 2011) should reflect on the significance of our proposal. All faster methods on the Middlebury website deploy GPU implementations and there is no aspect of our framework that precludes this.

For the third set of experiments we compared our refined output against those reported on the Middlebury benchmark. Depthmap refinement effectively doubled our execution time given the need to generate both left and right depthmaps and also adds a processing time penalty in the range of 0.3s to 0.5s depending on resolution. Figure 3.8 shows refined depth maps for DDS ($N = 3$) and SDDS ($B = 51$ and $k = 4$), while the corresponding quantitative evaluation results are listed in Table 3.3 (results for SDDS have been uploaded to Middlebury website). We find that SDDS consistently outperforms DDS, especially for the discontinuous regions like the lamp arm, which means SDDS is more efficient for sampling depths than using fixed patterns. Also, we can see SDDS has better performance than other adaptive weight based stereo approaches like HistoAggr (Min et al., 2011), FastBilateral (Mattoccia et al., 2009), RealTimeABW (Gupta et al., 2010), and FastAggreg (Tombari et al., 2008). Our results for Venus and Teddy are even better than the conventional adaptive weight algorithm (AdaptWeight (Yoon and Kweon, 2006)). For large continuous regions with outliers having very low costs for incorrect depths, our approach usually has better results. One reason is that due to repeated random sampling scheme, scores of incorrect depths are unlikely to be high enough to be selected as the potential depths for the whole block, so their influence is limited.

Table 3.3: Refined Depth map evaluation for non occlusion (nocc), all, discontinuous (disc) regions, and average percent bad pixels (APBP).

Algorithm	Tsukuba			Venus			Teddy			Cones			APBP(%)
	nocc	all	disc	nocc	all	disc	nocc	all	disc	nocc	all	disc	
AdaptWeight	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.3	18.6	3.97	9.79	8.26	6.67
SemiGlob	3.26	3.96	12.8	1.00	1.57	11.3	6.02	12.2	16.3	3.06	9.75	8.90	7.50
SDDS	3.31	3.62	10.4	0.39	0.76	2.85	7.65	13.0	19.4	3.99	10.00	10.8	7.19
FastBilateral	2.38	2.80	10.4	0.34	0.92	4.55	9.83	15.3	20.3	3.10	9.31	8.59	7.31
RealTimeABW	1.26	1.67	6.83	0.33	0.65	3.56	10.7	18.3	23.3	4.81	12.6	10.7	7.90
HistoAggr	2.47	2.71	11.1	0.74	0.97	3.28	8.31	13.8	21.0	3.86	9.47	10.4	7.33
DDS	3.39	3.70	11.6	0.53	0.97	4.14	7.92	14.3	21.1	4.61	10.9	12.1	7.94
FastAggreg	1.16	2.11	6.06	4.03	4.75	6.43	9.04	15.2	20.2	5.37	12.6	11.9	8.24

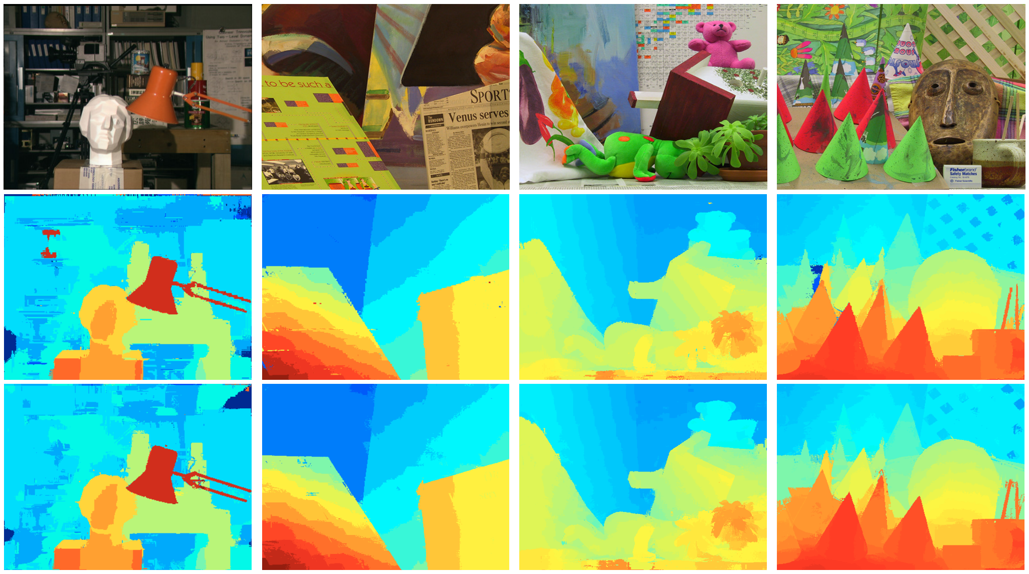


Figure 3.8: Comparison for refined depth maps. From top to bottom: color images, refined depth maps for DDS and SDDS

Discussion

We have presented an efficient and accurate modification to the standard exhaustive search paradigm in local stereo. We have achieved this by incorporating the concept of depth smoothness at the algorithm design level. Our approach incorporates sub-sampling at the depth as well as the spatial search space. The joint use of these orthogonal performance-oriented optimization mechanisms enables a dramatic reduction in the computational burden of local stereo methods, by reducing the total number of template comparisons.

One shortcoming of the SDDS is its fixed sampling scheme. No matter how simple or complex the local structure is, SDDS always samples fixed number of seeds to estimate the representative depth set, which cannot guarantee the completeness and compactness of the final candidate set. We will address this problem in Chapter 4, and provide an adaptive sampling scheme based on sequential test theory.

SDDS uses a sparse seed pattern to maintain the local structure, however, it is not most efficient. In Chapter 6 we propose a novel method to approximate and fill local structures by plane propagation.

CHAPTER 4: SEQUENTIAL OPTIMAL SAMPLING

Introduction

Dense stereo depth/disparity estimation methods commonly rely on the exhaustive enumeration of the photo-consistency cost volume attained from an *a priori* determined set of depth/disparity hypotheses. This is inherently inefficient given that, in the absence of scene structure priors, all pixels share a common hypotheses search space designed to cover the entire scene volume. One aim of this thesis is to propose a novel framework to reduce the candidate hypotheses per pixel. The reduction especially benefits high resolution stereo for modern high resolution digital cameras or satellite terrain height map estimation (with an additional computational burden due to the rational polygonal camera model (RPC) (Dial and Grodecki, 2005) during the photo-consistency cost computation).

Recent randomized (Bleyer et al., 2011) and structured (Min et al., 2011) sampling schemes are efficient and robust mechanisms for disparity estimation. The concepts of sparsity and propagation are recurring themes across these efficiency driven optimizations of the basic disparity search framework. The underlying property being exploited is that of scene structure regularity due to the assumption of local depth correlations among adjacent pixels. These assumptions are typically encoded as predetermined sampling distributions or data propagation schemes. Since these broad *a priori* assumptions are in general error prone we, instead, favor the explicit adaptive sampling of the depth (or disparity) search space to build incremental models of the local scene structure. Depth sampling schemes like Histogram Aggregation (Min et al., 2011) and PatchMatch (Bleyer et al., 2011) can be characterized by their neighborhood structure, sampling budget and the search range being analyzed. Namely, in a $M \times M$ local patch K pixels are sampled across the full search space S by computing matching cost with aggregation window W. For example, Histogram Aggregation

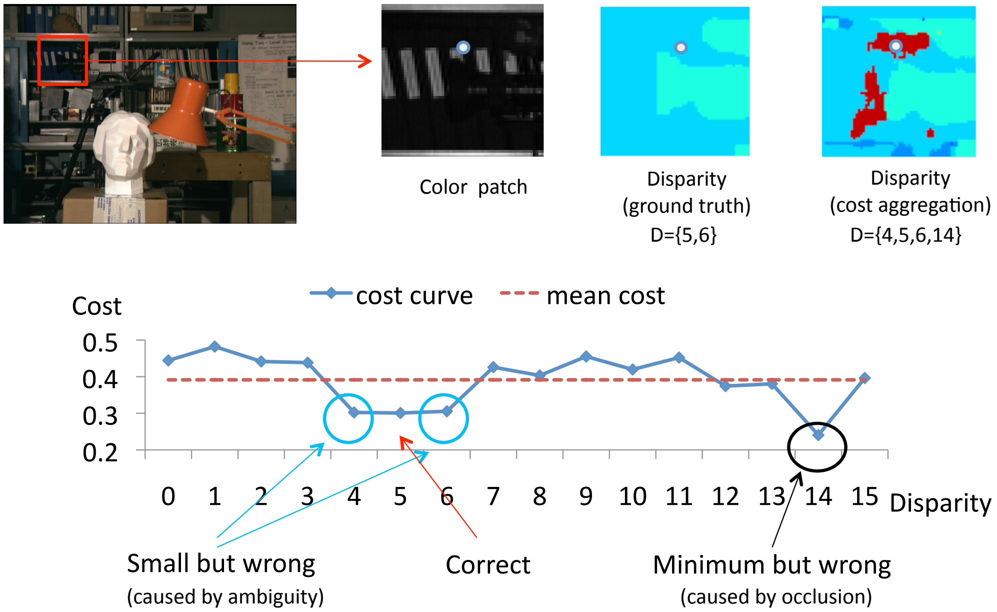


Figure 4.1: The unreliability of local photo-consistency matching. We depict an image block containing only two ground truth disparities. Exhaustive local search overestimates the disparity set.

sampled $(M/3)^2$ pixels on the entire space with pixel-wise matching, and PatchMatch is initialized by sampling M^2 pixels on a randomly selected disparity using an aggregation window W.

In this chapter, I propose an efficient sampling scheme for building an accurate model of the local disparity structure. Our sampling scheme strives to minimize the number of sampling computations while providing statistical guarantees of coverage sufficiency based on the sequential probability ratio test (SPRT). This data driven sampling scheme enables an adaptive framework for optimal depth sampling leading to a turnkey solution for reduced complexity depth sampling. SDDS proposed in Chapter 3 is also a sparse sampling framework for reducing the candidate hypotheses per pixel. For each local patch, it randomly computes the matching cost for one pixel per disparity, and then finds a candidate set by repeated sampling with a constant number (3 or 4) of iterations. In contrast, the optimal sampling scheme proposed in this chapter balances sampling completeness and efficiency in a statistical framework, enabling the adaptive termination of local structure sampling.

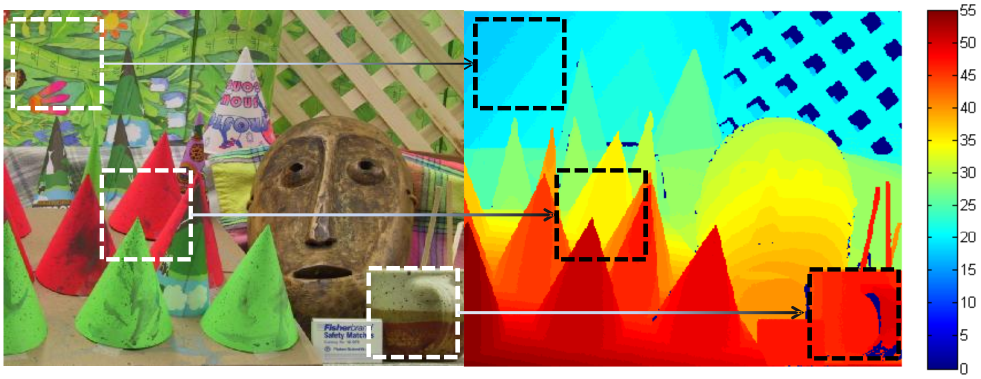


Figure 4.2: Three regions with variant local scene structure and different sampling requirements. Note that texture variability is not a robust cue for predicting scene complexity.

Properties of Stereo Sampling Schemes

Stereo depth sampling aims to attain a representative scene structure by exploring the photo-consistency cost volume. A sampling operation refers to determining a pixel's depth estimate from the enumeration of its cost function across the entire depth range. Our focus is on sampling schemes for representing the local scene structure as *a reduced set of candidate disparities* D, with D being a sufficient representation of the local structure i.e. contains all distinct disparities in a local neighborhood. Conversely, we aim for an efficient sampling scheme which *minimizes the number of sampling operations* while sufficiently sampling.

It is well known that local photo-consistency is a noisy measurement. In the absence of a data discrimination framework, noisy depth observations will be mistakenly added to the candidate disparity set D (see Figure 4.1 as illustration) requiring robustness to achieve an effective sampling scheme. A naive sampling scheme is to specify a fixed sampling ratio and perform random sampling (RS) for a predetermined portion of pixels. Such open loop sampling disregards the observed local structure and may be arbitrarily inefficient or insufficient, i.e. over-sampling in simple and flat regions while under-sampling in regions with complicated and overlapping structures. The structure of smooth regions (like the top block in Figure 4.2) is easy to recover using a few samples, while the patch with complex structure (the middle patch in Figure 4.2) requires much more samples. Furthermore, adjusting the sampling rate based on the color information (such as working on superpixels or image segments (Klaus et al., 2006)) is error prone as, in general, there is not a correlation between appearance variability and structure variability. Our adaptive sampling scheme overcomes these limitations and complexities.

The compactness of the local disparity set D refers to its cardinality and is related to the robustness of our sampling operations. We build an incremental model of the local disparity candidate set D and rely on the sequential probability ratio test (SPRT) to determine an optimal stopping criteria for the random sampling within each local neighborhood.

Sequential Probability Ratio Test

A general SPRT model

The Sequential Probability Ratio Test (SPRT) is a pairwise hypothesis testing technique commonly used in decision theory for likelihood based hypothesis testing as for example used in efficient RANSACs (Chum and Matas, 2008). Given two hypotheses H_0 and H_1, along with sequential observations $x_k(k = 1, ..., n)$, suppose the corresponding likelihoods for these two hypotheses $P(x_k|H_{i=\{0,1\}})$ are already known. In the SPRT model, testing is controlled by the accumulated likelihood ratio L:

$$L_n = \prod_{k=1}^{n} \frac{P(x_k|H_0)}{P(x_k|H_1)} = L_{n-1} \cdot \frac{P(x_k|H_0)}{P(x_k|H_1)}. \tag{4.1}$$

Given thresholds T_1 and T_0 ($T_1 \leq T_0$), for each observation the SPRT model (as shown in Figure 4.3) will be one of following three states:

- $L \geq T_0$: stop testing and accept H_0;
- $L \leq T_1$: stop testing and accept H_1;
- $T_1 < L < T_0$: wait for a new observation.

The SPRT model is completely data driven to sample far less to reach the same conclusion as with a predetermined number of samples (Wald, 1947).

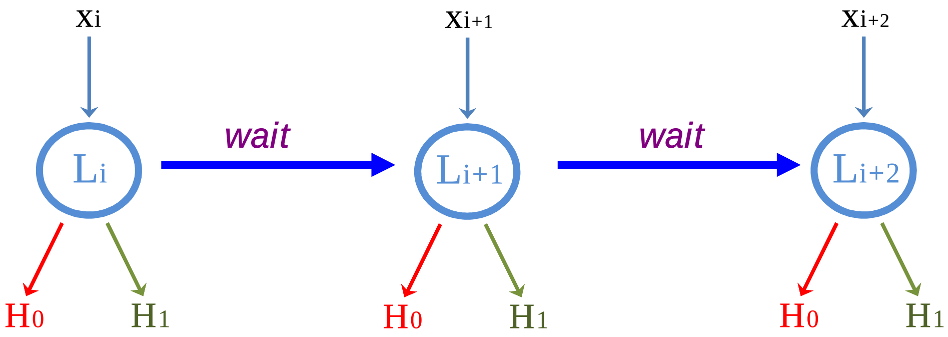


Figure 4.3: SPRT model: for each observation (x_i), the subject decides whether to accept (H_0 or H_1), or collect more information.

The thresholds T_0 and T_1 have to be determined prior to the computation by for example analyzing the behavior of the sampling sequences. Without loss of generality, we assume H_0 is the correct hypothesis. Given a sequence of arbitrary consecutive observations x_k of a correct model, for any given threshold T_0 there will be sequences that are accepted as they meet the constraint $L \geq T_0$, other valid sequences will be wrongly neglected (i.e. $L \leq T_1$ and then accept H_1). The portion of such "wrong" combinations is the error for likelihood $P(x_k|H_0)$, denoted by e_0, which quantifies the representative ability of the likelihood function L_n. Similarly, e_1 denotes the error for likelihood $P(x_k|H_1)$. Let $x = (x_1, ..., x_n)$, $R_0 = \{x : L_n \geq T_1\}$, $R_1 = 1 - R_0 = \{x : L_n < T_1\}$, and $p_i(x) = \prod_{k=1}^{n} P(x_k|H_i)$, where $i = 0, 1$. It yields that

$$\begin{aligned} 1 - e_0 &= \int_{R_0} p_0(x)dx \\ &= \int_{R_0} \frac{p_0(x)}{p_1(x)} p_1(x)dx \\ &= \int_{R_0} L_n p_1(x)dx \\ &\geq T_0 \int_{R_0} p_1(x)dx, \quad since\ L_n \geq T_0 \\ &= T_0 e_1 \end{aligned} \tag{4.2}$$

$$
\begin{aligned}
1 - e_1 &= 1 - \int_{R_0} p_1(x)dx \\
&= \int_{R_1} \frac{p_1(x)}{p_0(x)} p_0(x)dx \\
&= \int_{R_1} L_n^{-1} p_0(x)dx \\
&\geq T_1^{-1} \int_{R_1} p_0(x)dx, \quad \mathit{since}\ L_n^{-1} \geq T_1^{-1} \\
&= T_1^{-1} e_0
\end{aligned} \tag{4.3}
$$

The thresholds T_1 and T_2 are bound by errors: $T_0 \leq \frac{1-e_0}{e_1}$ and $T_1 \geq \frac{e_0}{1-e_1}$ (Wald, 1947). Thus we can predetermine the errors to make both as small as possible. By modeling the error of our likelihood models we can define a robust and efficient sampling mechanism. Next we define the sampling model for the candidate disparities.

Sequential Optimal Sampling for Stereo

Recall that a depth sampling model finds a set of candidate depth D for each local patch by sampling K pixels in the search space S and evaluating their matching costs for a given aggregation window. We design a scheme that dynamically adjusts the value of K according to the previous observations. The observation x_k is represented by the cost profile of the kth randomly selected pixel p, comprised by the matching cost for all candidate depths $d \in S$. Our matching cost is computed based on color and gradient values as in Bleyer et. al. (Bleyer et al., 2011).

In our SPRT scheme, hypothesis H_0 is that the current disparity set D is sufficient with a probability α (called α-sufficient), and H_1 is that D should be expanded to $D'(\supseteq D)$. Different from the standard SPRT scheme, our model performs an incremental test, where hypotheses H_0 and H_1 keep changing until D is accepted. $P(x_k|H_0)$ and $P(x_k|H_1)$ are the likelihoods to accept D and D' respectively. Since D' is a superset of D, we have $P(x_k|H_0) \leq P(x_k|H_1)$. Accordingly, the accumulated likelihood ratio L is monotonically decreasing for any given pair of hypotheses H_0 and H_1. Hence, we can see that our SPRT model based on evolving hypotheses is similar to

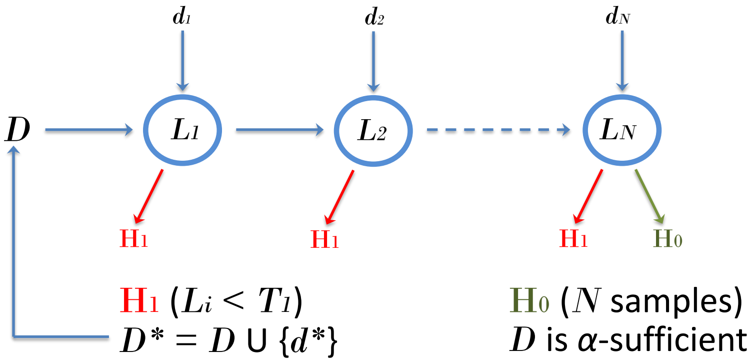


Figure 4.4: Sequential Optimal Sampling (SOS) model. Given a depth set D and a new observations d_i, if the accumulated likelihood drops below threshold T_1, D is augmented. Sampling halts after N consecutive samples without updates to D.

a "one-sided" model, which only considers whether to expand the current D by comparing the accumulated likelihood ratio L with the threshold T_1.

A candidate depth set D is α-sufficient if it has not been updated in N consecutive observations, and our optimal sampling aims to find an α-sufficient depth set with the minimum samples. Values for hypothesis errors e_0 and e_1 are attained from two user parameters: α_{suff} (sufficiency) and α_{conf} (confidence), where α_{suff} is an acceptable accuracy of the reduced disparity set (e.g. for $\alpha_{\text{suff}} = 90\%$, the reduced set contains the true depths of at least 90% of the pixels within the sampling block), and α_{conf} is an estimated probability of finding a new depth through N independent samples, e.g. our confidence on the 90% sufficiency assertion. Given the two parameters, since $\alpha_{\text{conf}} = 1 - (\alpha_{\text{suff}})^N$, we have $N = ln(1 - \alpha_{\text{conf}})/ln(\alpha_{\text{suff}})$, which is an overestimate as it assumes sampling with replacement.

