The Dissertation Committee for Thayne Richard Coffman
certifies that this is the approved version of the following dissertation:

Stochastic Methods in Computational Stereo

Committee:

Alan C. Bovik, Supervisor

J. K. Aggarwal

Brian L. Evans

Risto Miikkulainen

Edward J. Powers

Stochastic Methods in Computational Stereo

by

Thayne Richard Coffman, S.B.; M.Eng.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2011

Dedicated to those who have come before me in this field,

and those who will come after.

Acknowledgements

First and foremost I acknowledge God. He is the Great Scientist who created epipolar geometry and the other laws of our universe, and the Great Teacher who gives us the joy of rediscovering them. He also moved many things into place and out of place to let me do this.

Second to God, there is Professor Bovik. I am deeply grateful for the specific things he's taught me, for his guidance, and for the sharing of his perspectives. He has of course taught me many lessons about research. These include looking at problems from the broadest possible perspective, digging relentlessly for the root causes of unexpected results, and the value of formulating problems in the continuous domain first. He has also taught me many other lessons – some from direct instruction and others from simply watching the way he approaches and enjoys his work after 30 years in the field.

Thank you to my wife Mae, who gets my permanent vote for the best wife ever. Your encouragement and forbearance have been amazing. Thank you for biting your lip through five years of me telling you I was two years from finishing, and thank you for actually encouraging me to keep going even though I bet you had some doubts about my perception of time. You were willing to stay in Austin so long you eventually decided

you wanted to stay permanently! You continue to be a source of perspectives and ideas that make me a better person, and that I wouldn't come to on my own.

To my son Asher, if you ever read this, know that you were one of my biggest inspirations to push hard at the finish. Playing with you and watching you learn new things gives me joy that I simply did not understand before you got here.

Thank you to my parents for being some of my first and best examples of God's love. Thank you for your love and support through the years, and providing for my earlier education. Thank you also for teaching me the value of learning and how to take personal responsibility for bettering myself. Many of the pieces of me that I like started with you.

Thank you to 21st Century Technologies corporately, and to Sherry Marcus, Darrin Taylor, and Irene Williams individually for financial support, the flexibility to pursue this degree, and for building a company culture where it was encouraged. Thanks to all the 21CT employees who fulfilled responsibilities and completed tasks that I would have otherwise have been expected to do. Thanks also to Irene and Robert for a critical conversation at a low point in the process.

The work in this dissertation is my own, but pieces of it were done in the context of 21CT contracts to which others contributed and by which I bought useful things like food. Thank you to the 21CT employees that worked alongside me these years, and particularly the former Sensor Exploitation Group – Todd, Ron, Matt, Shaun, Will, and Terry. Long may we reign as Halloween costume contest champions.

Having a full-time off-campus job meant that I didn't invest as much time in building relationships with other members of the LIVE lab. In my interactions with you, though, you were always friendly and eager to help and support however you could. Specific thanks to Kalpana, Umesh, Shalini, Gautam, Hamid, Abtine, Yang, Mehul, Joonsoo, and Sina. I truly appreciate your welcoming and supportive attitudes, and also value the times I did interact with you on technical and non-technical things.

Thank you to the professors on my committee, who each contributed to this endeavor in different ways. Thanks to Professor Aggarwal for our long history of enjoyable and productive collaborations. Thanks to Professor Evans for your attention to detail, help in navigating and planning for various UT processes, and focus in your classes on teaching students how to write research papers. Thanks to Professor Miikkulainen for your contributions to our joint efforts and your easygoing but committed work style while making them. Special thanks to Professor Powers for going to great lengths to participate in both my qualifying proposal and my dissertation defense.

Finally, thank you to the contract sponsors who provided financial support for pieces of this work. Without that support, this would not have happened. Thank you specifically to T.J. Klausutis and his associates, who started this whole ball rolling many years ago with a decision to fund this type of work at 21CT.

This work was supported in part by the United States Air Force under contracts FA8651-04-C-0233 and FA8651-05-C-0117. The description of that work was approved

for public release by the Air Force Research Laboratory under Public Affairs Case Number 96ABW-2009-0122. Approved for Public Release, Distribution Unlimited.