We call this SPRT-based sampling scheme Sequential Optimal Sampling (SOS), which is able to sample less and provide a significantly tighter search space for each local image region. Figure 4.4 shows the flowchart of the SOS model, where given the original depth set D, the model state L (the accumulated likelihood ratio) is updated by each new observation d. To improve the sensitivity to photo consistency noise we design our likelihood function to be more tolerant to costs slightly greater than the minimum matching cost. Let $c(x, d)$ be the matching cost for pixel x at depth d, $\bar{c}(x)$ the mean cost for x across all depth hypotheses, and d^* the depth with the minimum matching cost. We define the score for each depth d as:

$$s(d, x) = \exp(-1 + \frac{\bar{c}(x) - c(x, d)}{\bar{c}(x) - c(x, d^*)}) \tag{4.4}$$

The range of the score $s(d, x)$ is $(0, 1]$, with higher scores being sought. Thus, even if the current best disparity estimate d^* is not included in the current depth set D, if there exists some $d \in D$ whose score is relatively high, we delay the inclusion of d^* into D until we have gathered more sampling evidence to mitigate spurious depth outlier estimates. Accordingly, in this new support test model, newly encountered depths will be stored in a candidate pool instead of being directly added into D, and the sample information (the coordinates and the cost curve) will be recorded and will then be used in updating the candidate depth set.

The likelihood $P(x|H)$ is defined as follows:

$$P(x|H) = \frac{\max_{d \in D} s(d, x)}{\sum_{\tilde{D}} \max_{d \in \tilde{D}} s(d, x)} \tag{4.5}$$

where $\tilde{D}$ is an arbitrary subset of the entire search space. Let $D' = D \bigcup \{d'\}$ be the superset of current reduced depth candidate set D, which will be used to replace D. Recall that H_0: D is α-sufficient, and H_1: D' is α-sufficient. Then the likelihood ratio is

$$\begin{aligned}
\frac{P(x|H_0)}{P(x|H_1)} &= \frac{\frac{\max_{d \in D} s(d,x)}{\sum_{\tilde{D}} \max_{d \in \tilde{D}} s(d,x)}}{\frac{\max_{d \in D'} s(d,x)}{\sum_{\tilde{D}} \max_{d \in \tilde{D}} s(d,x)}} \\
&= \frac{\max_{d \in D} s(d, x)}{\max_{d \in D'} s(d, x)} \\
&\geq \frac{\max_{d \in D} s(d, x)}{\max_{d \in D \bigcup \{d^*\}} s(d, x)} \\
&= \max_{d \in D} s(d, x)
\end{aligned}$$

where equality is achieved when $d' = d^*$. Then the accumulated likelihood ratio L is

$$L_n = \prod_{i=1}^{n} \frac{P(x_i|H_0)}{P(x_i|H_1)} \geq \prod_{i=1}^{n} \max_{d \in D} s(d, x_i) = L_n^* \tag{4.6}$$

The lower bound of the accumulated likelihood ratio L_n^* is used to replace L_n in the test, i.e. D will be updated if L_n^* is less than a predefined threshold T_1. Since there will be several different depths in the candidate set, we exploit the recorded sample information to accumulate the likelihood ratio for each candidate, and choose the one with the lowest value as d' to expand the current depth set D. According to the SPRT theory (Wald, 1947), the lower bound for threshold T_1 is given by $T_1 \geq \frac{e_0}{1-e_1}$ where e_0 and e_1 are errors for hypotheses H_0 and H_1 respectively. Since D' is the superset of D, which yields $e_0 \geq e_1$, we choose $T_1 = \frac{e_0}{1-e_0} \geq \frac{e_0}{1-e_1}$ to be a more strict threshold, and $e_0 = 1 - \alpha_{\text{suff}}$ according to the definition of α-sufficiency.

Bounding the Output Search Space

The number of attained disparities through sequential optimal sampling for a given image region is dependent on the observed scene structure. Accordingly, image regions covering a large number of disparities will be assigned a larger set of candidate depths. However, the efficiency of many stereo algorithms (such as Belief Propagation) is heavily impacted by the maximum size of all candidate sets. To make the candidate depth sets more balanced over the entire image, we introduced a quad-tree based recursive sampling strategy to leverage sampling patches consistent with the true depth distribution, which attains disparity sets of bounded cardinality K across the entire image. This adaptive sampling scheme guarantees that the search space of each pixel won't exceed K, so the upper bound of computation cost for stereo is limited. In order to achieve such behavior, when we detect that the current set of K depths is still incomplete, instead of adding a new depth, the patch is recursively partitioned into four sub-blocks and the optimal sampling process restarted for each new partition. Figure 4.5 shows an image automatically segmented by constrained optimal sampling. Note how complex regions are sampled by smaller windows while large homogenous regions like the background remain unchanged.

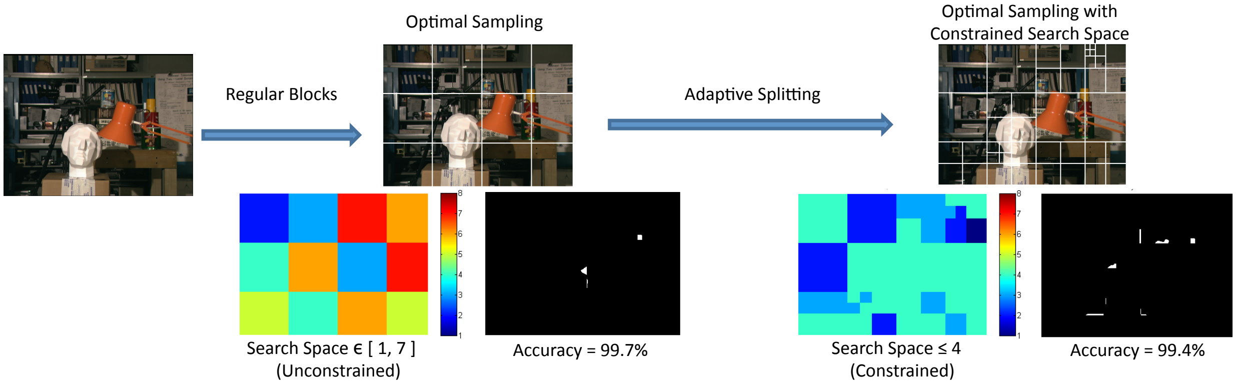

Regular Blocks
Optimal Sampling
Adaptive Splitting
Optimal Sampling with Constrained Search Space
Search Space ∈ [1, 7] (Unconstrained)
Accuracy = 99.7%
Search Space ≤ 4 (Constrained)
Accuracy = 99.4%

Figure 4.5: Optimal sampling with constrained search space. Limiting the size of the disparity set by spatial partitioning incurs in a marginal accuracy penalty at partition boundaries.

Optimal Search Space Refinement

For one local patch, there will be more than one acceptable reduced space, and the one obtained by SPRT model may not be the best combination, since the reduced search space is built incrementally, and the depths added at the beginning are only determined by few observations. Here we propose a method to refine the final search space. During the sequential test, the cost profile for each sampled pixel will be recorded in a history table.

Let $c_i(d_j)$ be the relative cost difference for depth d_j defined in Equation 4.4, and suppose the depth set found by SPRT model contains K elements, then to find the optimal depth set for the given patch is to find a set D_K with at most K elements such that the product of samples' likelihood is greater than any other combinations D', as illustrated in Equation (4.7).

$$\begin{aligned} 1 \quad &\leq \quad \frac{\prod_{i=1}^{n} \max_{d\in D_K} \exp(-1+c_i(d))}{\prod_{i=1}^{n} \max_{d'\in D'} \exp(-1+c_i(d'))} \\ &= \quad \exp(\sum_{i=1}^{n} \max_{d\in D_K}(c_i(d)) - \sum_{i=1}^{n} \max_{d'\in D'} c_i(d')) \\ &\Rightarrow \quad \sum_{i=1}^{n} \max_{d\in D_K}(c_i(d)) \geq \sum_{i=1}^{n} \max_{d'\in D'} c_i(d')) \end{aligned} \tag{4.7}$$

Suppose there is a cost table, and the element $c_{i,j}$ is the relative cost difference for the jth depth in the ith samples. The problem then becomes a binary integer programming task: to find

K columns to maximize the sum of scores $\sum_{i=1}^{n} \max_{j \in D_K} c_{i,j}$. However, such a problem does not have an $O(n)$ solution. Here we use a greedy method to approximate the optimal solution. First, we find the depth d_k that has the maximum sum of scores $\sum_{i=1}^{n} c_{i,k}$, and put it as the first depth in the candidate set. Then elements in the table are updated as $c_{i,j} = \max(0, c_{i,j} - c_{i,k})$, and find the next depth with the maximum sum of scores in the new table. Repeat K times to get the approximated optimal set, which is then compared with the original candidate set generated by optimal sampling, and choose the one with the better accumulated score as the final set. Also, when applying constrained optimal sampling scheme, the patch will be split only when the optimal set is still incomplete.

Recovering Isolated Structures

Sequential optimal sampling is designed to achieve disparity set α sufficiency for each given region. Accordingly, our coverage may be incomplete whenever there are isolated structures comprising a region coverage near or below the $1 - \alpha$ confidence threshold. The shortcoming of sampling in regular square blocks is that some isolated regions (such as the tip of the lamp in Figure 4.5) might be missed. Although the portion of such areas is much smaller with respect to the entire patch, it is still worth to recover the missed search space for some boundary sensitive applications. In this paper, the search space for the local patch will be propagated to small neighboring regions, so that isolated areas (as the errors shown in Figure 4.5) will have a joint search space that contains the correct depth. The length of the propagation is set as 10% (called propagation ratio) of the patch size. In practice such propagation results in a small dilation of each sampling block aimed at mitigating the boundary effects of our block partitioning scheme.

Experiments

We evaluate our search space reduction through SOS sampling and present results of its combination with existing stereo algorithms. For ground truth evaluation and benchmarking we used the Middlebury Stereo datasets (Scharstein and Szeliski, 2002, 2003b; Scharstein and Pal,

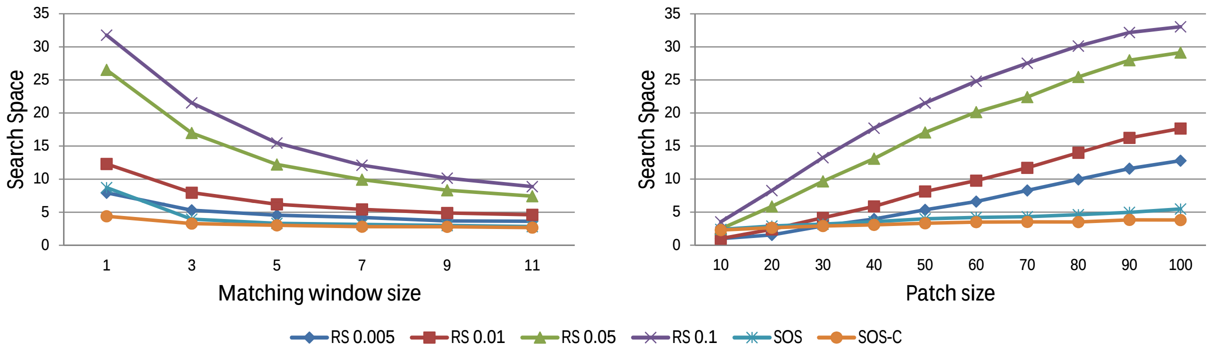


Figure 4.6: Cardinality: Sequential optimal sampling (SOS) vs. random sampling (RS) for various matching window size (left) and sampling neighborhood size (right).

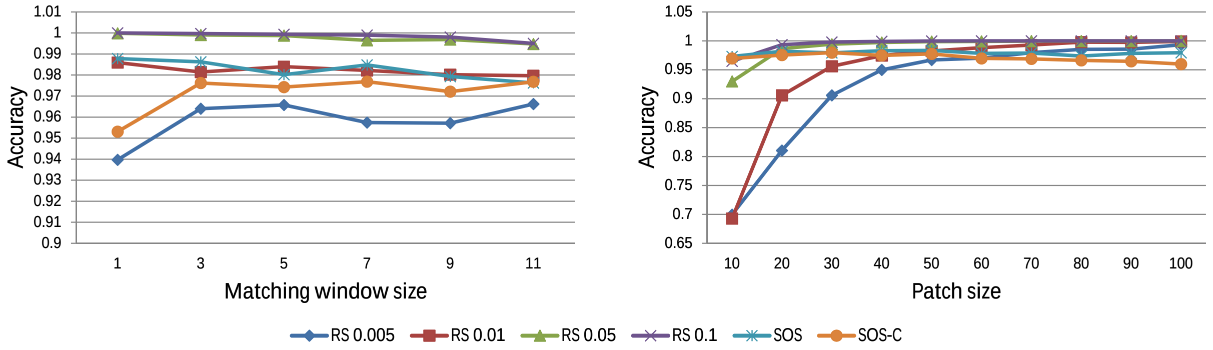


Figure 4.7: Accuracy: SOS vs. RS for various matching window size (left) and sampling neighborhood size (right).

2007; Hirschmuller and Scharstein, 2007). All algorithms were implemented in C++ and executed on an Intel Xeon CPU W3540 2.93GHz. The default aggregation window size is 3×3 for our depth sampling preprocessing step. Matching cost computation parameters are set to the default parameters proposed in (Bleyer et al., 2011). The SOS stopping parameters were set to $\alpha_{\text{suff}} = 0.90$ and $\alpha_{\text{conf}} = 0.95$.

Search Space Reduction from SOS

Initially we compare our Sequential Optimal Sampling (SOS) scheme and SOS with constrained search space $\|D\| \leq 5$ (SOS-C) against the Random Sampling scheme RS(X), which randomly selects pixels with the fixed sampling ratio $X = \{0.005, 0.01, 0.05, 0.1\}$ in each patch and uses their disparities to form the reduced search space. The reduced search space is evaluated in three aspects:

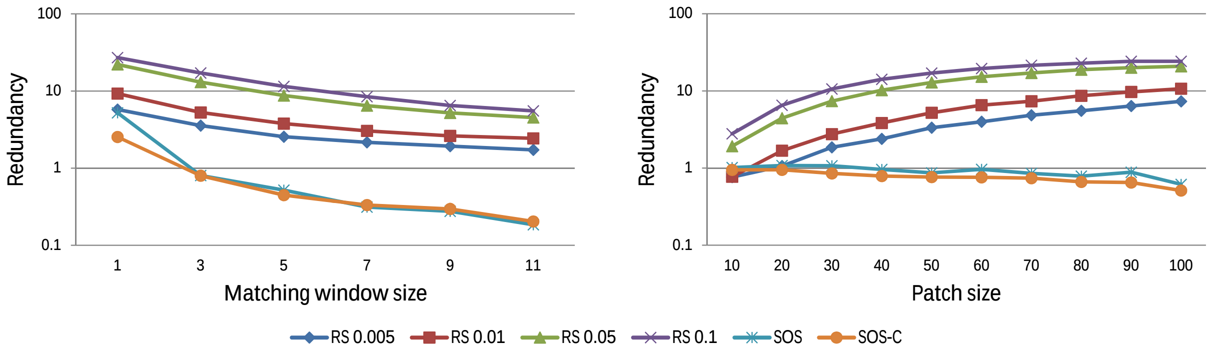


Figure 4.8: Redundancy: SOS vs. RS for various matching window size (left) and sampling neighborhood size (right).

cardinality, accuracy, and redundancy. The results of SOS are evaluated on non-overlapping blocks of default size 50×50. Our evaluation is based on the average data of the five test images: tsukuba, venus, teddy, cones, and art.

Compactness. Figure 4.6 compares the reduced search space for SOS and SOS-C against RS(X) with multiple fixed sampling ratios. Both SOS and SOS-C consistently provide smaller search spaces $M \times M$, irrespective of patch size. Moreover, our proposal found more compact disparity sets than the random sample variants geared at performing less sampling (e.g. RS 0.005). Thus our optimal sampling models effectively reduce the search space.

Accuracy. We analyze the fraction of pixels whose ground truth disparity is present in the reduced set. In Figure 4.7 we can see the accuracy of SOS and SOS-C are always above 95% with arbitrary matching windows sizes and patch sizes, showing that our optimal schemes are able to obtain a stable accuracy by adjusting the sampling ratio according to local structures, providing more flexibility than random sampling with a predefined ratio.

Redundancy Figure 4.8 measures the fraction of wrong disparities left in the reduced disparity set. Again our optimal sampling models have very small and stable redundancy (less than 1 spurious disparity added to the disparity candidate set), which is much better than any random sampling scheme. Notice that the two high accuracy (nearly 100%) schemes RS 0.05 and RS 0.1 also have huge redundancy (2% accuracy improvement with more than 20 useless depths), which intuitively is not a good balance between the efficiency and completeness.

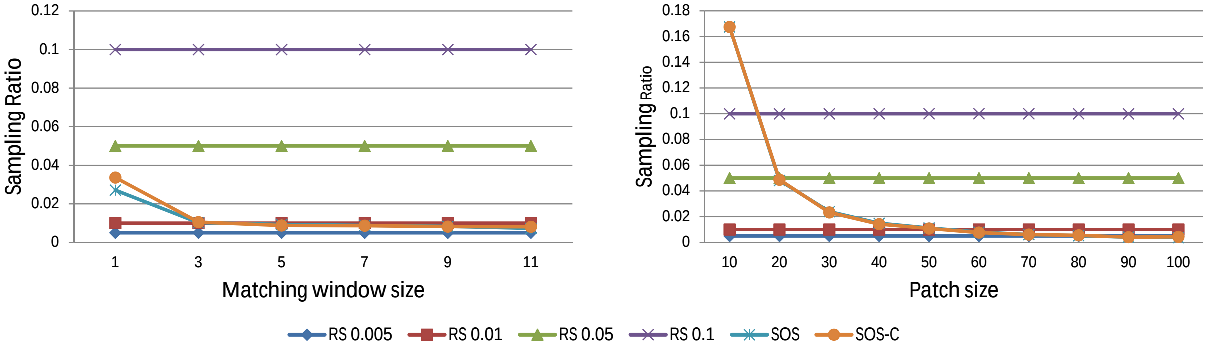


Figure 4.9: Sampling ratio comparison: SOS vs. RS for various matching window size (left) and sampling neighborhood size (right).

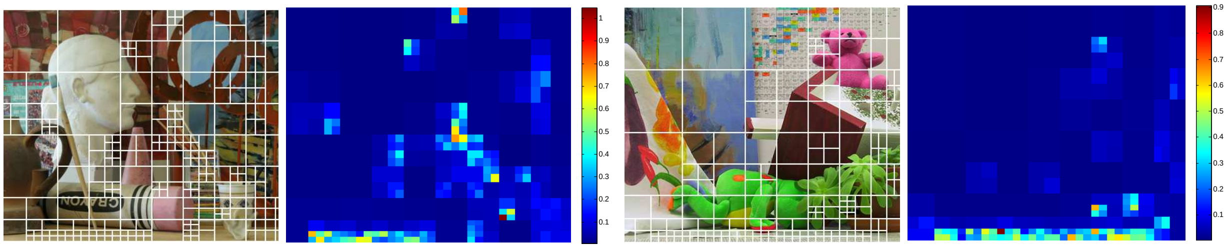


Figure 4.10: Local structures exploited by optimal sampling with constrained search space: initially, the image is divided into regular blocks (100×100 pixels), and the patch with more than $K = 5$ different disparities will be automatically divided into small regions until the number of different disparities within that patch is less or equal to K. Left of pair images show spatial partitioning while right of the pair images show the sampling ratio.

Sampling efficiency. We focus on the total number of samples required to estimate the local structure. The sampling ratios for SOS and SOS-C are shown in Figure 4.9, and we observe a stable ratio around 1%. Figure 4.10 shows the final patches generated by SOS-C and the corresponding sampling density in each of the patches. In general, the block size reveals the complexity of the local structure and all pixels in the image have a bounded (i.e. turnkey) reduced disparity set. Moreover, SOS-C successfully detects image regions with complex structure and recursively partitions said region. Accordingly, flat regions with few disparities are represented by relatively large blocks. However, we also find homogenous regions like the bottom of the image are over-segmented due to ambiguous texture, which is beyond the capability of local matching techniques.

Experiments show that the SOS schemes outperform the fixed ratio random sampling RS(X) schemes. Processing times of SOS-C for tsukuba, venus, teddy, cones and art are 21ms, 42ms, 106ms, 114ms, and 193ms respectively. Thus, SOS and SOS-C are reliable light-weight sampling schemes suitable as a stereo complexity reduction pre-process.

Stereo under SOS

We now evaluate the performance of the coupling of SOS as a pre-processing step for a variety of stereo algorithms. Namely, we compare the performance of two efficiency driven state of the art disparity sampling techniques PatchMatch (PM) (Bleyer et al., 2011) and HistogramAggregation (HA) (Min et al., 2011). As an additional baseline we include typical local and global stereo methods: Exhaustive search (EX) and Belief Propagation (BP) under the complete and the reduced disparity search space estimated through SOS (PM+S, HA+S, EX+S, and BP+S). The search space used is generated by SOS-C on 100×100 blocks with a maximum size of the disparity set of $|D| = 5$ and using propagation ratio $\gamma = 0.1$.

To enable a leveled comparison against algorithms working under the fronto-parallel assumption we modify PM for compliance to this assumption. The default window size for cost aggregation is 11, except for BP (no explicit cost aggregation). For HA (position-dependent), the spatial ratio is 3, and aggregation window is 31 (the default value used in (Min et al., 2011), which is similar

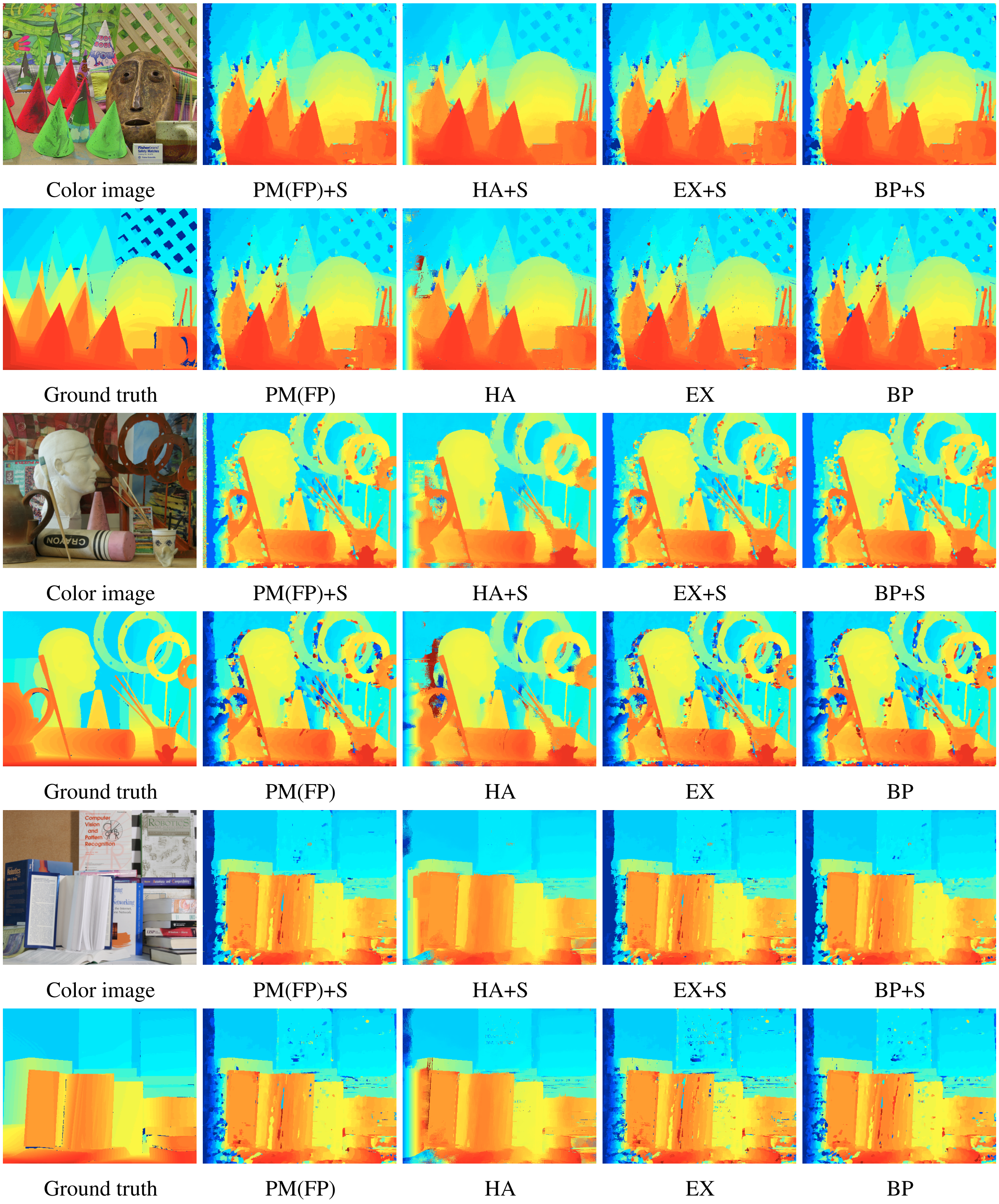
Time (s)	PM(FP)+S	PM(FP)	HA+S	HA	EX+S	EX	BP+S	BP
Tsukuba	1.76	1.91	1.40	1.80	2.54	5.13	7.88	34.00
Venus	2.49	2.80	2.10	3.09	3.81	9.48	14.57	78.30
Teddy	2.93	3.28	2.65	6.41	5.20	24.61	35.03	652.95
Cones	2.89	3.29	2.72	6.38	5.18	23.05	41.32	649.59
Art	3.26	3.72	3.70	8.21	7.76	31.22	102.38	1221.36
Books	3.11	3.5	3.06	8.13	6.27	32.34	58.33	1192.62

Table 4.1: Processing time for various stereo methods.

to aggregate cost from 11×11 pixels). For fronto-parallel PatchMatch (PM(FP)), the maximum number of iterations is four, and in each iteration the disparities are propagated starting from the top-left to the bottom-right, and then they are propagated back to the top-left. For BP, the maximum number of iterations is fifteen.

Figure 4.11 shows samples of raw disparity maps generated by the various stereo algorithms, and the corresponding processing times are listed in Table 4.1. We observe no significant quality loss between stereo algorithms under reduced and entire search spaces, while the processing time on reduced spaces is smaller than using the entire space, for PM(85%), HA(50%), EX(20%), and BP(6%).

We also compare PM(FP)+S and PM(FP) on the high resolution ($21M$) images of Kim et al. (2013) with a large candidate disparities set of 250 disparities. After our optimal sampling, the average search space is reduced to less than 10 disparities. Figure 4.12 shows the raw disparity maps for PM(FP) and PM(FP)+S, with the corresponding processing time. We also can see PM has many outliers around the bush regions which have been successfully removed by our sampling so that PM+S has much fewer outliers. While the goal of our SOS sampling scheme is to enable attainment of the same results as exhaustive disparity search from a reduced search space, in this case increased accuracy is a byproduct of our optimal local structure estimates due to the removal of ambiguous and wrong disparities.



Color image PM(FP)+S HA+S EX+S BP+S

Ground truth PM(FP) HA EX BP

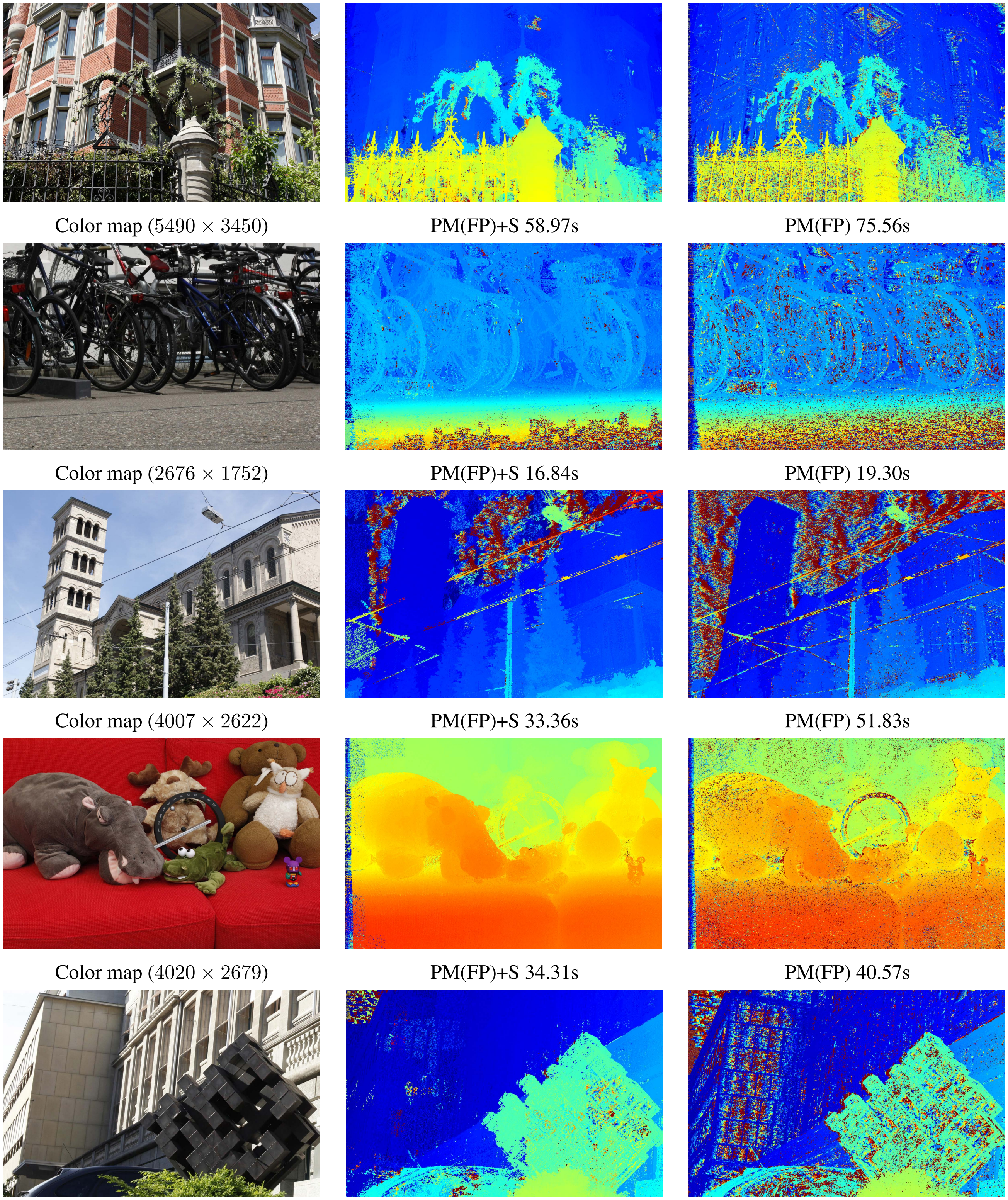
Color image PM(FP)+S HA+S EX+S BP+S

Ground truth PM(FP) HA EX BP

Color image PM(FP)+S HA+S EX+S BP+S

Ground truth PM(FP) HA EX BP

Figure 4.11: Raw disparity maps for various stereo methods.



Color map (5490 × 3450) PM(FP)+S 58.97s PM(FP) 75.56s

Color map (2676 × 1752) PM(FP)+S 16.84s PM(FP) 19.30s

Color map (4007 × 2622) PM(FP)+S 33.36s PM(FP) 51.83s

Color map (4020 × 2679) PM(FP)+S 34.31s PM(FP) 40.57s

Color map (2622 × 1718) PM(FP)+S 14.91s PM(FP) 17.94s

Figure 4.12: Raw disparity maps and processing time (10 threads) for PM(FP)+S and PM(FP) on high resolution images.

Discussion

In this chapter, we introduced a novel approach to reduce the disparity search space for stereo based on the Sequential Probability Ratio Test from sequential decision theory. Our method avoids unnecessary evaluation of irrelevant disparities for pixels of an image. Moreover, our method can be combined with a large variety of existing stereo estimation methods. As shown in our experimental evaluation, our method maintains the quality of the exhaustive disparity estimation at significantly lower computational costs.

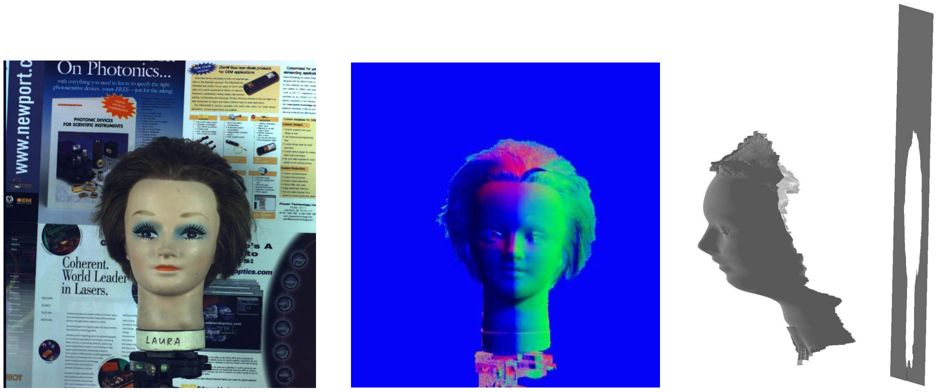
Sequential optimal sampling scheme provides an adaptive strategy to estimate the property of the local patch, where sufficient accuracy and compactness can be achieved by manually selected thresholds. One of its extension is to recover the structure of the local patch, which is further discussed in chapter 6.

CHAPTER 5: SURFLET-BASED STEREO

To improve the accuracy of stereo is one goal of this dissertation. As discussed in section 2.1.1, we need a pre-assumed local structure for aggregating pixel-wise matching costs, and the reliability of the aggregated matching costs is heavily influenced by the aggregation structure. In this chapter, we focus on the recovery of high quality local structures, and we will show that more accurate local geometry can be recovered by utilizing additional surface normal measurements, which also overcome the shortcoming of color-based stereo such as textureless regions. We address the problem of traditional surface from normal techniques first, and then propose a stereo framework integrating photo-consistency and surface normal.

Surface normals are usually obtained from illumination cues, and then used to generate high quality surfaces in Photometric Stereo (Woodham, 1980; Horn and Brooks, 1986; Frankot and Chellappa, 1989; Chow and Yuen, 2009). Under Woodham's original assumptions (Woodham, 1980) (Lambertian reflectance, known point-like distant light sources, and uniform albedo), the normal map can be obtained by inverting the linear equation $I = N \cdot L$, where I is a (known) vector of m observed intensities, N is the (unknown) surface normal, and L is a (known) $3 \times m$ matrix of normalized light directions. However, photometric stereo relies on knowing (or estimating) the directions of the light sources while making assumptions regarding their relative intensity. Light source calibration errors or other deviations from the assumed source properties may result in unacceptable distortions (Horovitz and Kiryati, 2001).

Another practical limitation of many photometric approaches is that they work with a single orthogonal image as input, so the reconstructed surfaces have no information about the absolute depths. Moreover, the surface integrability assumption used in many surface from normal algorithms only work well with single object scene geometries. Accordingly, it is hard to use them to recover



(a) (b) (c)

Figure 5.1: Multi-modal data fusion. Our approach uses (a) color images and (b) normal maps to estimate (c) reconstructed surfaces.

scenes comprising multiple objects, given that depth discontinuities are difficult to model in this context. In contrast, plane-sweep-based stereo (Collins, 1996; Yang and Pollefeys, 2003; Gallup et al., 2007) is an efficient approach to recover absolute depths based on photo-consistency measurements. However, the availability of sufficiently textured surfaces is not always fulfilled in real-world scenarios. Nevertheless, the integration of both of these complementary surface estimation approaches is a promising research path.

Recently, a novel spatial phase camera has been developed Photon-X [1], which is capable of passively recording an object's surface normals with an accuracy of two degrees of orientation (as shown in Figure 5.1b). This new sensing device measures the orientation of the last surface a viewing ray was reflected from as a function of the phase of the light wavelength captured by each CCD array element. Accordingly, we obtain a 3D orientation vector for each pixel in addition to the luminance information. In this chapter, we leverage this new sensing modality by developing a multi-modal surface reconstruction approach, which efficiently integrates measurements of the orientation and texture of multiple disjoint surfaces.

[1] Photon-X: http://www.photon-x.com/tech.html

Related work

Given the normal map $N = \{(n_x, n_y, n_z)\}$ defined over a pixel grid, we can obtain a gradient map $G = \{(g_x, g_y)\}$ where $g_x = -\frac{n_x}{n_z}$ and $g_y = -\frac{n_y}{n_z}$. The generating surface $Z(x, y)$ may then be obtained by minimizing the functional (Horn and Brooks, 1986)

$$\int\int((Z_x - g_x)^2 + (Z_y - g_y)^2)dxdy \tag{5.1}$$

Similar integral approaches (Klette and Schluns, 1996; Noakes and Kozera, 2003) share the common underlying assumption that the surface is uniformly integrable ($Z_{xy} = Z_{yx}$), i.e. the second partial derivatives are independent of the order of differentiation. Without this constraint, Equation 5.1 may have an infinite number of solutions. However, the integrability constraint is rarely satisfied in practice due to the presence of sharp edges and occlusion boundaries.

A number of approaches (Frankot and Chellappa, 1989; Hsieh et al., 1995; Karaçali and Snyder, 2003; Kovesi, 2005) project the gradient field onto a finite set of integrable basis functions $\phi(\cdot)$ in order to apply an integral method on a possibly non-integrable surface. In this case, the surface $Z(x, y)$ is represented as

$$Z(x, y) = \sum_{\omega \in \Omega} C(\omega)\phi(x, y, \omega) \tag{5.2}$$

where $\omega = (\omega_x, \omega_y)$ is a two-dimensional index, Ω is a finite set of indexes and the integrability of the surface $Z(x, y)$ is contingent on each $\phi(x, y, \omega)$ being integrable. In this scenario, the problem becomes finding the coefficients $C(\omega)$ that minimize the distances between the given possibly non-integrable surface and the approximated integrable surface. Frankot and Chellappa (1989) used Fourier basis functions, which are fast and robust to noise. Later, wavelet basis functions were adopted by Karaçali and Snyder (2003), who point out that the uniform integrability assumption will become invalid and cause significant distortions due to the impact of unknown discontinuities (edges and occlusions). They relaxed the constraint to partial integrability, and provide a global method to detect and localize unknown edges in the gradient field. The key to the edge detection is to examine the difference between the given gradient field and the gradient field obtained by the uniform integrability assumption. However, the method is complicated and the computational load in constructing the feasible gradient space descriptors grows rapidly with increasing image size. Kovesi (2005) uses shapelets (a redundant non-orthogonal set of basis functions) to enhance sharp transitions in the surface. However, the reconstructed shape is sensitive to the scale of the shapelet function being used, whose optimal determination is still an open problem.

More recent work (Chow and Yuen, 2009; Nehab et al., 2005; Lee and Kuo, 1996) uses multiple images to obtain the shape with absolute depths. Nehab et al. (2005) concluded that the prevailing errors in measured normals are low-frequency in nature, whereas positions measured by stereo triangulation contain mostly high frequency noise. The method first corrects the bias in the normals by low-pass filtering, then it formulates the corrected normals and measured positions as an optimization problem, where the objective function is a large sparse linear system. Chow and Yuen (2009) estimate the absolute depths for the singular points in intensity images by a novel sparse matching technique, then the surface is expanded from these singular points by the fast matching method. Lee and Kuo (1996) establish a unified framework for the integration of photometric ratio and stereo by employing perspective projection on a parametric surface via minimizing a cost function, which consists of a weighted sum of shading and stereo errors. Also, multi-view photometric stereo (Vlasic et al., 2009; Hernandez Esteban et al., 2008) utilizes illumination conditions, silhouettes and the shading cues to recover the absolute depths.

The aforementioned methods do not provide an explicit and efficient mechanism to detect and treat discontinuities in the scene. As a consequence, two overlapping objects may be fused together, leading to heavy distortions near the joint boundaries. To exemplify, a simple scene consisting of a bunny and a plate is shown in Figure 5.2, and the shape generated by the diffusion method for shape from normals (Agrawal et al., 2006) is compared with the ground truth and our method. We can see the plate reconstructed by the traditional method is very jagged. Moreover, since the two objects are treated as a uniform surface, it is impossible to recover the true relative distances between them. We

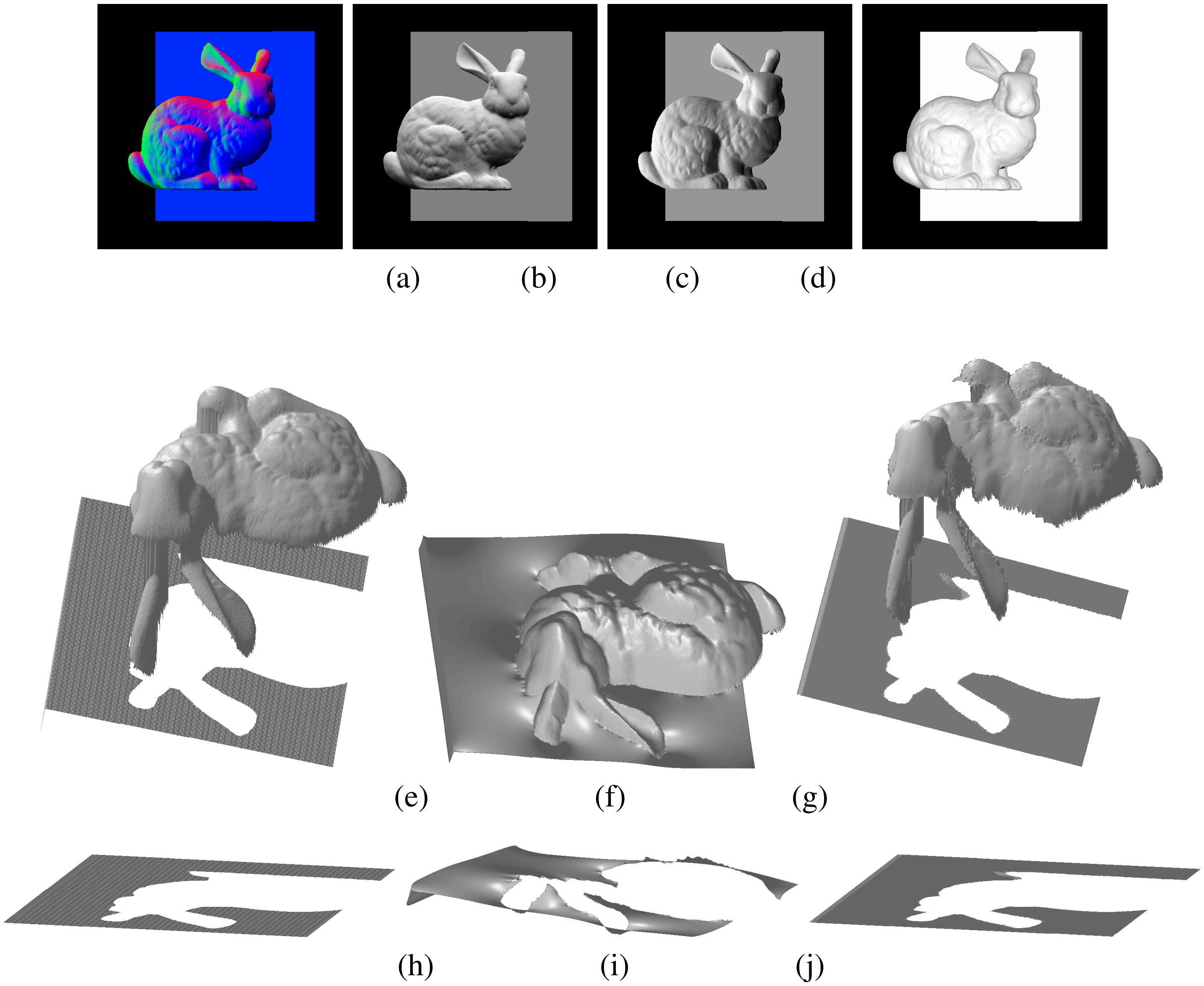


Figure 5.2: Distortion caused by discontinuities. Top: (a-d) normal map and its 3D components. Middle: scene geometry for (e) the ground truth, (f) the approach presented in (Agrawal et al., 2006) and (g) our approach. Bottom: (h-j) corresponding geometry for the background plate. Note the distortions in (i) generated by the integration approach (Agrawal et al., 2006).

also note that viewing geometry and object topology are another possible cause of surface ambiguity, as some object self occlusions (and the corresponding depth discontinuities) can not be properly reconstructed by a surface integration approach. For example, for fronto-parallel view of our scene, the bunny's ears in the ground truth model will not be directly connected to the head because of occlusions, but in the diffusion reconstructed surface the two components are smoothly connected. The methods proposed in this paper address all these limitations through a multi-modal approach.