This work was supported in part by the U.S. Army Aviation and Missile Command and the Defense Advanced Research Projects Agency (DARPA) under contract W31P4Q-09-C-0464. The description of that work was approved for public release by DARPA. The views, opinions, and/or findings contained in this document are those of the author and should not be interpreted as representing the official views or policies, either express or implied, of DARPA or the Department of Defense. Approved for Public Release, Distribution Unlimited.

Stochastic Methods in Computational Stereo

Publication No. _____

Thayne Richard Coffman, Ph.D.

The University of Texas at Austin, 2011

Supervisor: Alan C. Bovik

Computational stereo estimates 3D structure by analyzing visual changes between two or more passive images of a scene that are captured from different viewpoints. It is a key enabler for ubiquitous autonomous systems, large-scale surveying, virtual reality, and improved techniques for compression, tracking, and object recognition. The fact that computational stereo is an under-constrained inverse problem causes many challenges. Its computational and memory requirements are high. Typical heuristics and assumptions, used to constrain solutions or reduce computation, prevent treatment of key realities such as reflection, translucency, ambient lighting changes, or moving objects in the scene. As a result, a general solution is lacking.

Stochastic models are common in computational stereo, but stochastic algorithms are severely under-represented. In this dissertation I present two stochastic algorithms and demonstrate their advantages over deterministic approaches.

I first present the Quality-Efficient Stochastic Sampling (QUESS) approach. QUESS reduces the number of match quality function evaluations needed to estimate dense stereo correspondences. This facilitates the use of complex quality metrics or metrics that take unique values at non-integer disparities. QUESS is shown to outperform two competing approaches, and to have more attractive memory and scaling properties than approaches based on exhaustive sampling.

I then present a second novel approach based on the Hough transform and extend it with distributed ray tracing (DRT). DRT is a stochastic anti-aliasing technique common to computer rendering but which has not been used in computational stereo. I demonstrate that the DRT-enhanced approach outperforms the unenhanced approach, a competing variation that uses re-accumulation in the Hough domain, and another baseline approach. DRT's advantages are particularly strong for reduced image resolution and/or reduced accumulator matrix resolution. In support of this second approach, I develop two novel variations of the Hough transform that use DRT, and demonstrate that they outperform competing variations on a traditional line segment detection problem.

I generalize these two examples to draw broader conclusions, suggest future work, and call for a deeper exploration by the community. Both practical and academic

gaps in the state of the art can be reduced by a renewed exploration of stochastic computational stereo techniques.

Table of Contents

List of Tables

List of Figures

Chapter 1.

Introduction

1.1 Motivations

We live in a dynamic 3D physical world. Many important tasks require understanding the world's 3D structure and how it changes over time. Computational stereo is the process of estimating 3D structure from two or more passive images of a scene captured from different viewpoints. Structure is estimated by analyzing visual disparity – the changes of objects' apparent positions caused by the change in viewpoint. A general and robust approach to computational stereo would have widespread impact.

An explosive proliferation of intelligent autonomous systems is approaching. The DARPA Grand Challenge sought to achieve autonomous land navigation over 100 miles of unmodified roadways. In 2004 the challenge ended with zero successful teams; in 2005, five of 23 teams succeeded [29]. To be effective, autonomous systems must perceive the world in 3D [83], as did every successful team in the DARPA 2005 Grand Challenge.

Computational stereo is the most likely way to enable ubiquitous autonomous systems. Inexpensive active sensors (e.g., Microsoft's Kinect) have limited ranges. Long-

range active sensors are too expensive and complex for widespread use. Passive sensors have lower size, power, and weight requirements, reducing costs further. Support for the passive approach is drawn from nature, where most high-functioning animals and humans rely heavily on passive vision for navigation and interaction with their environment.

Autonomous systems applications are widespread, and can be found across the military, government, and commercial sectors. They provide the ability to remove humans from tedious or dangerous situations, reduce costs, improve service life, reduce risk to human lives, and multiply the effects of limited manpower.