Traditional stereo approaches rely on finding dense correspondences among images using local photo-consistency to discriminate among alternative matching hypotheses. For most stereo approaches, the photo-consistency evaluation is performed over an arbitrary neighborhood surrounding a pixel in the reference and a displaced pixel neighborhood in the matching image. The work of Yang and Pollefeys (2003) applies this concept to multiple images by applying an affine texture transformation to the matching images in order to efficiently model a plane sweeping through the space of depth hypotheses. The affine transformation corresponds to a plane orthogonal to the camera viewing axis. Gallup et al. (2007) later improved upon this approach by adaptively selecting a reduced set of main sweeping directions in accordance to sparse depth estimates from an independent structure from motion process. In that work, it was argued that aligning the sweeping direction with the main surface orientation improved accuracy by reducing the depth discretization effect caused by slanted planar surfaces not complying with the local fronto-parallel assumption. We go a step further by modeling the local surface of the matching neighborhood and using this parametric surface to sweep the space of depth hypotheses. This provides more accurate and robust depth estimates than the aforementioned plane sweeping approaches, as will be presented later in Figure 5.4.

Natural Cubic Spline Patches

This section presents the method to estimate local surface geometry (called surflet) in the neighborhood of a single pixel based on surface normal information. We model surflet as a continuous parametric function, which can be approximated by a set of intersecting 3D cubic splines defined over an image patch. Moreover, we use the surface normal measurements to formulate a linear system of equations from which we can determine least square depth estimates for each pixel in the surflet being considered.

For a surflet of size $k \times k$, the pixels in the same row or same column form k horizontal and k vertical curves. Note that the terms horizontal and vertical refer to the orientation on the image, as these are actually 3D curves generated by the object surface. Suppose the ith horizontal curve C_i^h consists of points $P_{i,j} = [x_{i,j}, y_{i,j}, z_{i,j}]$, $(j = 1, ..., k)$. Then, the segment between point $P_{i,j}$ and

$P_{i,j+1}$ can be approximated by a natural cubic spline as follows:

$$\begin{aligned} X_{P_{i,j},P_{i,j+1}}(\lambda) &= a_{i,j,0} + a_{i,j,1}\lambda + a_{i,j,2}\lambda^2 + a_{i,j,3}\lambda^3 \\ Y_{P_{i,j},P_{i,j+1}}(\lambda) &= b_{i,j,0} + b_{i,j,1}\lambda + b_{i,j,2}\lambda^2 + b_{i,j,3}\lambda^3 \\ Z_{P_{i,j},P_{i,j+1}}(\lambda) &= c_{i,j,0} + c_{i,j,1}\lambda + c_{i,j,2}\lambda^2 + c_{i,j,3}\lambda^3, \end{aligned} \tag{5.3}$$

where $a_{i,j,m}, b_{i,j,m}, c_{i,j,m}$ are the coefficients for the spline function, and $\lambda \in [0,1]$ is a scalar that parameterizes the traversal of a 3D point along a curve segment with $P_{i,j}$ and $P_{i,j+1}$ as its end points.

According to the spline continuity constraints (0th, 1st, and 2nd order), and assuming the second derivatives at the end points are 0, the spline coefficients $a_{i,j,1}$ can be obtained from the following system of linear equations:

$$\begin{bmatrix} 2\ 1 & & \\ 1\ 4\ \ 1 & & \\ & \ddots & \\ & & 1\ 4\ 1 \\ & & 1\ 2 \end{bmatrix} \begin{bmatrix} a_{i,1,1} \\ a_{i,2,1} \\ \vdots \\ a_{i,k-1,1} \\ a_{i,k,1} \end{bmatrix} = 3 \begin{bmatrix} x_{i,2} - x_{i,1} \\ x_{i,3} - x_{i,1} \\ \vdots \\ x_{i,k} - x_{i,k-2} \\ x_{i,k} - x_{i,k-1}. \end{bmatrix} \tag{5.4}$$

The coefficients $b_{i,j,1}$ and $c_{i,j,1}$ are obtained by similar formulations. Accordingly, the surface tangent vector $\overrightarrow{t_h}$ at point $P_{i,j}$ on the ith horizontal curve C_i^h is defined by

$$\begin{aligned} & \overrightarrow{t_h}(P_{i,j}) \\ = \ & [\frac{\partial X_{P_{i,j},P_{i,j+1}}}{\partial \lambda}, \frac{\partial Y_{P_{i,j},P_{i,j+1}}}{\partial \lambda}, \frac{\partial Z_{P_{i,j},P_{i,j+1}}}{\partial \lambda}] \mid_{\lambda=0} \\ = \ & [a_{i,j,1}, b_{i,j,1}, c_{i,j,1}] \\ = \ & f(x_{i,1}, y_{i,1}, z_{i,1}, ..., x_{i,k}, y_{i,k}, z_{i,k}) \end{aligned} \tag{5.5}$$

where f is a linear function and $x_{i,j}, y_{i,j}, z_{i,j}$ are the coordinates of the control points $P_{i,1}, ..., P_{i,k}$. For perspective images with a known intrinsic calibration matrix K, the 3D position of the pixel is determined by the depth $z_{i,j}$. Hence, $\overrightarrow{t_h}(P_{i,j})$ can be simplified to a function with k unknowns $z_{i,1}, ..., z_{i,k}$.

Let $\overrightarrow{n}(P_{i,j})$ be the normal measured at point $P_{i,j}$, there are two linear functions for depths $z_{i,1}, ..., z_{i,k}$ and $z_{1,j}, ..., z_{k,j}$:

$$\begin{aligned} \overrightarrow{n}(P_{i,j}) \cdot \overrightarrow{t_h}(P_{i,j}) &= \overrightarrow{n}(P_{i,j}) \cdot f_h(z_{i,1}, ..., z_{i,k}) = 0 \\ \overrightarrow{n}(P_{i,j}) \cdot \overrightarrow{t_v}(P_{i,j}) &= \overrightarrow{n}(P_{i,j}) \cdot f_v(z_{1,j}, ..., z_{k,j}) = 0 \end{aligned} \tag{5.6}$$

Equation 5.6 can be extended to a linear function for all the depths in the surflet (k^2 unknowns: $z_{1,1}, ..., z_{k,k}$). Given that each surface point is found at the intersection of two curves, there are two perpendicular tangents per point. So we can obtain $2k^2$ linear equations:

$$\underbrace{\begin{bmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,k^2} \\ m_{2,1} & m_{2,2} & \dots & m_{2,k^2} \\ & \vdots & & \\ m_{2k^2,1} & m_{2k^2,2} & \dots & m_{2k^2,k^2} \end{bmatrix}}_{M} \begin{bmatrix} z_{1,1} \\ z_{1,2} \\ \vdots \\ z_{k,k} \end{bmatrix} = 0 \tag{5.7}$$

where $m_{i,j}$ are constants for given intrinsic matrix and surflet size.

Let M denote the coefficient matrix in equation 5.7, and $M = USV^T$, where S is a diagonal matrix, and U,V are orthogonal matrices. The last column of V is a normalized nonzero solution for equation 5.7. Then, if we know the absolute depth for one point, we can recover the absolute depths for the whole surflet.

In our approach, the relationship between depth and normals is approximated by a linear function defined over a set of cubic spline coefficients. This modeling offers better accuracy and robustness than using an explicit linear relation among depth and normals. Moreover, by solving an over-determined system of equations in a least square manner, we are able to readily obtain a measure of the reliability of our estimate. In turn, unreliable relative depth estimates (i.e. with a poor fit) are identified as discontinuous surface patches. An examination of the minimum eigenvalue of the coefficient matrix in equation 5.7 is a straightforward criterion for the correctness of our estimates. However, a more geometrically meaningful criterion for discontinuity detection is to compute the difference between the estimated and observed normals. We define the normal difference by

$$D_{norm} = \sum_{1 \leq i,j \leq k} \frac{\| \overrightarrow{n}(P_{i,j}) - \overrightarrow{t_v}(P_{i,j}) \times \overrightarrow{t_h}(P_{i,j}) \|}{k^2}, \tag{5.8}$$

and use it as the quantitative criterion to evaluate the recovered depths. If the normal difference is greater than a pre-defined threshold, it means the surflet is not able to be approximated by our natural cubic spline model and it is highly probable that the current image patch contains a discontinuity.

Relative Surfaces through Surflet Aggregation

This section presents an efficient method to generate continuous 3D surfaces in the scene by a process of selectively aggregating and segregating surflets. The method introduced in Section 5.2 estimates the local geometry of a 3D surface patch by solving a linear system with k^2 unknowns. The quadratic growth of our linear system of equations renders the use of the surflet estimation method across an entire high resolution image as not computationally tractable. Moreover, any violations of the surface continuity assumption will cause severe distortions in our depth estimation.

In order to efficiently estimate an image-wide relative surface, we propose a region growing procedure that aggregates surflets to form disjoint surfaces whose relative depth is consistent with the input surface orientation measurements. Starting from one *seed* pixel, a local cubic spline patch is estimated and the surface is expanded by attaching neighboring surflets sequentially. The expansion of a given surface will stop at discontinuous regions (boundaries, edges or occlusions) and once a given surface can no longer be expanded in any direction, a new *seed* is selected. This process is repeated until all pixels in the image have been examined, yielding a segmentation of the scene into disjoint relative surfaces.

Suppose a is an arbitrary pixel in the image, and its corresponding $k \times k$ surflet is A. Notice that in this step, we only consider continuous patches, so if A is discontinuous, we discard a and randomly select a new pixel. We assume the depth of a is 1, and use it as the initial element in the fixed point set $F = \{a\}$. Suppose b is one of a's neighbors and its corresponding surflet is B. Let $d(x, Y)$ be a function that returns the estimated depth for a point x in surflet Y. By attaching patch B to A, the depth of point b will be scaled by s, where $s = \frac{d(b,A)}{d(b,B)}$. Then the attachment is validated according to the depth difference of a: $V_a = d(a, A) - s \cdot d(a, B)$. If V_a is small, which means the depth ratio of a to b estimated by two surflet are similar, then the attachment is valid; otherwise at least one approximated surflet is incorrect, so the attachment is invalid and point b is dropped. From these elements we define our depth scaling function as

$$s = \frac{1}{|F \cap P|} \sum_{x \in F \cap P} \frac{d(p, X)}{d(p, P)}, \tag{5.9}$$

where F is the current fixed point set, p is a new point to attach, and P is the surflet generated with p as the patch center. In addition, the function used to validate the correctness of the aggregation of a particular surflet to our existing surface estimate is defined by

$$V(s) = \frac{1}{|F \cap P|} \sum_{x \in F \cap P} (d(x, F) - s \cdot d(x, P))^2. \tag{5.10}$$

The attachment process repeats until all the points are added to the fixed point set or dropped at the end of this process, and then all the points in the fixed set form a continuous surface segment. In the same way, all the pixels in the image can be clustered to certain disconnected segments. The entire procedure is listed in Algorithm 2.

An example for a relative depth map and surface is shown in Figure 5.3. Compared to existing integral approaches, our method is more robust because the attaching operation usually involves more than one neighbors' information, and the validation operation guarantees that the attached

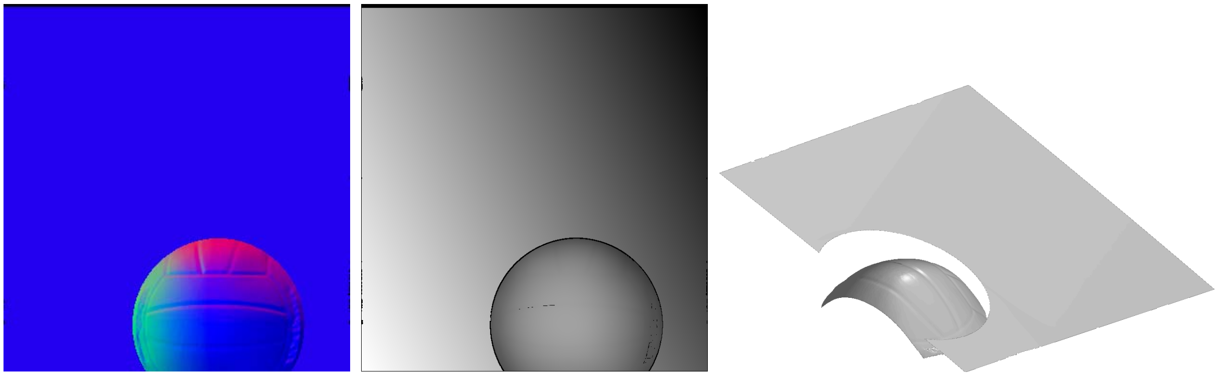


Figure 5.3: Relative depth map and surface generated from a single normal map.

surflet will coincide with existing surflets. Additionally, the validation operation will detect the discontinuity on the surface to prevent integrating disconnected segments. Although the generated related surface will vary for different expanding paths, the variation is bounded because of the validation operation.

Absolute Depth through Surflet Sweeping

After obtaining the connected surfaces, the next step is to obtain their absolute depth within the scene. An estimate of the absolute depth of a single pixel is sufficient to propagate depth information through an entire relative surface. In our approach, multiple 3D points are used to make the absolute depth estimate for a given surface more robust and mitigate any error propagation effects introduced in our surface generation method. Moreover, we estimate the absolute depth for a subset of points in the scene and adjust the relative surface to which they belong accordingly.

Feature points in the luminance image are identified using SIFT (Lowe, 1999). We generate the corresponding surflet to the SIFT feature and sweep the surflet along its viewing ray, as shown in Figure 5.4. By sweeping the surflet along the ray and projecting back to other images, we can accurately estimate the depth for the surflet. The surflet sweeping approach proposed here is similar to the plane sweeping (Collins, 1996; Yang and Pollefeys, 2003; Gallup et al., 2007), but points are no longer required to lie on a common plane. Moreover, by sweeping a parametric surface patch instead of a plane, we can model self-occlusions caused by local surface geometry. To reduce the

Algorithm 2 Relative Surface Generation

```
$U = \{p_1, ..., p_N\}$: initial set of all pixels to be attached
$D = \{0, ..., 0\}$: initial set of depth for all the pixels
$Segs = \{\}$: initial set of recovered surface segments
$si = 0$: index for the current segment
while $U \neq \{\}$ do
    $p = U\{1\}, U = U - \{p\}$
    generate surflet $P$ for $p$
    if $P$ is discontinuous then
        $continue$
    $si+ = 1$
    $S_{si} = \{p\}$: set of fixed pixels for current segment
    $D_p = 1$
    $N_{si}$: set of neighbor pixels for $S_{si}$
    $C_{si} = U \cap N_{si}$
    while $C_{si} \neq \{\}$ do
        $q = C_{si}\{1\}, C_{si} = C_{si} - \{q\}$
        generate surflet $Q$ for $q$
        if $Q$ is discontinuous then
            $U = U - \{q\}$
            $continue$
        computer depth $d$ for $q$ by function (5.9)
        validate $d$ by function (5.10)
        if $d$ is valid then
            $S_{si} = S_{si} \cup \{q\}, D_q = d, U = U - \{q\}$
            update $N_{si}$ and $C_{si}$
    $Segs\{si\} = S_{si}$
return $D$ and $Segs$
```

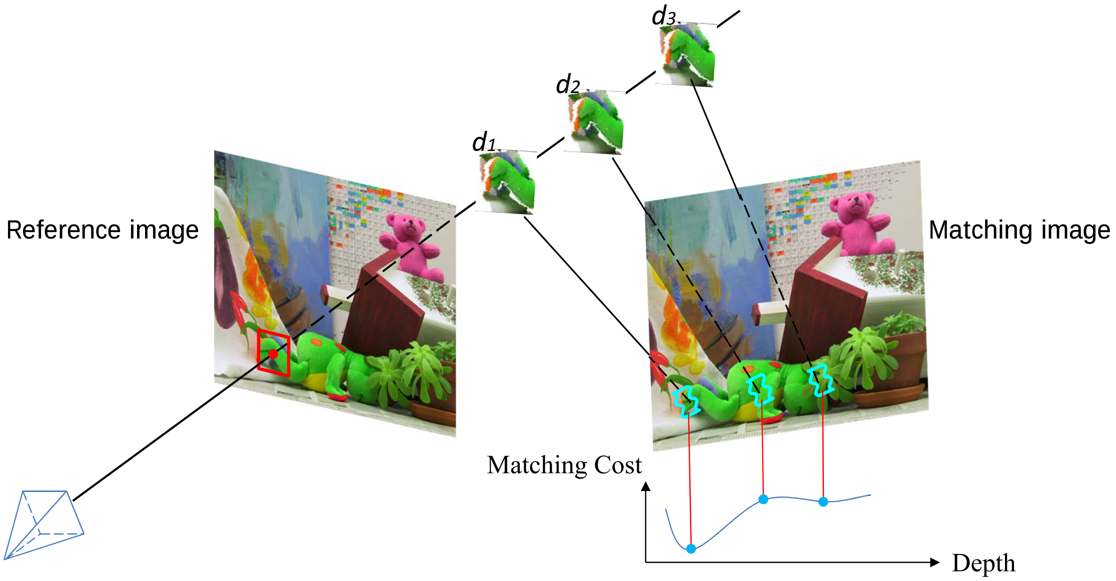


Figure 5.4: Surflet sweeping.

search space, we assume the surflet center can be seen in all the images, then the search space is limited by the boundaries of all the images. The projection of the reduced search space on each image is a segment along the epipolar line, and each pixel on the line segment delivers a depth interval.

The absolute depths obtained by surflet sweeping are then used to refine the surface segments. We foresee that surflet sweep stereo may provide erroneous estimates do to the limitations common to local stereo estimation approaches (e.g. foreground over-extension, multi-modal photo-consistency function). We deploy a RANSAC based data filtering approach to discard outlier depth estimates that would corrupt the absolute depth of our continuous relative surfaces. In fact, we use the pre-computed relative surface estimate as our fitting model. In this way, we obtain a filtered set of n feature points $p_1, ..., p_n$ with known depths $d_1, ..., d_n$ in the same segment. For each point x with relative depth d_r, its absolute depth is computed by equation 5.11.

$$d(x) = \frac{1}{\sum_{j=1}^{n} \frac{1}{\| x - p_j \|}} \sum_{i=1}^{n} \frac{d_i \cdot d_r}{\| x - p_i \|} \tag{5.11}$$

Experiments

We evaluated our approach on both synthetic and real data containing disconnected objects. The synthetic data contains the Stanford Bunny and the Happy Buddha models [2], whose surfaces are formed by about 1 million triangles. Each endpoint of the triangles is assigned a random value $[0, 1]$, so the color of the triangle is the mean of its three endpoints' colors, and the normal is computed by the cross product of two edges. We set 36 viewpoints surrounding the models, with a fixed intrinsic matrix and known extrinsic matrix (rotation and translation of the camera). At each viewpoint we generate 1 perspective grayscale image and 1 normal map by projecting the models to the image plane. In this experiment we choose six typical views, three front and three back images (Figure 5.5). The experiments on real data are captured by the spatial phase camera provided by Photon-X, and consist of three grayscale images (left, middle, right) and one normal map for the middle image (Figure 5.6).

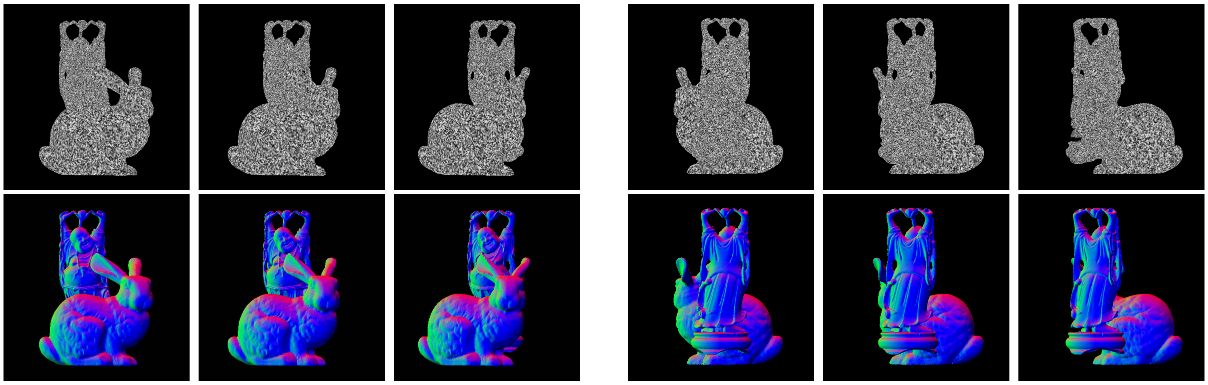


Figure 5.5: Grayscale and normal images for synthetic data.

We initially evaluate the generated depth maps using synthetic data. The model size is $215mm \times 115mm \times 205mm$, and the program runs several times with different patch sizes: 3×3, 5×5, 7×7, and 9×9. The depth maps and errors for synthetic data are shown in Figure 5.7 and Table 1, where we observe that greater overall reconstruction accuracy can be achieved by using the two smallest patch sizes. This is mainly attributed to improved local surface approximations near discontinuity boundaries. We also note that small patch sizes are computationally less demanding

[2] Stanford 3D Scanning Repository: http://www.graphics.stanford.edu/data/3Dscanrep/

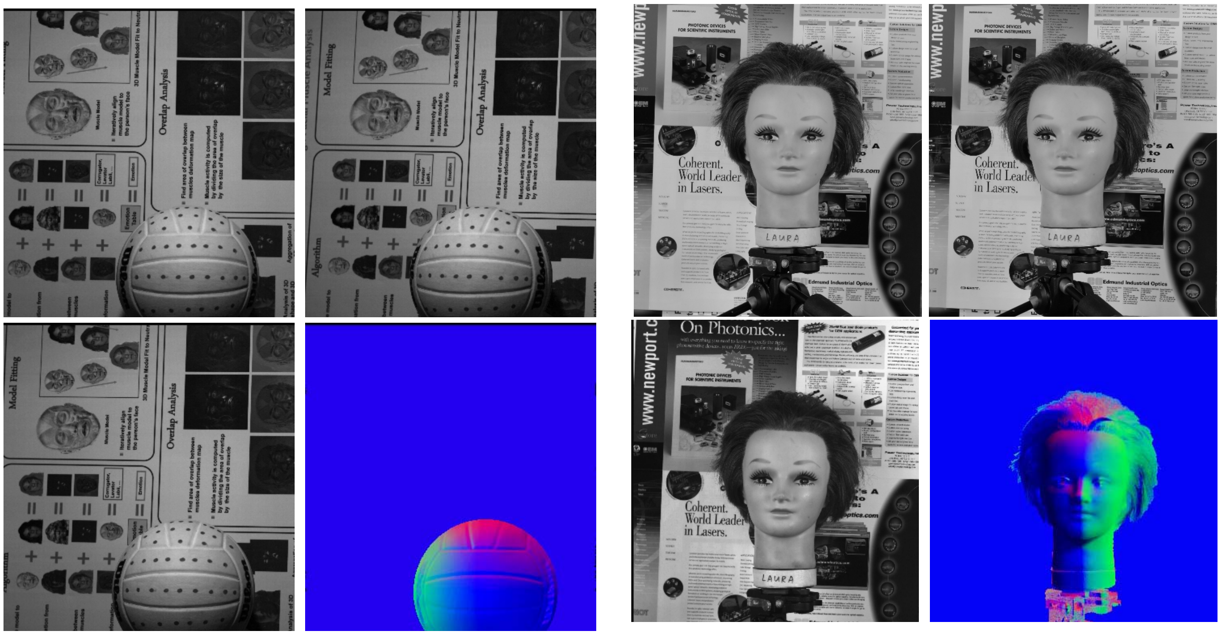


Figure 5.6: Grayscale and normal images for real data.

as the number of equations is proportional to the number of pixel in the patch. Accordingly, in following experiments, the default patch size is 3×3.

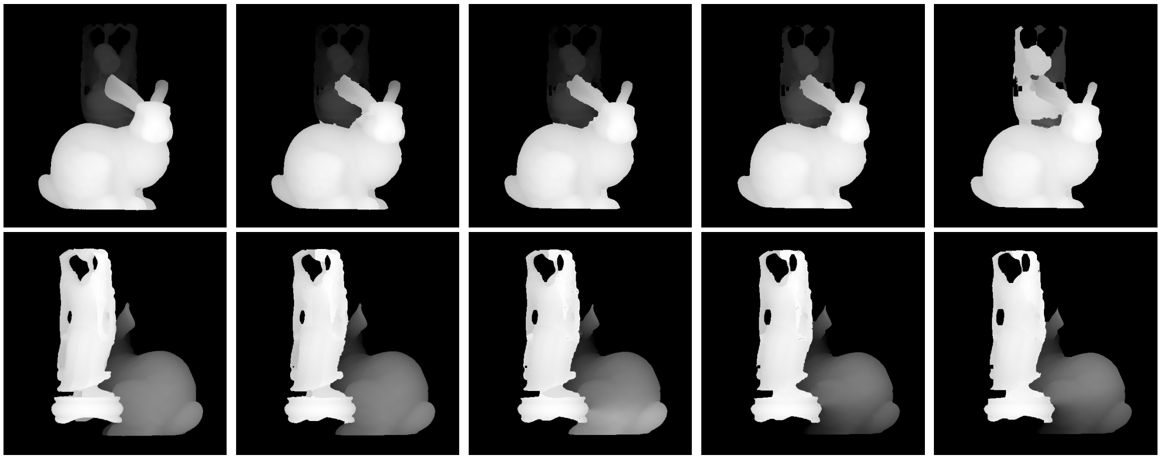


Figure 5.7: Depth maps for synthetic data. From left: Ground truth, patch size s=3,5,7, and 9.

Next, we investigate the capability of recovering discontinuous regions by comparing our method with shapelets approach (Kovesi, 2005) and the diffusion method proposed in (Agrawal et al., 2006). Note that the shapelets and the diffusion algorithms treat normal maps as orthogonal images, so the reconstructed surfaces differ slightly. The first row of Figure 5.8 shows the reconstructed

Patch	depth error (front)		depth error (back)	
size	mean	median	mean	median
3	2.5mm	0.38mm	2.0mm	0.4mm
5	2.5mm	0.30mm	6.4mm	1.1mm
7	2.0mm	0.39mm	7.9mm	2.0mm
9	20.9mm	0.47mm	9.7mm	1.5mm

Table 5.1: Mean and median depth error for synthetic data with different patch size.

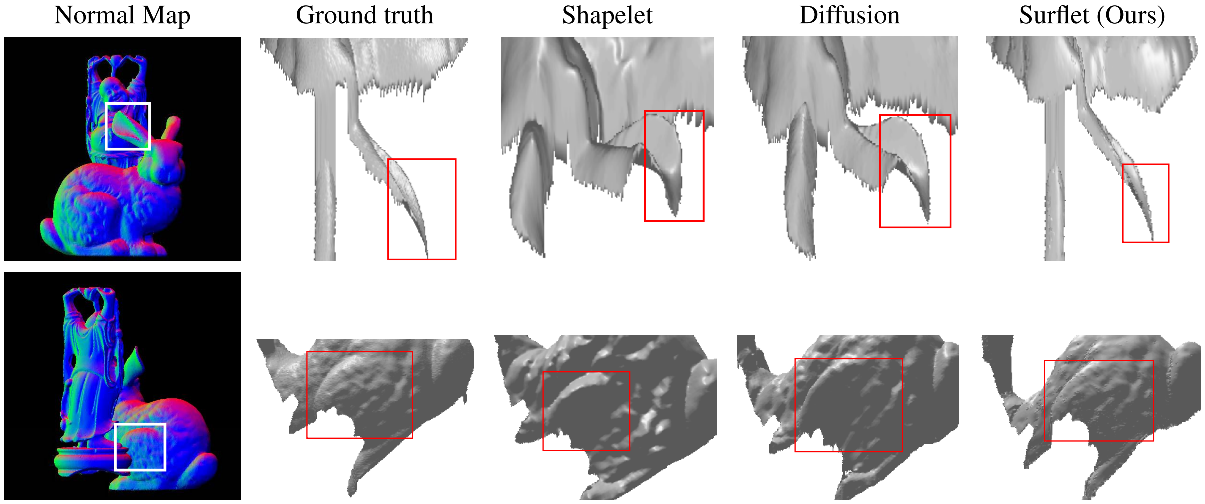


Figure 5.8: Comparison for discontinuous region reconstruction. The first column shows investigated overlapping regions (bounded by boxes), followed by surfaces reconstructed by Ground truth, Shapelets (Kovesi, 2005), Diffusion(Agrawal et al., 2006), and our approach.

ear of the bunny in the first (front) synthetic image. Since the ear is overlapped with Buddha's body, we can see the ear reconstructed by either shapelets or diffusion method has a sharp distortion near the tip. Also in the second row of Figure 5.8, where the Bunny's body is overlapped with Buddha's pedestal, we find that surfaces obtained by shapelets and diffusion methods become concave around the overlapping region. Moreover, only the surfaces reconstructed by our method maintain a consistent geometry w.r.t. the ground truth.

Figure 5.9 shows the estimated surfaces with their corresponding (estimated) normal maps. We can see that the front and back objects are clearly separated, and neither is distorted by the other, especially for the regions near the boundaries. However, for the real data there are still a number of jagged points along the shape boundary. Since we only use three images for photo-consistency based stereo matching, we expect these issues to be further reduced by considering additional images. The existence of small gaps in our face model (i.e. in the hair region) is due to the detection of disconnected surface regions (obtained through the analysis of normal information) where no SIFT features are found. While these artifacts can be improved by adjusting the discontinuity threshold, it is evident that the development of an improved adaptive scene segmentation mechanism is an important aspect to be addressed in future works.

In addition to evaluating the overall performance of our approach, we also tested the behavior of our multi-view surflet sweeping module. Namely, we compared it against a fronto-parallel plane sweep (Yang and Pollefeys, 2003) and an oriented plane sweep (Gallup et al., 2007). Our results (Figure 5.10) illustrate how modeling the local surface geometry consistently provides a better depth estimate based on photo-consistency. As expected, patches with a set normal surface measurements which diverge from the fronto parallel assumption are increasingly better estimated by our approach. Also, as the size of the matching template increases, surflets sweeping does a better job of modeling arbitrary surfaces.

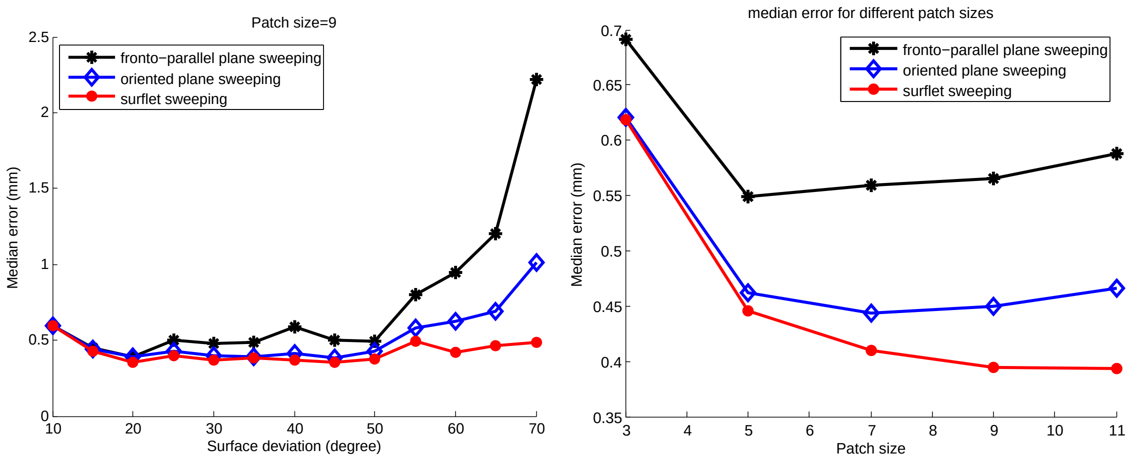


Figure 5.10: Surflet sweep accuracy comparison. At left, depth error as a function of the surface deviation from the fronto-parallel assumption. At right, depth error as a function of varying patch size.

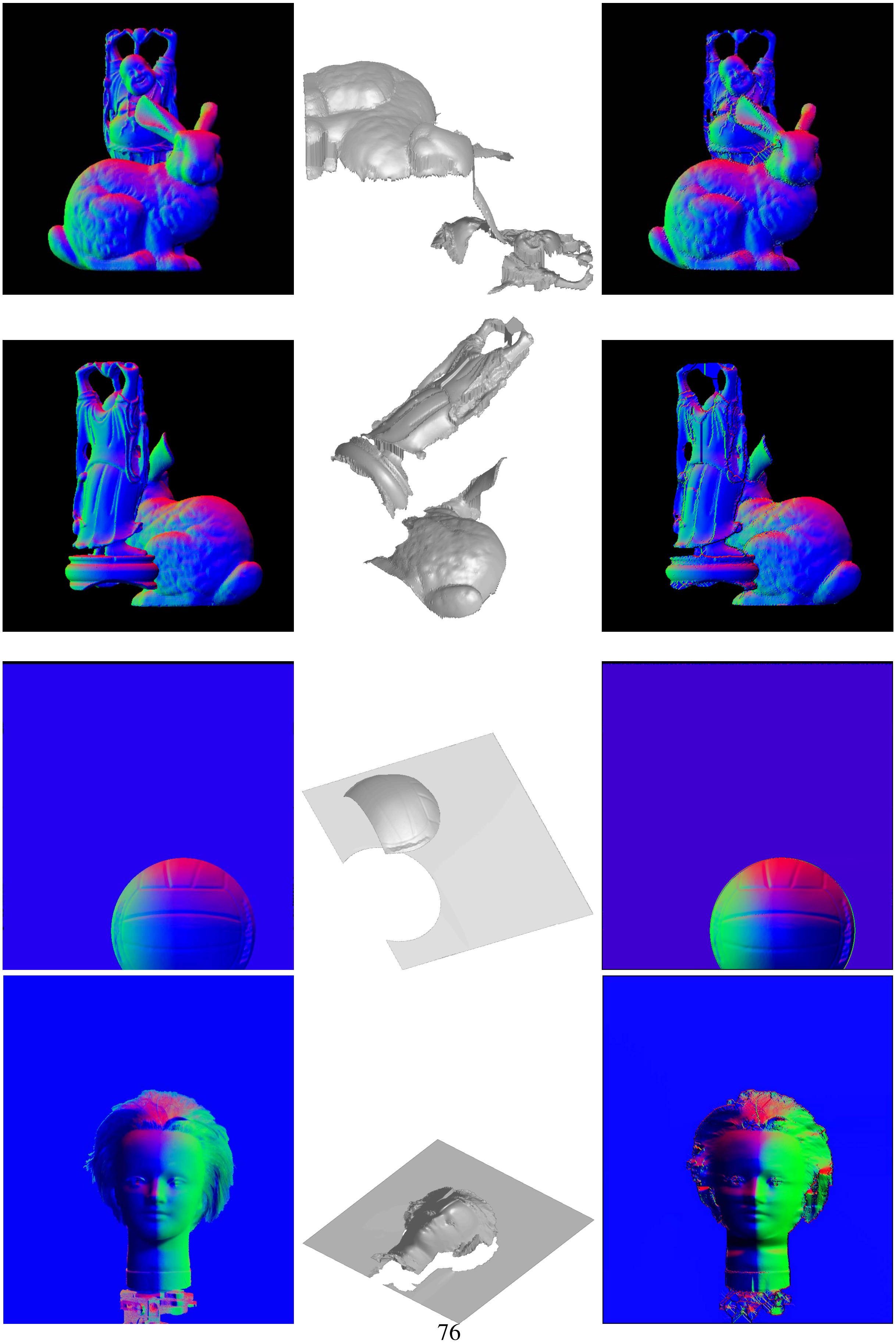


Figure 5.9: Recovered surfaces and normal maps. Left column: original normal maps used as input; middle column: recovered surfaces; right column: normal maps estimated from recovered depth maps.

Discussion

Besides the color information, we introduced approaches to obtain high quality surface reconstruction by normals. In contrast to previous surface from normal methods, our method can efficiently detect surface discontinuities, enabling its application to scenes containing multiple objects, which is then suitable to integrate with color-based stereo approaches. Accurate depth estimates are obtained through multi-view surflet sweeping, due to the precise local geometry estimation from surface normals. The proposed stereo approach outperforms existing plane sweep methods by being able to better model local surface geometry and incorporating this information into our photo-consistency estimates.

CHAPTER 6: STEREO BY STRUCTURE PROPAGATION

Introduction

In Chapter 5 we obtained high quality local structures with surface normal information captured by special hardware, which is hard to be satisfied in many applications. In this chapter, we will discuss how to approximate local structures when we only have color images.

By separating the entire image into proper small local patches, the complexity of the scene structure is also simplified due to the relative smoothness of each local patch. In Chapter 4, we utilize this property to reduce the search space by sequential optimal sampling. Actually, besides the candidate depths/disparities, the sampled pixels also provide the spatial cues of the local geometry, which promotes a structure-oriented stereo strategy that outperforms state of the art sampling based stereo techniques in accuracy and speed.

One difficulty of the matching cost aggregation in stereo is that in theory there are infinite kinds of depth distributions for pixels within the aggregation window, which has to be approximated by some simple structures such that the depth of each neighboring pixel can be determined based on few factors (e.g. the depth of the center pixel). The most popular structure is planes, and a general framework is called Plane Sweeping (Collins, 1996). Plane Sweeping (PS) is a depth-map generation framework for general multi-view configurations (Seitz et al., 2006). PS evaluates for each pixel a sequence of depth hypotheses through the comparison of the reference and matching images, using a 3D plane-induced homographic transformation between both images. Hence, the depth estimation process relies on an explicit and pre-determined sampling or discretization of the space of feasible homography inducing planes $\pi \in \mathbb{P}^3$. The parameterization of π can be decomposed into a depth value $d \in \mathbb{R}_+$ defined along the reference camera's optical axis, and a plane orientation vector $\mathbf{n} = (n_x, n_y, n_z) \in \mathbb{R}^3$, where $\|\mathbf{n}\| = 1$. Single orientation PS (typically fronto-parallel) defines for each pixel a 1D search, which depends exclusively on the hypotheses depth parameter d selected from a set of values $\mathbf{D} = \{d_1, \ldots, d_D\}$. Implicitly, a set of orientations $\mathbf{O} = \{\mathbf{n}\}$, consisting of a single element, is also being evaluated. In rectified stereo (Bleyer et al., 2011), the plane $\pi : d = a \cdot x + b \cdot y + c \cdot z$ for pixel $p(x_0, y_0)$ is determined by a normal vector $\mathbf{n}$ and the disparity z_0, where $a = -\frac{n_x}{n_z}$, $b = -\frac{n_y}{n_z}$, and $c = \frac{n_x x_0 + n_y y_0 + n_z z_0}{n_z}$. And we can enforce fronto-parallel plane $\pi : d = z_0$ by setting $\mathbf{n} = (0, 0, 1)$.

In following section, we will discuss how to estimate the structure of the local patch by planes, and propose an efficient propagation-based stereo algorithm. All the discussion is based on the rectified images, and can be straightforwardly extended into multi-view stereo.

Related work

Integrated multi-view stereo approaches for 3D model generation commonly strive to estimate surface geometry either by generalizing the results of sparse reconstructions or by fitting a global surface parametric model to the observed data. (Furukawa and Ponce, 2007; Bradley et al., 2008; Liu et al., 2009; Kolev et al., 2010). Furukawa and Ponce (2007) implemented multi-view stereopsis as a match, expand, and filter procedure. First, some (feature) points are recovered in 3D space, and then triangulated as sparse patches, which were then spread to nearby pixels to form a dense set of patches. The authors use visibility constraints to eliminate incorrect matches. Bradley et al. (2008) applied scaled window matching in binocular stereo, and integrate outlier removal in a lower dimensional meshing. Liu et al. (2009) integrated silhouette information and epipolar constraint into the variational method for continuous depth map estimation, and then synthesized depth maps according to path-based NCC (normalized cross correlation) metric. Kolev et al. (2010) generalized the photo-consistency-weighted minimal surface approach by means of an anisotropic metric to integrate a specified surface orientation into the optimization process.

The plane sweeping approach is a category of local stereo methods that performs matching across multiple unrectified images. Collins (1996) first introduced this method to reconstruct sparse features by back-projecting them onto successive fronto-parallel planes. The algorithm was then extended for dense reconstruction in (Yang et al., 2002; Nozick et al., 2005), where each pixel has a cost volume in the discretized depth space, and the final depth is the one with the minimum cost. Due to perspective distortion, such fronto-parallel plane sweeping methods are not proper for reconstructing oblique surfaces (Zabulis and Kordelas, 2006). Gallup et al. (2007) used slanted planes for facade reconstruction, where the sweeping directions are estimated by the sparse points cloud obtained from structure-from-motion. Carceroni and Kutulakos (2002) used fronto-parallel plane sweeping to estimate depths first and then used surfel (surface element) to estimate orientation. A common problem in existing sweeping approaches is that depth planes are sampled by uniformly discretizing a predefined depth range. Zabulis and Daniilidis (2004); Zabulis and Kordelas (2006) estimated the surface normals by oriented plane sweeping. They back-projected images onto a planar patch in 3D space and computed the correlation. For each orientation of the patch a correlation score is computed and the orientation that yields the highest correlation was selected as the best one.

PatchMatch (Bleyer et al., 2011) first introduces the slanted planar surface in rectified stereo. In PatchMatch, each pixel is initialized with a random plane and then update the plane by three kinds of propagation: Spatial Propagation, View Propagation, and Temporal Propagation. However, the number of sampled planes (equals the number of all pixels on the image) is still very small with respect to the space of candidate planes, a iterative plane refinement procedure is required to converge to the correct plane. In contrast, our method proposed in this chapter significantly reduces the search space of candidate planes, and does not need any iterative plane refinement during the propagation.

Local Structure Estimation

Compared with the entire scene, the structure of each local patch is relatively simple. However, like finding the candidate search space discussed in Chapter 4, we need to consider how to guarantee the completeness and sufficiency of the exploited structures.