Computational stereo also has many practical applications outside of autonomous systems. It can be used for high-accuracy metrology of buildings and large outdoor areas where other techniques are infeasible. Large-scale 3D modeling supports remote surveying and virtual reality walkthroughs for training and environment familiarization. 3D modeling with computational stereo can also help preserve knowledge and representations of historical artifacts (small or large) when the artifacts themselves are destined to decay over time.

The potential benefits expand further if we consider downstream image processing and computer vision algorithms that can exploit the 3D models generated by computational stereo. Many object recognition approaches now leverage 3D object models instead of 2D. Emerging video compression techniques use 3D models to improve compression rates [5]. The tracking algorithm of [74] uses 3D environment

models to predict occlusion and de-occlusion in order to improve track segment fusion. A colleague and I developed that tracking algorithm based on models created by the QUESS approach of Chapter 3.

Finally, the pervasive existence of autonomous systems that would be allowed by robust and general computational stereo would facilitate future research in autonomous navigation, path planning, emergent behaviors, and multi-agent systems.

1.2 Challenges

After 40 years of research, the community has yet to produce an approach to computational stereo that is sufficiently general and robust.

Broad challenges remain, fundamentally caused by the fact that computational stereo is an under-constrained inverse problem. As a result there are strong limits on what can be determined solely from the input [71]. Heuristics or assumptions must be added to define unique or optimal solutions. Further simplifying assumptions are also usually needed to make computations tractable.

These assumptions and heuristics often make the resulting approaches less general or less robust to realistic scenarios. Common assumptions brush away realities such as specular reflection, translucency, ambient lighting changes, complex structures (e.g., trees), unconstrained camera motion, camera parameter uncertainty, and even moving objects in the scene.

Computational complexity and memory consumption remain significant issues for many leading approaches even with the use of simplifying assumptions. This places severe limits on reconstruction in terms of spatial extent, range resolution, absolute range, and recovered detail.

Unconstrained camera motion can cause reconstruction problems to become ill-posed. Some of the problematic motions are extremely common – for example motion directly forward towards the scene. Any assumptions limiting camera geometry (like motion restrictions or the use of multi-camera rigs), however, adversely impacts cost or generality.

Some of the most interesting and far-reaching applications are in the area of vision-aided navigation for autonomous systems. In addition to the challenges described above, those applications add the need for real-time operation and often the need to use limited computational resources. They also typically require high absolute ranges to be modeled with compact sensor geometries.

Despite all these challenges, achieving 3D perception from passive 2D imagery remains an active research field because the potential benefits are so significant. Most humans and animals navigate the world relying primarily on passive binocular vision, and have amazing capabilities that are wholly unmatched by artificial systems.

A final challenge is indicated by the fact that human and animal visual perception exploit much more than just visual disparity to perceive 3D structure. We understand

natural systems to use a complex mix of low- and high-level visual cues, proprioception, and even semantic information [113]. Computational stereo algorithms attempt to estimate 3D structure using only low-level visual cues, but the only truly successful general solutions to date use many other sources of information as well.

1.3 Approach

I explore stochastic computational stereo algorithms. There is a dichotomy in the current literature. Computational stereo algorithms based on stochastic models are widely represented, but stochastic algorithms are very rare. The use of stochastic algorithms can provide advantages over deterministic algorithms in certain important scenarios.

In some cases stochastic algorithms can reduce computational complexity while retaining accuracy or other desirable performance qualities. One example is to stochastically sample a signal or computation that would otherwise be exhaustively sampled. This can often reduce runtime. It can also address some less common scenarios in which exhaustive sampling is in fact impossible – e.g., the sampling of a function that is defined on a real-valued domain.

In other cases stochastic algorithms can improve accuracy or other desirable performance qualities at the cost of additional computational complexity. For example, the rough approximations used by some deterministic techniques can be replaced by improved stochastic approximations. Here, stochastic techniques improve the estimation of quantities that are infeasible to compute analytically.