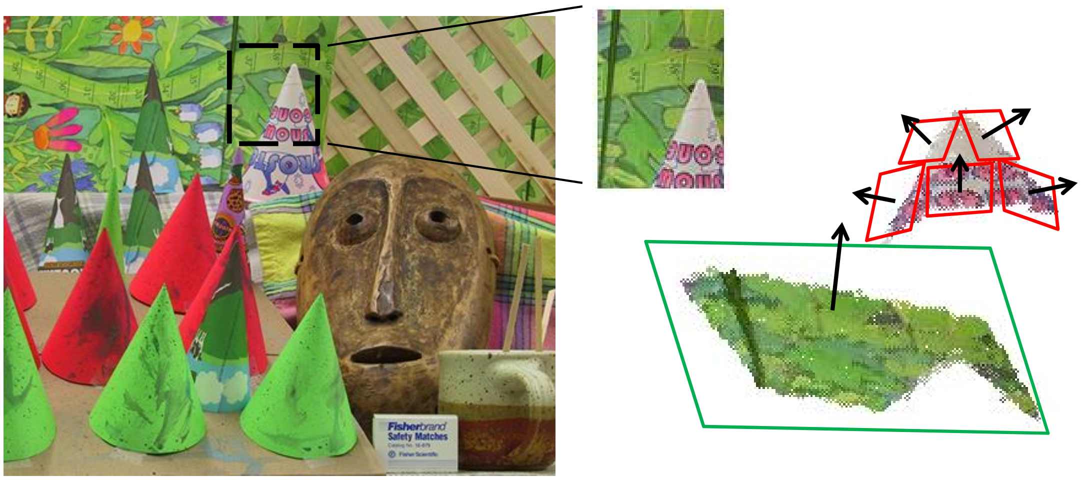


Figure 6.1: Structure pattern for the local patch.

Structure Pattern for Local Patches

The local structure can be approximated by a group of slanted planes (as illustrated in Figure 6.1). Besides the orientation and depth/disparisy of the plane, the positions of those planes should also be known, and the combination of planes and their distribution is called the *Structure Pattern* of the local patch.

An exhaustive approach is to find the slanted plane that best represents the local geometry of each pixel, which is obviously intractable because of infinite candidate planes. A reasonable way is to divide the local patch into several disjoint sub-patches, where each sub-patch can be approximated by one slanted surface. In this ideal case, we just need to find a proper plane for one pixel from each sub-patch to achieve a complete and sufficient structure pattern of the local patch. An intuitive way to find this ideal structure pattern is to segment the local patch, and then approximate each segment by a slanted plane. However, since there is no explicit or implicit correlation between the depth and color, just performing color-based segmentation cannot guarantee the completeness and sufficiency of the discovered structure pattern. For example, in Figure 6.1 the background poster contains many different paintings, each of which will be treated as a distinct segment and then approximated by one slanted plane. However, all of those regions have the same structure (one fronto-parallel plane), so this over-segmentation will lead to redundant computational costs. Due to the homogenous texture, the front cone in Figure 6.1 will be treated as a single segment, but in fact it has a curved surface and should be approximated by a batch of planes with gradually changed orientations. Therefore, color-based segmentation will lead to incomplete structure pattern, and we need a better sampling method that considers both color and depth information of the local scene.

Structure Pattern Sampling

Here we estimate the structure pattern by pixel-wise sampling. Instead of computing aggregated matching costs for all possible oriented planes per pixel, we find the best structure by fitting planes among multiple pixels, which is much more efficient and robust.

For a given local patch, we randomly select some sampled pixels (treated as seeds), and compute their pixels-wise matching costs, where each seed will be assigned with the disparity that has the minimum matching cost in the candidate disparity set. All seeds are connected with each other and the edge between any two seeds will be assigned with a weight according to their color similarity and spatial distances using Equation 2.2. Then all the edges whose weights are smaller than certain per-defined threshold T_c are removed, and those seeds are grouped into several connected components. Each sampled pixel s will be fitted with oriented planes by RANSAC (Fischler and Bolles, 1981; Raguram et al., 2013): in each iteration we randomly select three sampled pixels from the same component, and generate a plane $\pi(x, y)$: $d = a \cdot x + b \cdot y + c$, where x and y are coordinates of the seed, d is the corresponding best disparity, and a, b, and c are coefficients to be determined. Aggregating the weighted matching cost for pixel s from all the seeds belonging to the same component with respect to the generated plane, and keep the plane if its cost is better than the current cost (the default cost is computed from the fronto-parallel plane). Once a new plane is found, the local pixel will also check the neighboring planes π^+ and π^-, which are generated by increasing and decreasing c with 1 disparity, and keep the one with the best matching cost. Repeated sampling, until no better matching cost has been found for consecutive $R(= 10)$ randomly generated planes. In this way, we can find the best planes for all sampled pixels, which will be used as seeds for structure propagation.

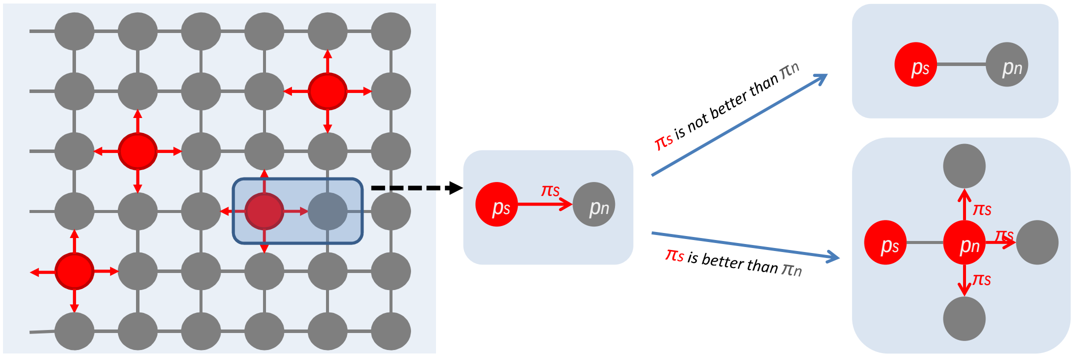

p_s
π_s
p_n
π_s is not better than π_n
p_s
p_n
π_s is better than π_n
p_s
p_n
π_s
π_s
π_s

Figure 6.2: Plane Propagation.

One issue of the local structure exploration is the completeness. As discussed in in Chapter 4, we can set a large random sampling threshold to guarantee over-sampling even in the most complicated local patch. A better way is selecting seeds by the sequential optimal sampling proposed in Chapter 4. Our optimal sampling scheme adaptively separates the image into proper sub-regions, whose local structures are also inherently represented by the candidate disparity set and the spatial distribution of the corresponding sampled pixels. Another benefit is that the number of sampled pixels are adaptive to the local geometry, so that it avoids generating insufficient similar planes for simple smooth regions.

Local Structure Propagation

In this section, we will discuss how to integrate explored local structure patterns to obtain a dense depth map. Ideally, the combination of all local structure patterns could be treated as a sparse skeleton of the entire scene. According to the local smoothness assumption, neighboring pixels should have similar structures as the sampled seeds, which means each pixel can directly choose the slanted plane from its most "similar" seed.

Propagation from Structure Pattern

Here we propose a propagation-based strategy to pass potential structure information across the image. Initially, each seed p_s propagates its plane $\pi_s(x, y)$ to its four neighbors (red pixels in Figure 6.2), and each neighbor p_n will perform one of the following actions when receiving π_s:

- If $d_n = \pi_s(x_n, y_n)$ is not in p_n's search space D_n, discard π_s;
- Otherwise:
  - If the neighbor p_n has not received any disparity before, set p_n's plane $\pi_n = \pi_s$, and propagate π_s to p_n's neighbors;
  - If p_n's plane π_n equals π_s, do nothing;
  - If $\pi_n \neq \pi_s$, compare their corresponding matching costs (for the first appeared plane, compute the matching cost and store in the local cost table), if π_s's cost is better, update local plane by the best of $\{\pi_s^-, \pi_s, \pi_s^+\}$, and propagate the new plane to p_n's neighbors;

In each iteration, pixels that update their local plane will be the seeds in the next iteration, and the propagation will stop when no updates are occurring. Since the SOS (Chapter 4) is able to automatically split the local patch into suitable portions along with appropriate number of samples, this seed propagation scheme corresponds to the implicit smoothness of the local patches. Hence, many pixels can directly “borrow” the plane (and corresponding disparity) from seeds without computing the matching costs. Matching cost estimation will only need to be performed for pixels that already have an assigned local plane (either by seed initialization or subsequent propagation) and are receiving a contradicting disparity estimate from one of their neighbors. Even if there are some incorrect seeds involved in the propagation, their influence is restricted within the regions whose reduced search space contains such erroneous disparities, which is more robust and leads to less computational effort spent on wrong planes.

Parallel Propagation Scheme

For high resolution images, a more efficient solution is parallel processing, and here we give a parallelized version of our structure propagation stereo. First, the image is divided into K disjoint $M \times M$ sampling blocks, and each block runs sequential optimal sampling independently by one thread. After all blocks have been sampled, we redivide the image into multiple $M' \times M'$ propagation blocks ($M' = 2M \ or \ 3M$), and perform the propagation algorithm independently (as illustrated in Figure 6.3). The average accuracy of SOS is about 95%, which means there may be few fine structures missing from the recovered structure pattern, and most of these missing structures come from isolated regions along the sampling block boundaries. Thus by propagating structures in a larger regions, the structure cues missing from one sampling block will be complemented from neighboring blocks, which improves the reliability and accuracy of the final result. After all propagation blocks converge to stable results, the pixels along the boundaries will pass their local best planes to the neighboring pixels that belongs to other blocks, and neighboring blocks will run the inner propagation again. Repeat this procedure until the entire image converge to a stable disparity map. In practice, we found that just performing the parallel block propagation is almost good enough to generate accurate results without any discontinuity along the block boundaries, which means the exploited structure patterns are sufficient.

Disparity Post-Processing

Next we perform a disparity refinement where the reliable pixel disparity estimates are identified through left-right cross-validation and unreliable pixels near the left image boundary are assigned the median of the neighboring reliable pixels on the same row. Remaining unreliable pixels are filled by an improved interpolation method proposed in (Hirschmuller, 2008). For each occluded pixel (a reference pixel whose disparity is smaller than the disparity of its corresponding pixel on the matching image (Hirschmuller, 2008)), we will find the closet reliable pixels based on 16 directions, and classify those reliable pixels as similar neighbors and dissimilar neighbors according to their

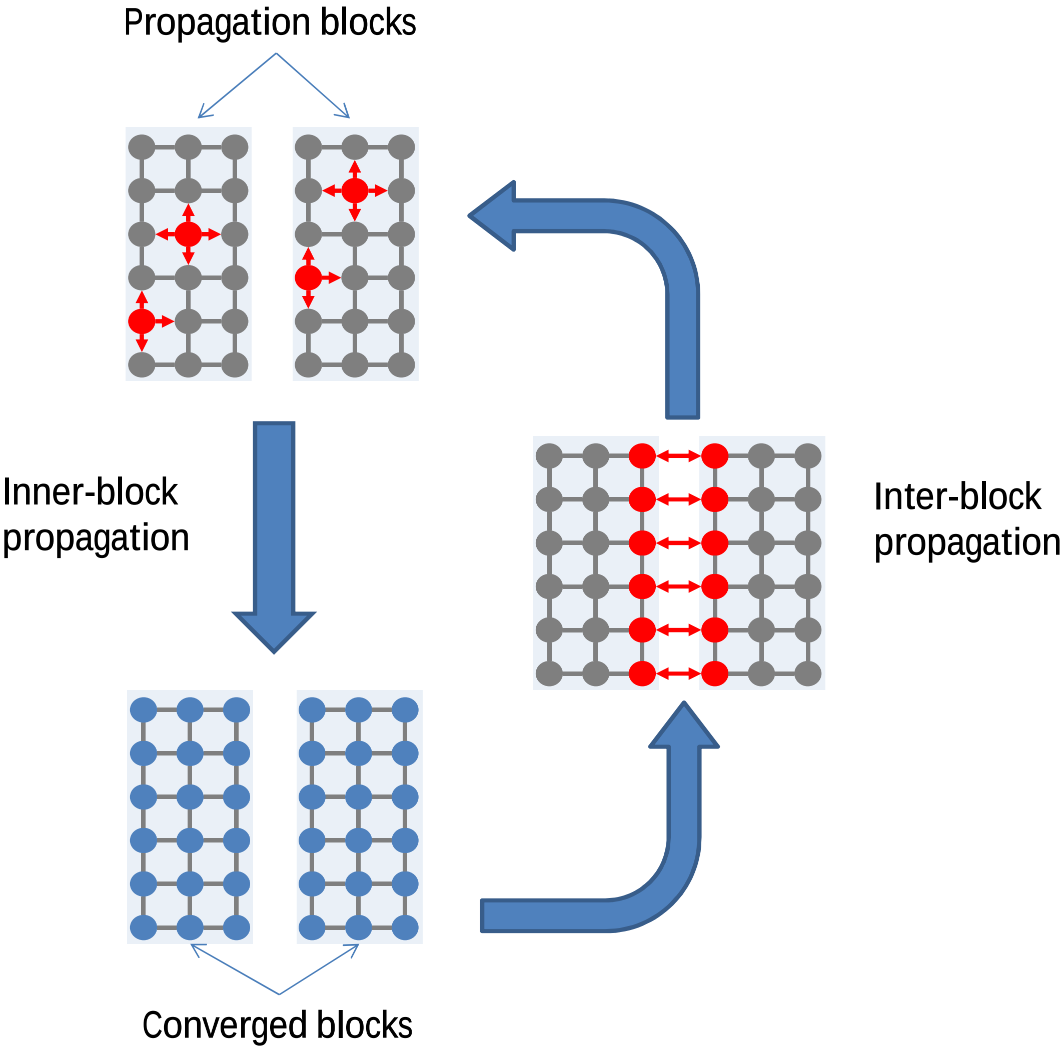


Figure 6.3: Parallel Propagation Scheme.

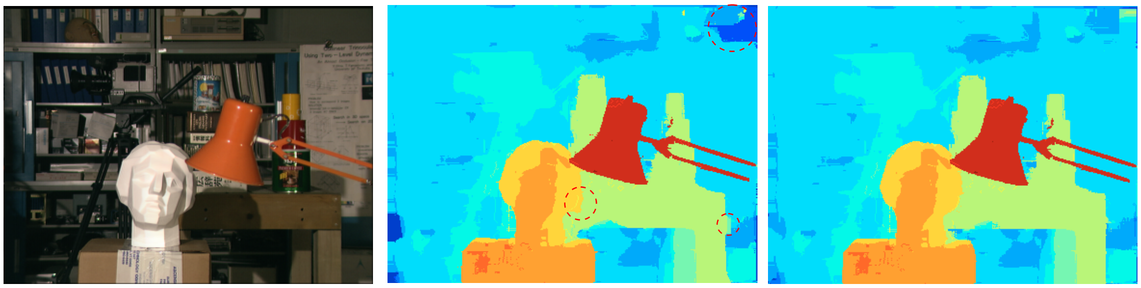


Figure 6.4: Homogenous component refinement. From left to right: color image, disparity maps before and after homogenous component refinement.

color similarity. If there are some similar neighbors found, set the local disparity as the minimum disparities of those similar neighbors; otherwise, set the local disparity as the minimum disparities of the dissimilar neighbors. For each mismatched pixel (the reference pixel has greater disparity), the local disparity is set as: i) the disparity of the most similar neighbor, or ii) the minimum disparity of all neighbors if there is no similar neighbor.

A process deemed homogenous component refinement, see figure 6.4. Our proposed refinement strives to correct remaining errors by propagating with the disparity that dominate the entire component. First, we separate the image into disjoint homogenous components by connecting the neighboring pixels if their color difference is below a pre-defined threshold. Homogenous components that contain too many difference disparities (≥ 10) will be discarded because there does not exist any dominated disparity. For other homogenous components, initially all pixels with the dominated disparity are selected as seeds, and each seed will check its four neighbors. If neighbor pixel's disparity is similar (± 1) as the local's, keep it unchanged. Otherwise, change the neighbor's disparity to the local disparity. And the new visited pixels become new seeds for the next iteration. Repeat this procedure until all the pixels in the component have been visited. The effect of the this refinement is shown in Figure 6.4 (right). Finally, the disparity map will be smoothed by weighted median filter first with a large window size (about 2% of the image width), and then with a small window size (=3).

Experiments

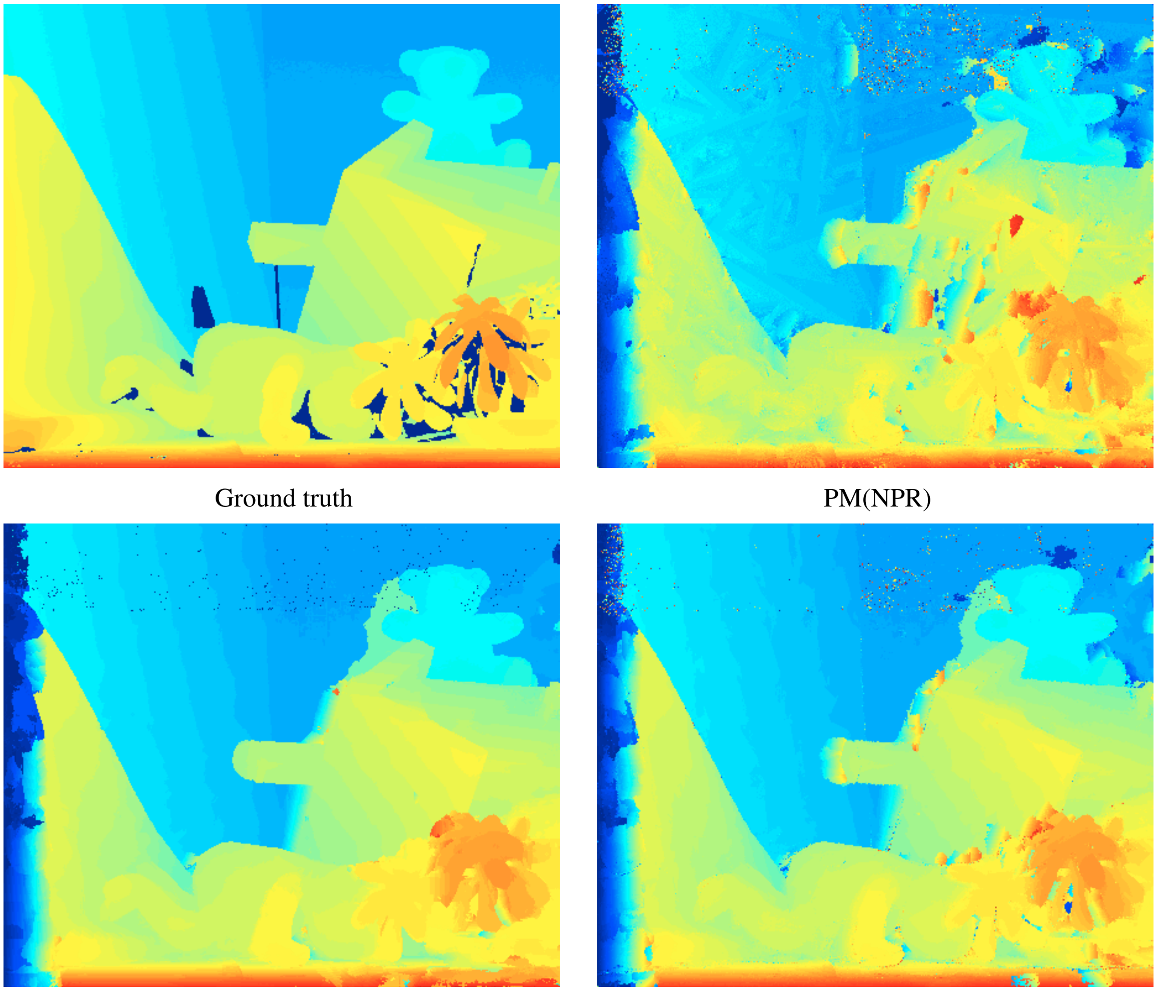
We evaluate our proposed plane propagation scheme (SOS$^+$) under the Middlebury Stereo datasets (Scharstein and Szeliski, 2002). The initial seeds for SOS$^+$ are sampled by our SOS scheme proposed in Chapter 4 with default parameters. Matching cost computation parameters are set to the default parameters proposed in (Bleyer et al., 2011). All algorithms were implemented in C++ and executed on an Intel Xeon CPU W3540 2.93GHz.

Stereo with Slanted Planes

In the first experiment, we investigate the effect of aggregating matching cost across oriented planes by comparing our SOS$^+$ algorithm against PatchMatch (PM). The aggregation window for both methods is 31×31 (similar as the default value in (Bleyer et al., 2011)). For PM, the maximum number of iterations is four, and in each iteration the planes are propagated starting from the top-left to the bottom-right, and then they are propagated back to the top-left. Two types of PM are tested: the first one, PM(NPR), propagates the randomly initialized planes, and the second one (PM) incorporates the iterative plane refinement (Bleyer et al., 2011). The raw disparity maps of image “Teddy” are shown in Figure 6.5, and the corresponding processing times are: SOS$^+$ 17.41s, PM(NPR) 220.54s, and PM 747.97s. Without the iterative plane refinement step, there are many ambiguous regions that can not be recovered by PM’s propagation scheme, but recovered by our SOS$^+$.