This dissertation develops two specific examples of stochastic computational stereo algorithms, and demonstrates their improved performance over existing approaches. The advantages of these stochastic techniques are generalized beyond the specific examples through analysis. The generalized examples are used as inspiration to call for future work in which these or other techniques could be used to improve additional stereo approaches. They are ultimately used to motive the need for further exploration of stochastic computational stereo algorithms.

1.4 Contributions

The primary goal of this dissertation is to show by example(s) that stochastic computational stereo algorithms can provide benefits over deterministic approaches, and then to argue that the research community should deepen its exploration of stochastic computational stereo algorithms. The specific examples are generalized by analyzing how and why they improve performance.

The following is an overview of the contributions presented in this dissertation:

1. The Quality-Efficient Stochastic Sampling (QUESS) stereo approach

   I present a novel computational stereo approach that stochastically samples the local match quality function using a cooperative algorithm. The approach is applied to 3D reconstruction from calibrated stereo image pairs and from calibrated monocular video. I demonstrate competitive performance on a popular

image pair benchmark. I also demonstrate superior performance to two alternatives on a calibrated video dataset.

2. Extension of the Hough transform with distributed ray tracing (DRT)

   I extend the Hough transform with DRT to produce two novel variations, the distributed ray Hough transform (DRHT) and a variation named Palmer-K_{DRT}. I demonstrate superior performance in a line parameter estimation task, particularly on reduced-resolution input or Hough-domain accumulators.

3. A Hough transform-based dense computational stereo approach

   I present a novel dense computational stereo approach based on the standard Hough transform. The approach is applied to a calibrated multi-view stereo benchmark and is demonstrated to outperform a naïve baseline approach.

4. Extension of Hough transform-based stereo with distributed ray tracing

   I extend the computational stereo approach of contribution #3 to use DRHT in place of the standard Hough transform. I demonstrate improved performance on a multi-view stereo benchmark, particularly on reduced-resolution input.

5. Advantages of stochastic sampling of match quality

   I analyze the advantages of QUESS over an example approach that exhaustively samples the match quality function. I show that QUESS has advantages in match quality computations, memory, and scalability. I show that QUESS makes those traits more robust to camera geometry, scene geometry, and other difficult-to-

control factors. I also show that the advantages of QUESS hold against *any* approach that exhaustively samples the match quality function.

6. Advantages of the use of DRT in computational stereo

   I discuss the advantages of using DRT over regular sampling in computational stereo algorithms. I identify additional computational stereo algorithms to which DRT could be applied in the future.

7. Identification of the need for further research of stochastic stereo algorithms

   I state and support the conjecture that that the research community should reduce the under-representation of stochastic computational stereo algorithms.

1.5 Dissertation Outline

The rest of this dissertation is organized as follows. Chapter 2 presents background information needed to understand the contributions in context. The chapter discusses computational stereo, the Hough transform, and distributed ray tracing. Computational stereo is defined in terms of its inputs, outputs, and its fundamental approach to the problem. Major issues and subcomponents are discussed and it is differentiated from closely related problems. Relevant computational stereo approaches are described. The Hough transform is described in its basic form, with common extensions, and as applied to computational stereo. Aliasing in the Hough domain is discussed. Ray tracing and its extension to distributed ray tracing are discussed. The chapter ends by describing limitations in the state of the art in these areas.

Chapter 3 presents the Quality Efficient Stochastic Sampling (QUESS) computational stereo approach, which can be applied to two-frame stereo pairs or calibrated monocular video. QUESS is evaluated on both two-frame and calibrated video inputs, and its performance is compared to alternative approaches. Its advantages are discussed and generalized.

Chapter 4 extends the Hough transform in two novel ways by incorporating distributed ray tracing (DRT) to yield the distributed ray Hough transform (DRHT) and an approach called Palmer-K_{DRT}. The two new variations are compared to alternatives on traditional line detection tasks. This is relevant because Hough transform improvements gained by extension with DRT also improve Hough transform-based computational stereo algorithms, as shown in Chapter 5.

Chapter 5 presents a novel wireframe-to-dense computational stereo algorithm based on the Hough transform. The approach is presented using the standard Hough transform (SHT). It is then extended to use an existing Hough transform variation and alternatively the DRHT from Chapter 4. The results of using SHT, DRHT, and the other variation are measured on a fundamental scene and a multi-view stereo benchmark. The advantages of DRHT are discussed and generalized.

Chapter 6 presents specific and general conclusions, as well as directions for future work.

Chapter 2.

Background

This chapter provides a brief overview of relevant background information. I begin by defining the computational stereo problem. I then briefly describe relevant issues and approaches. The Hough transform is discussed to inform extensions that are developed in Chapter 4 and applied to computational stereo in Chapter 5. Ray tracing and distributed ray tracing are then described. I end with a discussion of limitations in the existing state of the art in computational stereo.

2.1 Computational Stereo

Computational stereo is an under-constrained inverse problem in which the structure of a 3D physical scene is estimated from two or more passive images of the scene that have been captured from different viewpoints. Despite 40 years of active research, a general solution that is feasible and robust remains unknown.

Computational stereo does not encompass all methods for estimating 3D scene structure. It can be defined and differentiated from other methods by considering two of its key aspects: its input data, and the analysis of visual disparity.

2.1.1 Input Types

The definition of computational stereo typically assumes the use of *passive imaging sensors* that produce sampled 2D signals by measuring energy that is reflected or radiated from a scene. They may operate on electro-optic (EO) (a.k.a. “visible light”), infrared, ultraviolet, or other wavelengths, or may be multispectral or hyperspectral.

I do not include radar, lidar, sonar, structured light, or other sensors that transmit energy into the scene and analyze the timing or content of reflections. This definition of permissible inputs is common to most practitioners, with the possible exception of excluding structured light. Regardless of categorization, the use of computational stereo algorithms on structured light inputs has shown substantial success [104].

2.1.1.1 Uncalibrated vs. Calibrated

Calibrated stereo approaches assume that the extrinsic (position and orientation), and intrinsic parameters (field of view, skew, etc.) of the input images are known. *Uncalibrated* approaches assume that some or all of these parameters are unknown.

In the absence of any camera parameters, 3D structure estimation becomes significantly harder, and even when “solved”, little information of value is recovered. *Relative* camera position and orientation, and also intrinsic camera parameters, can be estimated using well-studied techniques [45][46]. Uncalibrated stereo approaches often estimate these parameters first and then address the calibrated problem.

Knowing relative extrinsic parameters allows 3D scene structure to be estimated up to an arbitrary similarity transform (position, orientation, and isotropic scale). Geo-registering the 3D scene requires knowledge of absolute camera position and orientation. These are typically provided by Global Positioning System (GPS) and Inertial Measurement Unit (IMU) receivers attached to the sensor. Further details of what can and cannot be recovered with different parameter knowledge are given in [42], [45], and [34].

2.1.1.2 Two-View, Multi-View, and Video Reconstruction Algorithms

A wide variety of camera configurations has been used. These often include custom platforms mounting one or more cameras (e.g., [2], [60]), which fix the cameras' relative geometry and synchronize their shutters, and then exploit those constraints to simplify computation. Stereo algorithms are often bound to one camera configuration or class of configurations. The algorithms can be categorized as follows.

Two-view approaches analyze images in pairs to extract depth by exploiting visual differences and the geometry between the two cameras, be it fixed or variable.

Multi-view approaches fuse and analyze the information in many images together. Algorithms may process the input images in pairs as in a two-frame algorithm, or process images jointly as a batch. The images may be captured from arbitrary viewpoints with significant separation and visual changes between them.

Video input is a subset of multi-view input that sometimes drives specialized approaches. Video algorithms assume that consecutive images exhibit only incremental

changes in viewpoint. This does not preclude the use of multiple sensors (e.g., binocular video). Video input can be processed by formulating multiple two-frame stereo problems (through a frame pairing process) or by applying specialized multi-view algorithms. Modeling from calibrated monocular video has received less attention than other camera configurations, although examples are available (e.g., [97], [119]).