Stereo on Fronto-Parallel Plane

To enable leveled comparison against algorithms working under the fronto-parallel assumption, Histogram Aggregation (HA) (Min et al., 2011) and typical local Exhaustive search (EX), we modify SOS$^+$ and PM for compliance to this assumption. The default window size for cost aggregation is 11. For HA (position-dependent), the spatial ratio is 3, and aggregation window is 31 (the default value used in (Min et al., 2011), which is similar to aggregate cost from 11×11 pixels).



Ground truth

PM(NPR)

SOS$^+$

PM

Figure 6.5: Raw disparity maps comparison for SOS$^+$ and PM, and PM(NPR) is the PM without plane refinement.

Time (s)	Tsukuba	Venus	Teddy	Cones	Art	Books
SOS$^+$(FP)	**0.33**	**0.41**	**0.89**	**0.88**	**1.13**	**0.94**
PM(FP)	1.91	2.80	3.28	3.29	3.72	3.5
HA	1.80	3.09	6.41	6.38	8.21	8.13
EX	5.13	9.48	24.61	23.05	31.22	32.34

Table 6.1: Processing time for various stereo methods.

Figure 6.6 shows samples of raw disparity maps generated by the various stereo algorithms, and the corresponding processing times are listed in Table 6.1. We observe that SOS$^+$(FP) is the fastest method for all test images. For example, the processing time of SOS$^+$(FP) for image "Books" is $0.94s$, which is even smaller than time spent on the random initialization for PM(FP) (computing the matching cost for a random disparity per pixel takes $1.28s$). Note that SOS$^+$(FP) evaluates on reduced search spaces corresponding to local structures (exploited by SOS), which will converge quickly and many pixels just receive the propagated disparity values without any matching cost computation. These results indicate SOS$^+$ is more efficient than sampling methods not exploiting local scene structure.

We also compare SOS$^+$(FP) and PM(FP) on the high resolution $(21M)$ images of (Kim et al., 2013) with a large candidate disparities set of 250 disparities. Figure 6.7 shows the raw disparity map and corresponding processing time for PM(FP) and SOS$^+$(FP), and we also can see SOS$^+$(FP) outperforms PM(FP) in both quality and speed. And figure 6.8 compares the raw disparity maps for SOS$^+$(FP), PM(FP), and PM(FP)+SOS, and although SOS$^+$(FP) and PM(FP)+SOS work on the same reduced search space, our propagation scheme is still much faster (30 minutes less) than PatchMatch. The corresponding 3D point cloud of the terrain estimated by SOS$^+$(FP) is given in Figure 6.9.

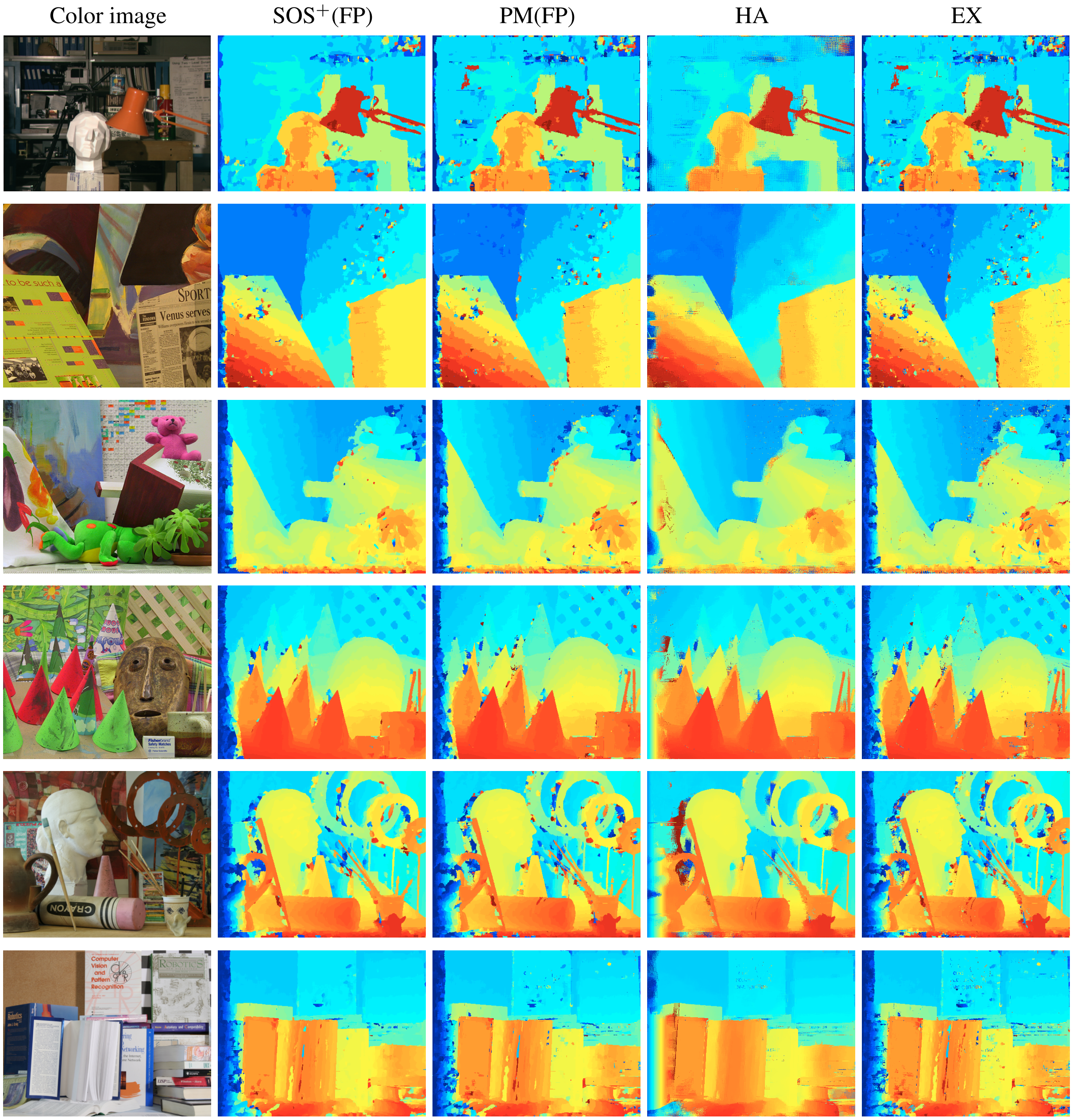


Figure 6.6: Raw disparity maps for various stereo methods.

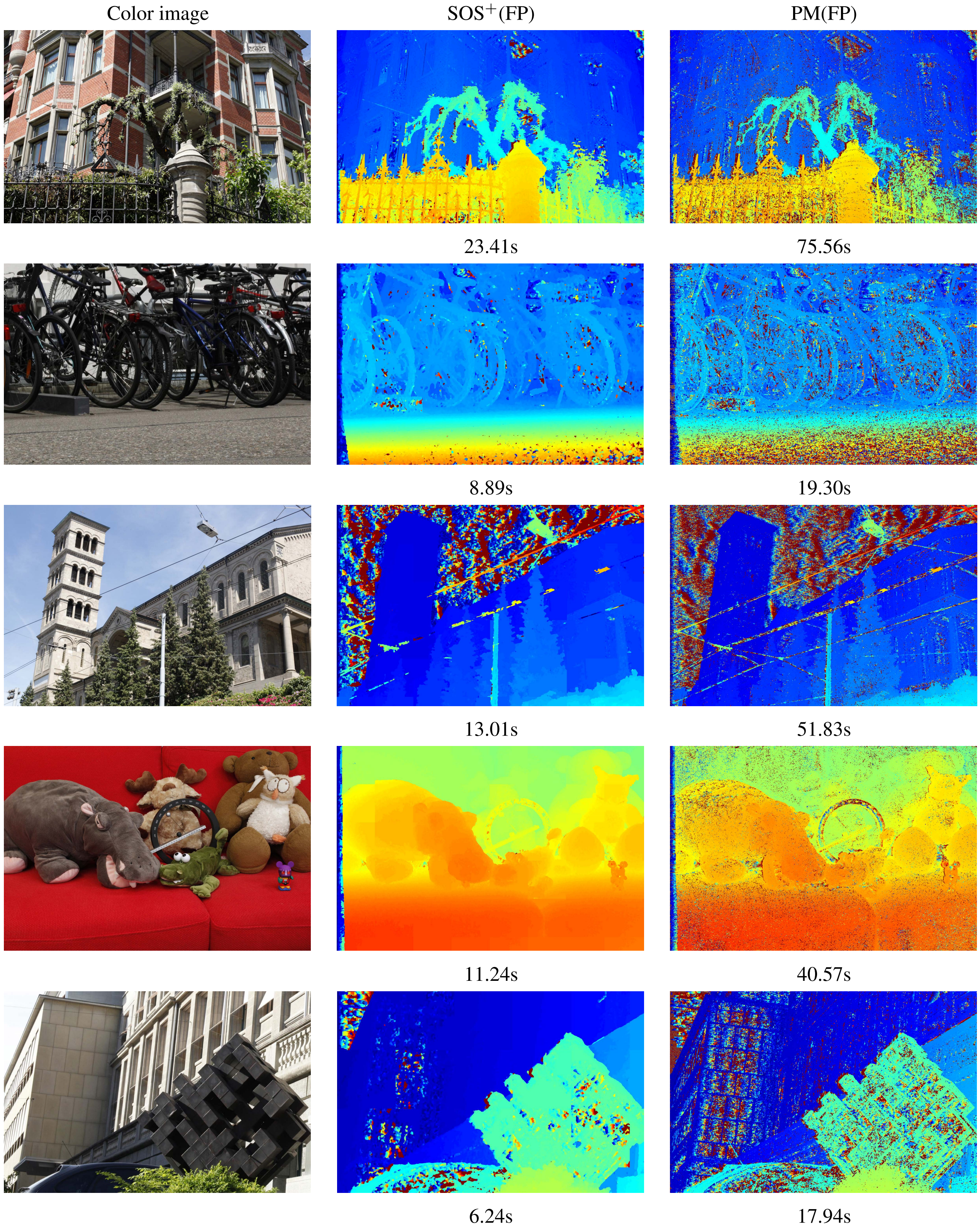
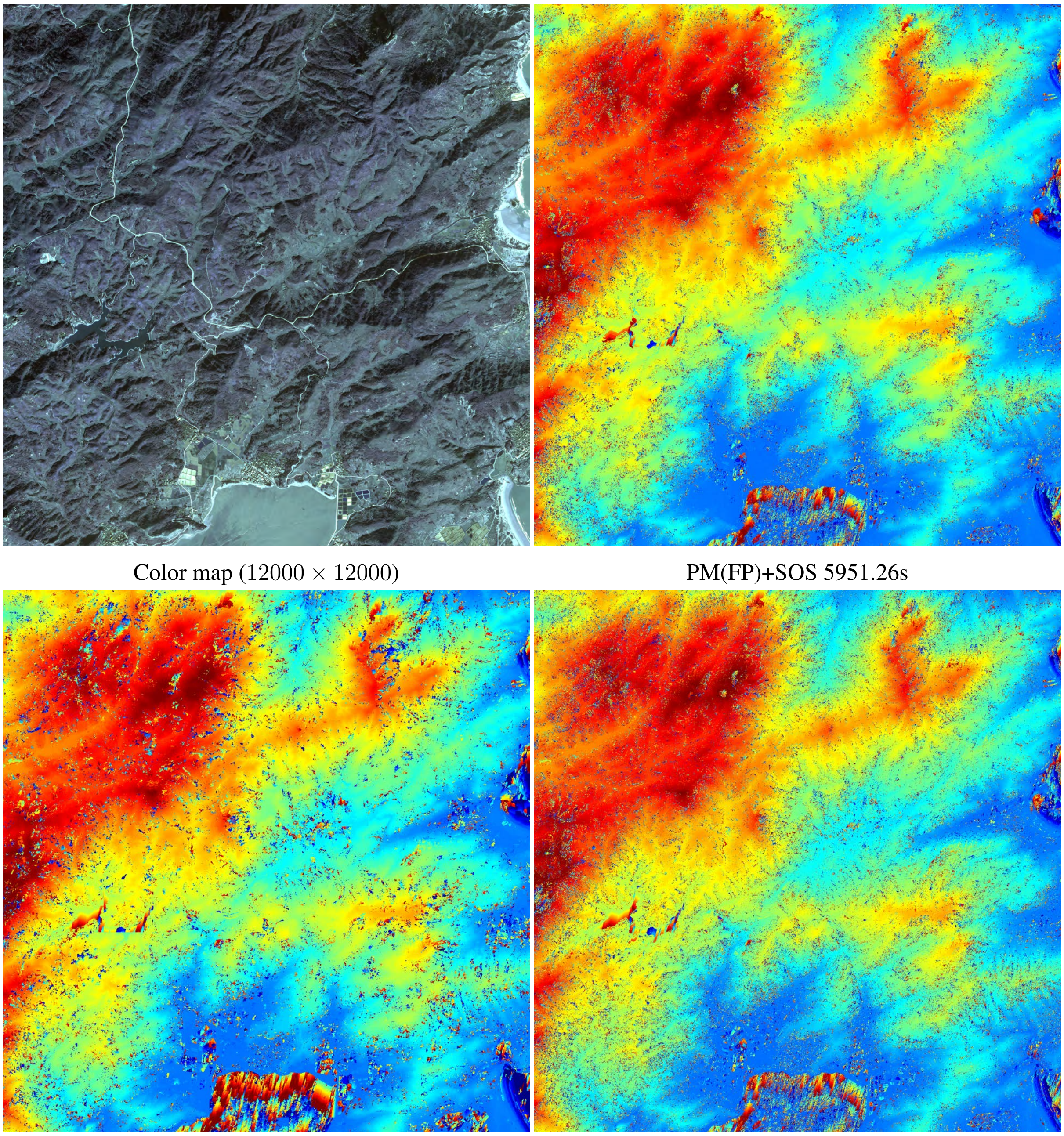


Figure 6.7: Raw disparity maps and processing time (10 threads) for high resolution ($21M$) images.



Color map (12000 × 12000)

PM(FP)+SOS 5951.26s

SOS^{+}(FP) 4073.94s

PM(FP) 6440.36s

Figure 6.8: Raw disparity maps and processing time (10 threads) for SOS^{+}(FP),PM(FP)+SOS, and PM(FP) for satellite image.

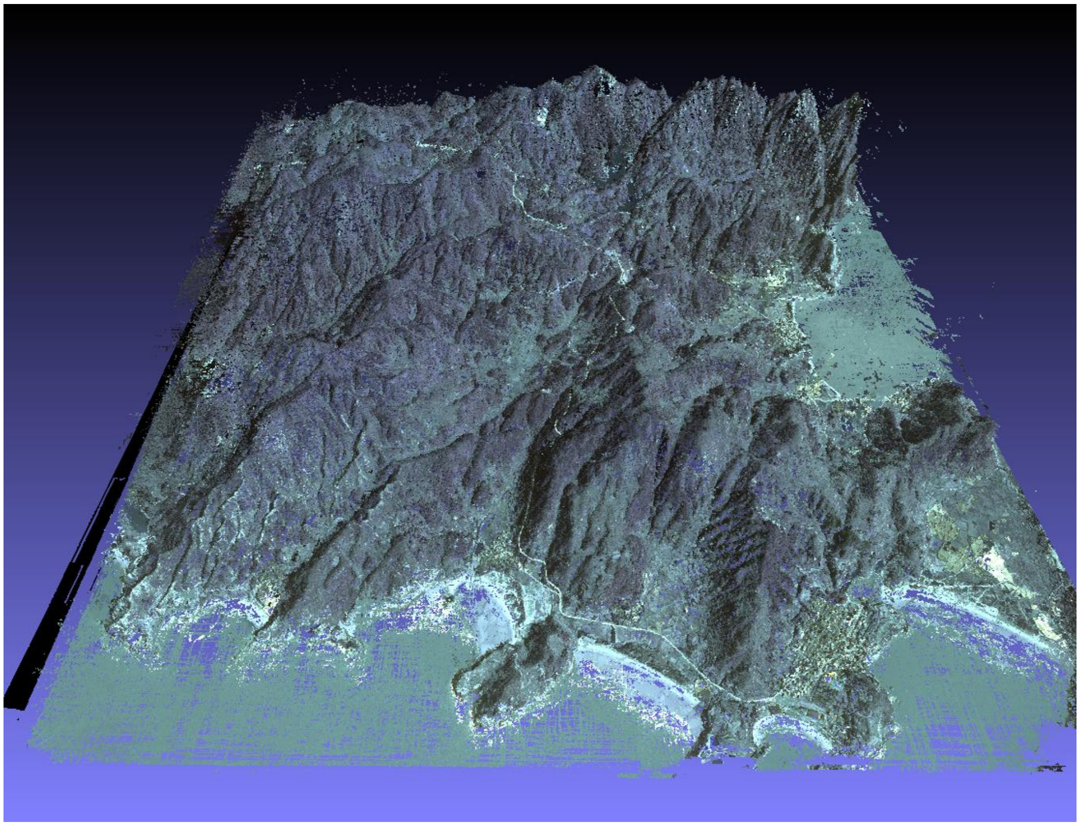


Figure 6.9: Corresponding point cloud for the depth map from two satellite images as estimated by SOS^{+}(FP).

	Tsukuba	Venus	Teddy	Cones	APBP
	(nocc,all)	(nocc,all)	(nocc,all)	(nocc,all)	(%)
SOS$^+$	(1.45,1.63)	(0.21,0.32)	(3.13,8.45)	(2.43,7.10)	4.30
PM(Bleyer et al., 2011)	(2.09,2.33)	(0.21,0.39)	(2.99,8.16)	(2.47,7.80)	4.59
SOS$^+$(FP)	(1.58,1.81)	(0.21,0.31)	(5.67,11.0)	(2.57,7.70)	5.37
NLF(Yang, 2012)	(1.47,1.85)	(0.25,0.42)	(6.01,11.6)	(2.87,8.45)	5.48
AW(Yoon and Kweon, 2006)	(1.38,1.85)	(0.71,1.19)	(7.88,13.3)	(3.97,9.79)	6.67
SG (Hirschmuller, 2005)	(3.26,3.96)	(1.00,1.57)	(6.02,12.2)	(3.06,9.75)	7.50
SDDS(Wang et al., 2012)	(3.31,3.62)	(0.39,0.76)	(7.65,13.0)	(3.99,10.00)	7.19
HA(Min et al., 2011)	(2.47,2.71)	(0.74,0.97)	(8.31,13.8)	(3.86,9.47)	7.33

Table 6.2: Disparity map evaluation for non occlusion (nocc), all regions, and average percent bad pixels (APBP).

Evaluation for Refined Disparity Maps

Figure 6.10 shows the refined disparity maps for our constrained SOS$^+$(FP) and SOS$^+$ algorithms, and the quality evaluation are listed in Table 6.2, with the rank 23 and 10 in the Middlebury benchmark for SOS$^+$ and SOS$^+$(FP) respectively. The quality of the SOS$^+$(FP) algorithm is similar to PatchMatch (rank 22) with main differences coming from the ground region of the teddy image, which can only be recovered by using oriented planes. However, the processing time of SOS$^+$(FP) is much faster than other investigated stereo algorithms. In practice, SOS$^+$ is more suitable for the scene with large slanted surfaces, and SOS$^+$(FP) is more efficient for time-sensitive applications.

Discussion

We proposed a novel approach to approximate local structures by slanted plane surfaces. By integrating the SOS scheme, the exploited structures are complete and compact enough for stereo without using any iterative plane refinement procedure as other plane-based algorithms. Our propagation-based stereo scheme is more efficient than state-of-art stereo methods. As shown in our experimental evaluation, our method maintains the quality of the exhaustive disparity estimation at significantly lower computational costs.

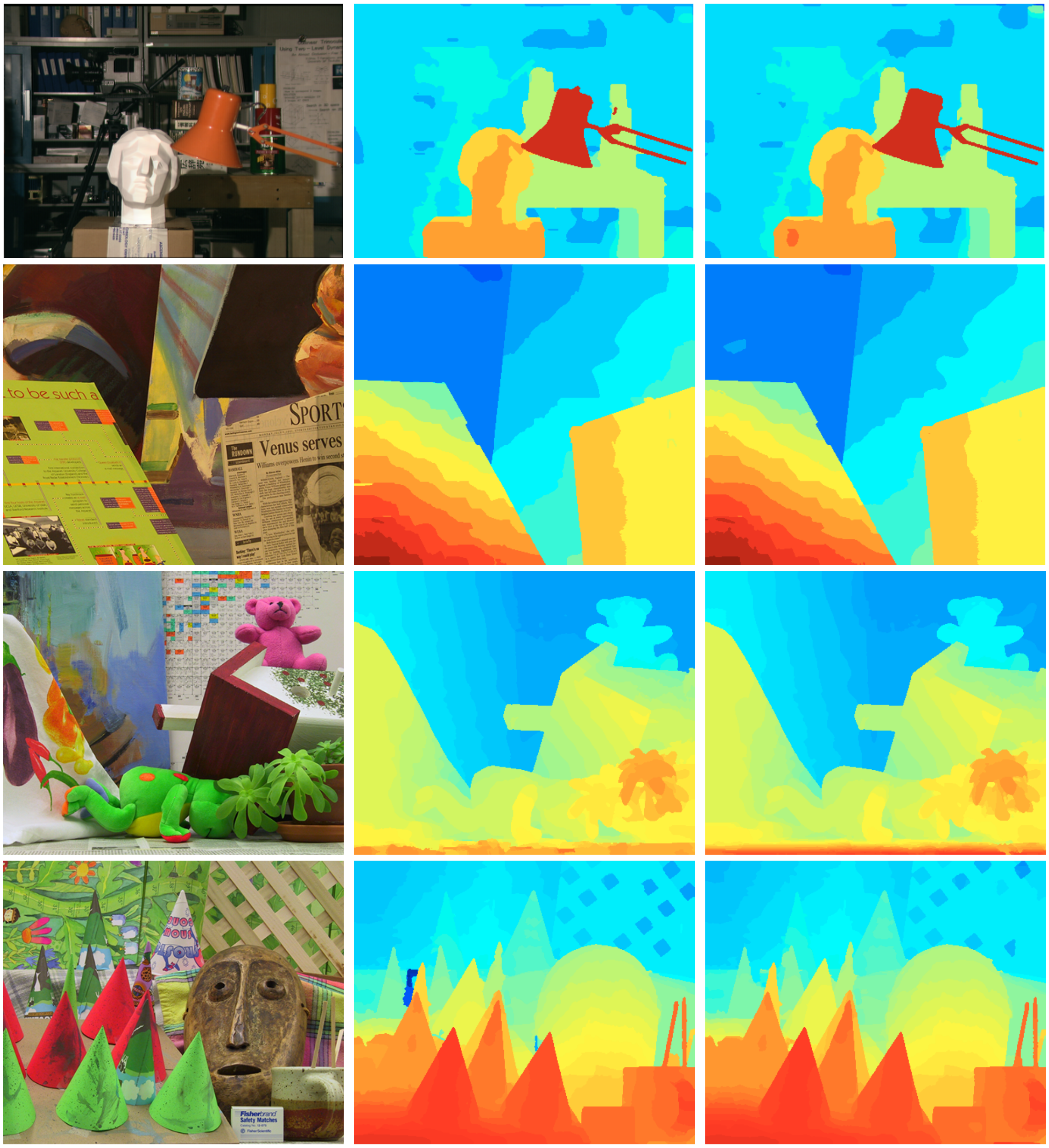


Figure 6.10: Refined disparity maps for SOS$^+$(FP) (middle column) and SOS$^+$ (right column) on Middlebury Benchmark.