2.1.2 Reconstruction by Analysis of Visual Disparity

2.1.2.1 Parallax

Computational stereo algorithms estimate scene structure by analyzing *parallax*, also called *disparity*, which is the apparent motion of objects in the scene as the camera or observer moves from one viewpoint to another.

The concept of parallax is shown in Figure 2-1 for an intuitive understanding. Imagine looking out the passenger's side window of a car traveling on a straight highway. In this simplest case, the motion between viewpoints is limited to translation perpendicular to the scene (called *fronto-parallel motion*), with no rotation, vertical motion, or motion towards or away from the scene. Objects that are closer to the car appear to move a larger distance (or equivalently, move faster) as the car drives. Objects that are farther away appear to move less, or more slowly. Objects at infinite distance would not appear to move at all.

Figure 2-1: The parallax effect creates visual disparity, which is a difference in the projected location of an object as viewed from two viewpoints. For horizontal camera motion, objects closer to the camera exhibit larger disparities than distant objects. The tree in the foreground appears to shift a greater distance than the house in the background. As a result it appears on opposite sides of the window in the two images.

The general case is shown more formally in Figure 2-2. Given two viewpoints and a pair of *corresponding pixels* u_A and u_B that show the same location in the scene, the depth of that location is estimated by triangulation. Rotation, vertical motion, and non-fronto-parallel motion complicate the relationship between depth and disparity.

The two cameras, *A* and *B*, are called the *left* and *right* cameras. They have origins O_A and O_B, principal axes Z_A^+ and Z_B^+, and image planes I_A and I_B. In the *projective camera model*, the 3D location $P=(X, Y, Z)'$ is projected to location $u_A=(x_A, y_A)'$ in image plane I_A through the relationship

$$(x_A, y_A)' = \left(\frac{X}{Z}, \frac{Y}{Z}\right)', \tag{1}$$

and similarly for camera B and image plane I_B.

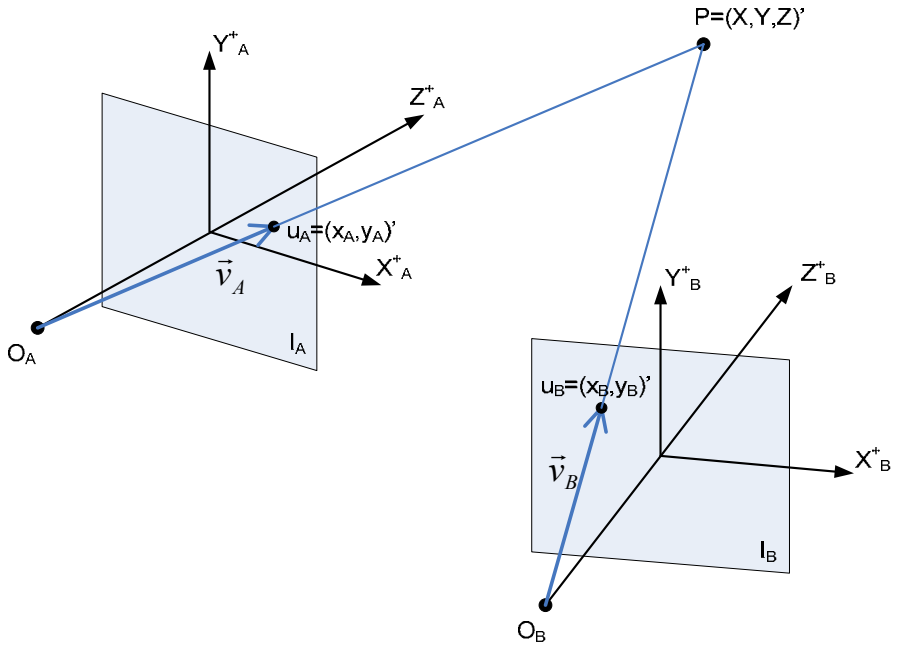


Figure 2-2: A more formal statement of two-camera geometry. Camera A has origin O_A, principal axis Z_A^+, and image plane I_A. Similarly for camera B. A point P in the scene projects to 2D locations u_A and u_B in cameras A and B, respectively.