CHAPTER 7: CONCLUSION

Summary

This dissertation focuses on the efficiency and accuracy of high resolution stereo. Novel sampling schemes are presented to reduce the computational complexity for most stereo algorithms, efficient propagation-based stereo algorithm is proposed by utilizing robust local structures, and a tractable framework of integrating depths and normals is introduced to improve the accuracy of recovered surfaces. The techniques proposed in this thesis remarkably improve the performance of high-resolution stereo, and can be easily incorporated into existing systems. The problem of high resolution stereo is studied in three aspects.

Search Space Refinement

The depth range of the entire scene on the image must cover the depths of all the objects, and the larger the depth range is, the more computation costs are wasted on unsuccessful localizations and comparisons. In Chapter 3 we pointed out that the pixel's search space is only relevant to the depth range of the local patch, which is much smaller than the search space of the entire scene. Two search space reduction methods are proposed: SDDS (Chapter 3) and SOS (Chapter 4). The key insight of SDDS is that correct depths tend to keep low matching costs, while the matching costs of incorrect depths are random. Thus by sampling matching costs several times, there will be observable differences between correct and incorrect depths on the aggregated cost curve. We compute the matching costs of a sparse pattern of pixels only on the reduced candidate depth set, and then vote the depths of other pixels. Our SDDS outperforms, in terms of solution quality and speed, state of the art work on complexity reduction for local stereo (Min et al., 2011). To overcome the shortcoming of fixed sampling schemes, we propose a statistical framework to analyze the correlation between the complexity of the local structure and the probability of encountering a new depth. According to the SPRT theory, the accuracy of the candidate depth set obtained by our sequential sampling scheme is the same as any over-sampling scheme with a fixed number of samples. Since SOS is an incremental sampling model, the compactness of the candidate set is much better than any fixed random sampling. By integrating SOS as a pre-processing step, state-of-the-art stereo methods can achieve the same quality with much less processing time.

Local Structures Estimation and Aggregation

High quality reconstruction usually requires sophisticated aggregation structures other than fronto-parallel planes to obtain more accurate and reliable matching costs. Thus the accuracy of stereo will be heavily influenced by the accuracy of estimated structures of local patches. In Chapter 5, we use natural cubic spline to construct surflets (local patches), and show that surflet-based matching is more accurate than fronto-parallel and oriented plane based matching. The surflet is a more accurate representation of the local structure, which is generated based on the surface normal information. For applications without normal information, we also showed how to use oriented planes to approximate local structures by pixel-wise sampling in Chapter 6, where the completeness and robustness of the extracted structure pattern is also guaranteed by SPRT theory.

An implicit assumption of stereo is the local smoothness property: neighboring pixels tend to have similar geometric structures. Therefore, the task of dense depth map stereo can be translated into extracting all distinct local structures and propagating them to other pixels. With extracted local structures, a structure propagation algorithm is presented in Chapter 6, which is much more efficient than state of the art stereo approaches (Bleyer et al., 2011; Min et al., 2011). The benefit of using oriented plane for cost aggregation is that pixels within the aggregation window do not need to have the same depth, so that the aggregated matching cost on slanted surface is more accurate than the cost aggregated from fronto-parallel plane. Furthermore, the plane refinement procedure is not necessary for our propagation algorithm, due to the sufficient structure pattern explored by our optimal sampling scheme. The quality of photo-consistency-based stereo is restricted to the image resolution, which may not satisfy the requirement of high quality surface modeling. Since surface normal provides fine geometry information of the object, Chapter 5 showed a surflet stitching algorithm to generate a continuous surface with automatic discontinuity detection, and a stereo framework that integrates the normal and color information.

Future Work

Potential directions for future work include a self-adapting sampling framework, SPRT-based Model Completeness Validation, adaptive aggregation structures, and a complete stereo framework integrating color and normal.

A Self-adapting Sampling Framework

The sequential optimal sampling (SOS) scheme proposed in Chapter 4 follows a quad-tree search strategy, which iteratively divides the investigated region into four equal sub-regions. However, such fixed separation pattern does not consider the true structure of the local patch, which prevents to find the optimal solution. For example, if the depth range of the patch center was much larger than the surrounding areas, each of them will have a large search space by just dividing the region into four equal sub-patches. Obviously, the optimal solution is to separate the center and the surrounding regions, which requires reforming the investigated regions based on previous sampling results. Hence, a challenging future work is to design a self-adapting sampling scheme that can adjust sampling regions to be consistent with local structures.

SPRT-based Model Completeness Validation

Our sampling scheme can also be extended into object modeling. To obtain a complete 3D surface of an object, one usually needs to capture multiple images at different positions, but how to evaluate the completeness is not a trivial problem. Suppose there has been a rough 3D model and we need to check whether the new image contains any new region or feature missing from the current model. We can use the SOS model judge whether all local structures on the new image have been fully exploited with very few sampled pixels. Furthermore, the SOS model will also tell us where those missing structures should be. The feedback of the optimal sampling model will help us avoid running stereo algorithms on overlapping areas, so that we can just focus on the missing region, and complete the model by capturing more details (zooming in) or moving to a better viewing position.

Adaptive Aggregation Structures

Another future work would be a closed loop adaptation of aggregation structure. So far, our method extends from fronto-parallel plane to oriented plane, but still not enough for complicated regions. It is interesting to evolve more complex aggregation structures, like curves or even arbitrary shapes, and can also be incorporated with segmentation results like utilizing the information of superpixels. Also for large aggregation window, we only need to aggregate costs from some of the most distinct neighbors, and how to find these good neighbors is an interesting problem.

A General Stereo Framework Integrating Color and Normals

Photo-consistency stereo approaches recover depths based on color cues, while surface from normals is based on geometry information. On one hand, it is easy to get plenty of high quality input images, but the quality of recovered depth maps sometimes is not good enough; for normal approaches, the input image is difficult to capture and sometimes is very noisy, but it is able to recover the details even for homogenous regions. Ideally, we could get better depth estimation by combining these two approaches. The precision of color-based stereo approaches is limited by differences of matching costs, and the precision of normal approaches depends on pixel's normal and its depth (only in perspective geometry). A theoretical analysis of the uncertainty for depth from color and normals is needed to establish a general stereo framework with mixed information.

BIBLIOGRAPHY

Agrawal, A., Raskar, R., and Chellappa, R. (2006). What is the range of surface reconstructions from a gradient field? In *ECCV*.

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. In *IJCV*, pages 2(3):283–310.

Beeler, T., Bickel, B., Beardsley, P., Sumner, B., and Gross, M. (2010). High-quality single-shot capture of facial geometry. *ACM Trans. Graph.*, 29(4):40:1–40:9.

Besse, F., Rother, C., Fitzgibbon, A., and Kautz, J. (2012). Pmbp: Patchmatch belief propagation for correspondence field estimation. In *BMVC*.

Bleyer, M., Rhemann, C., and Rother, C. (2011). Patchmatch stereo - stereo matching with slanted support windows. In *BMVC*.

Bolles, R. C., Baker, H. H., and Hannah, M. J. (1993). The jisct stereo evaluation. In *DARPA Image Understanding Workshop*, pages 263–274.

Bradley, D., Boubekeur, T., and Heidrich, W. (2008). Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In *CVPR'08*.

Carceroni, R. L. and Kutulakos, K. N. (2002). Multi-view scene capture by surfel sampling: From video streams to non-rigid 3d motion, shape and reflectance. *International Journal of Computer Vision*, pages 175–214.

Chow, C. K. and Yuen, S. Y. (2009). Recovering shape by shading and stereo under lambertian shading model. *Int. J. Comput. Vision*, 85(1):58–100.

Chum, O. and Matas, J. (2008). Optimal randomized ransac. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1472–1482.

Collins, R. T. (1996). A space-sweep approach to true multi-image matching. In *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, CVPR '96, pages 358–, Washington, DC, USA. IEEE Computer Society.

Dial, G. and Grodecki, J. (2005). Rpc replacement camera models. In *Proceedings of ASPRS 2005 Conference*.

Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient belief propagation for early vision. In *CVPR (1)*, pages 261–268.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S., and Pollefeys, M. (2010). Building rome on a cloudless day. In *Proceedings of the 11th European conference on Computer vision: Part IV*, pages 368–381.

Frankot, R. T. and Chellappa, R. (1989). Shape from shading. chapter A method for enforcing integrability in shape from shading algorithms, pages 89–122. MIT Press, Cambridge, MA, USA.

Furukawa, Y., Curless, B., Seitz, S. M., and Szeliski, R. (2010). Towards internet-scale multi-view stereo. In *CVPR*.

Furukawa, Y. and Ponce, J. (2007). Accurate, dense, and robust multi-view stereopsis. In *CVPR'07*.

Furukawa, Y. and Ponce, J. (2010). Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1362–1376.

Gallup, D., Frahm, J.-M., Mordohai, P., Yang, Q., and Pollefeys, M. (2007). Real-time plane-sweeping stereo with multiple sweeping directions. In *CVPR*.

Gupta, R., Cho, S.-Y., and Gambini, A. (2010). Real-time stereo matching using adaptive binary window. In *3DPVT*.

Hannah, M. J. (1974). *Computer Matching of Areas in Stereo Images*. Stanford University.

Hansard, M., Lee, S., Choi, O., and Horaud, R. (2012). *Time-of-Flight Cameras: Principles, Methods and Applications*. Springer Publishing Company, Incorporated.

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

Hawe, S., Kleinsteuber, M., and Diepold, K. (2011). Dense disparity maps from sparse disparity measurements. In *ICCV*.

He, K., 0001, J. S., and Tang, X. (2010). Guided image filtering. In Daniilidis, K., Maragos, P., and Paragios, N., editors, *ECCV (1)*, volume 6311 of *Lecture Notes in Computer Science*, pages 1–14. Springer.

Hernandez Esteban, C., Vogiatzis, G., and Cipolla, R. (2008). Multiview photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:548–554.

Hirschmuller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *CVPR*.

Hirschmuller, H. (2008). Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Anal. Mach. Intell.*

Hirschmuller, H., Buder, M., and Ernst, I. (2012). Memory efficient semi-global matching. I-3.

Hirschmuller, H. and Scharstein, D. (2007). Evaluation of cost functions for stereo matching. In *CVPR*. IEEE Computer Society.

Horn, B. K. P. and Brooks, M. J. (1986). The variational approach to shape from shading. *Comput. Vision Graph. Image Process.*, 33:174–208.

Horovitz, I. and Kiryati, N. (2001). Bias correction in photometric stereo using control points. In *Proceedings of the Vision Modeling and Visualization Conference 2001*, VMV ’01, pages 391–398. Aka GmbH.

Hsieh, J.-W., Liao, H.-Y. M., Ko, M.-T., and Fan, K.-C. (1995). Wavelet-based shape from shading. *Graph. Models Image Process.*, 57:343–362.

Humenberger, M., Engelke, T., and Kubinger, W. (2010). A census-based stereo vision algorithm using modified semi-global matching and plane fitting to improve matching quality.

Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A., and Fitzgibbon, A. (2011). Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST ’11, pages 559–568. ACM.

Kanade, T., Kato, H., Kimura, S., Yoshida, A., and Oda, K. (1995). Development of a video-rate stereo machine. In *Proc. of International Robotics and Systems Conference (IROS ’95), Human Robot Interaction and Cooperative Robots*, volume 3, pages 95 – 100.

Karaçali, B. and Snyder, W. (2003). Reconstructing discontinuous surfaces from a given gradient field using partial integrability. *Comput. Vis. Image Underst.*, 92:78–111.

Kim, C., Zimmer, H., Pritch, Y., Sorkine-Hornung, A., and Gross, M. (2013). Scene reconstruction from high spatio-angular resolution light fields. *ACM Trans. Graph.*

Klaus, A., Sormann, M., and Karner, K. (2006). Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. ICPR.

Klette, R. and Schluns, K. (1996). Height data from gradient fields. In *Proceedings of SPIE (the international Society for Optical Engineering) on Machine Vision Applications, Architectures, and Systems Integration*, pages 204–215.

Kolev, K., Pock, T., and Cremers, D. (2010). Anisotropic minimal surfaces integrating photoconsistency and normal information for multiview stereo. In *ECCV ’10*, pages 538–551.

Kolmogorov, V. and Zabih, R. (2001). Computing visual correspondence with occlusions via graph cuts. Technical report, Ithaca, NY, USA.

Kolmogorov, V. and Zabih, R. (2002). Multi-camera scene reconstruction via graph cuts. In *Proceedings of the 7th European Conference on Computer Vision-Part III*.

Koschan, A., Rodehorst, V., and Spiller, K. (1996). Color stereo vision using hierarchical block matching and active color illumination. In *Proceedings of the 1996 International Conference on Pattern Recognition (ICPR ’96) Volume I - Volume 7270*, ICPR ’96, pages 835–, Washington, DC, USA. IEEE Computer Society.

Kovesi, P. (2005). Shapelets correlated with surface normals produce surfaces. In *Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2*.

Lee, K. M. and Kuo, C. C. J. (1996). Shape from photometric ratio and stereo. *Journal of Visual Communication and Image Representation*, 7:155–162.

Liu, Y., Cao, X., Dai, Q., and Xu, W. (2009). Continuous depth estimation for multi-view stereo. In *CVPR'09*, pages 2121–2128.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ICCV '99, pages 1150–, Washington, DC, USA. IEEE Computer Society.

Lu, J., Yang, H., Min, D., and Do, M. N. (2013). Patch match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation. In *CVPR*, pages 1854–1861. IEEE.

Marr, D. and Poggio, T. (1976). Cooperative computation of stereo disparity. In *Science*, page 194:283C287.

Marr, D. C. and Poggio, T. (1979). A computational theory of human stereo vision. In *Proceedings of the Royal Society of London*, page B 204:301C328.

Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., and Angelova, A. (2007). Computer vision on mars. *Int. J. Comput. Vision*, 75(1):67–92.

Mattoccia, S., Giardino, S., and Gambini, A. (2009). Accurate and efficient cost aggregation strategy for stereo correspondence based on approximated joint bilateral filtering. In *ACCV*.

Mei, X., Sun, X., Zhou, M., Jiao, S., Wang, H., and Zhang, X. (2011). On building an accurate stereo matching system on graphics hardware. In *GPUCV*.

Michael, M., Salmen, J., Stallkamp, J., and Schlipsing, M. (2013). Real-time stereo vision: Optimizing semi-global matching.

Min, D., Lu, J., and Do, M. N. (2013). Joint histogram-based cost aggregation for stereo matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10):2539–2545.

Min, D., Lu, J., and Minh, N. D. (2011). A revisit to cost aggregation in stereo matching. In *ICCV*.

Nehab, D., Rusinkiewicz, S., Davis, J., and Ramamoorthi, R. (2005). Efficiently combining positions and normals for precise 3d geometry. *ACM Trans. Graph.*, 24(3).

Noakes, L. and Kozera, R. (2003). Nonlinearities and noise reduction in 3-source photometric stereo. *J. Math. Imaging Vis.*, 18:119–127.

Nozick, V., Michelin, S., and Arqus, D. (2005). Image-based rendering using plane-sweeping modelisation. In *MVA'05*, pages 468–471.

Petschnigg, G., Szeliski, R., Agrawala, M., Cohen, M., Hoppe, H., and Toyama, K. (2004). Digital photography with flash and no-flash image pairs. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 664–672, New York, NY, USA. ACM.

Raguram, R., Chum, O., Pollefeys, M., Matas, J., and Frahm, J.-M. (2013). Usac: A universal framework for random sample consensus. *IEEE Trans. Pattern Anal. Mach. Intell.*

Rhemann, C., Hosni, A., Bleyer, M., Rother, C., and Gelautz, M. (2011). Fast cost-volume filtering for visual correspondence and beyond. In *CVPR*.

Rusinkiewicz, S., Hall-Holt, O., and Levoy, M. (2002). Real-time 3d model acquisition. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 438–446.

Scharstein, D. (1994). Matching images by comparing their gradient fields. In *ICPR*, volume 1, pages 572–575.

Scharstein, D. and Pal, C. (2007). Learning conditional random fields for stereo. In *CVPR*. IEEE Computer Society.

Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*.

Scharstein, D. and Szeliski, R. (2003a). High-accuracy stereo depth maps using structured light. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'03, pages 195–202, Washington, DC, USA. IEEE Computer Society.

Scharstein, D. and Szeliski, R. (2003b). High-accuracy stereo depth maps using structured light. In *CVPR (1)*, pages 195–202. IEEE Computer Society.

Seitz, P. (1989). Using local orientation information as image primitive for robust object recognition. In *SPIE Visual Communications and Image Processing IV*, volume 1199, pages 1630–1639.

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., and Szeliski, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 519–528, Washington, DC, USA. IEEE Computer Society.

Sizintsev, M. (2008). Hierarchical stereo with thin structures and transparency. In *Proceedings of the 2008 Canadian Conference on Computer and Robot Vision*.

Smisek, J., Jancosek, M., and Pajdla, T. (2013). 3d with kinect. In *Consumer Depth Cameras for Computer Vision Advances in Computer Vision and Pattern Recognition*.

Tappen, M. F. and Freeman, W. T. (2003). Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*.

Tombari, F., Mattoccia, S., and Addimanda, E. (2008). Near real-time stereo based on effective cost aggregation. In *ICPR*.

Van Meerbergen, G., Vergauwen, M., Pollefeys, M., and Van Gool, L. (2002). A hierarchical symmetric stereo algorithm using dynamic programming. *Int. J. Comput. Vision*.

Veksler, O. (2006). Reducing search space for stereo correspondence with graph cuts. In *BMVC*.

Vlasic, D., Peers, P., Baran, I., Debevec, P., Popović, J., Rusinkiewicz, S., and Matusik, W. (2009). Dynamic shape capture using multi-view photometric stereo. In *ACM SIGGRAPH Asia 2009 papers*, pages 174:1–174:11, New York, NY, USA. ACM.

Wald, A. (1947). *Sequential Analysis*. Dover.

Wang, L., Jin, H., and Yang, R. (2008). Search space reduction for mrf stereo. In *Proceedings of the 10th European Conference on Computer Vision: Part I*.

Wang, Y., Dunn, E., and Frahm, J.-M. (2012). Increasing the efficiency of local stereo by leveraging smoothness constraints. In *3DIMPVT*.

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

Yang, Q. (2012). A non-local cost aggregation method for stereo matching. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yang, Q., 0002, L. W., and Ahuja, N. (2010). A constant-space belief propagation algorithm for stereo matching. In *CVPR*, pages 1458–1465. IEEE.

Yang, Q., Wang, L., Yang, R., Stewénius, H., and Nistér, D. (2009). Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *PAMI*.

Yang, R. and Pollefeys, M. (2003). Multi-resolution real-time stereo on commodity graphics hardware.

Yang, R., Welch, G., and Bishop, G. (2002). Real-time consensus-based scene reconstruction using commodity graphics hardware. In *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, PG '02, pages 225–, Washington, DC, USA. IEEE Computer Society.

Yoon, K. J. and Kweon, I. S. (2006). Adaptive support-weight approach for correspondence search. *IEEE Trans. PAMI*, 28:650–656.

Yu, W., Chen, T., and Hoe, J. C. (2009). Real time stereo vision using exponential step cost aggregation on gpu. In *Proceedings of the 16th IEEE International Conference on Image Processing*, ICIP'09, pages 4225–4228, Piscataway, NJ, USA. IEEE Press.

Zabih, R. and Woodfill, J. (1994). Non-parametric local transforms for computing visual correspondence. In *Proceedings of the Third European Conference on Computer Vision (Vol. II)*, ECCV '94, pages 151–158, Secaucus, NJ, USA. Springer-Verlag New York, Inc.

Zabulis, X. and Daniilidis, K. (2004). Multi-camera reconstruction based on surface normal estimation and best viewpoint selection. In *3DPVT'04*, pages 733–740.

Zabulis, X. and Kordelas, G. (2006). Efficient, precise, and accurate utilization of the uniqueness constraint in multi-view stereo. In *3DPVT'06*, pages 137–144.