Projection can be modeled as a linear operator in *homogeneous coordinates* [45], which append a scale factor w to each vector. The representation of a point in Euclidean space is unique up to this scale. For example, $P_{h1} = (x, y, z, 1)'$ and $P_{h2} = (2x, 2y, 2z, 2)'$ both represent the Euclidean point $P = (x, y, z)'$. Given a homogeneous 3D point P_h, the inhomogeneous 3D camera origin O_A (a length-3 vector), a rotation matrix R_A representing the relative orientation of X_A^+, Y_A^+, and Z_A^+ to world coordinates, and a matrix K_A representing camera A's intrinsic parameters (focal length, field of view, skew, etc.), the 2D homogeneous coordinates of P's projection to I_A are

$$u_{Ah} = K_A R_A \left[I \middle| -O_A \right] P_h \,. \tag{2}$$

Triangulation algorithms use the origins O_A and O_B, rotations R_A and R_B, and projections u_{Ah} and u_{Bh} to estimate P for some or all points in the scene. Absent noise and error, it is simple to intersect of $\vec{v}_A$ and $\vec{v}_B$ to find P. In practice, imperfect camera parameters and errors in matching u_A to u_B (the primary error mode of stereo algorithms) typically cause $\vec{v}_A$ and $\vec{v}_B$ to be skew. A unique solution therefore does not exist. Various triangulation strategies are discussed in [43]. The choice of strategy can have significant effects on depth estimate accuracy.

The separation between the origins O_A and O_B forms the *stereo baseline* that creates visual differences in the two images. The length of the baseline and its orientation relative to the axes Z_A^+ and Z_B^+ are also key factors that affect depth estimate accuracy.

2.1.2.2 Epipolar Geometry

Figure 2-3 illustrates a fundamental characteristic of two-camera stereo geometry called the *epipolar constraint*. Given the projection of P onto I_A or I_B, its projection onto the other image plane must lie on a known line, independent of depth. The projections u_A and u_B in 2D homogeneous coordinates are bound by the constraint $u_A^{'} F u_B = 0$, where F is a 3x3 *fundamental matrix* [45].

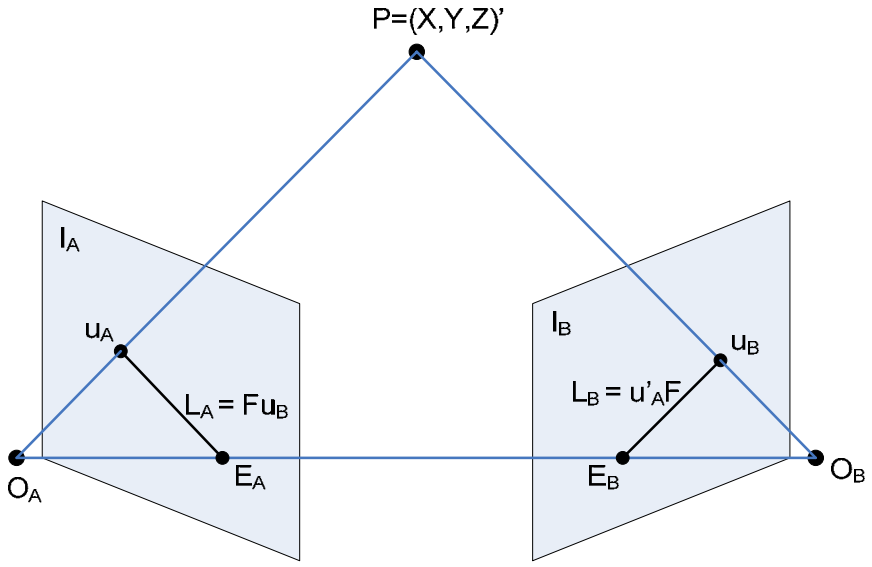


Figure 2-3: Epipolar geometry. The projections u_A and u_B of a point P in 3D space are bound by an epipolar constraint. The fundamental matrix F can be computed from camera A and B parameters. Given u_A, the projection of P onto I_B lies on the epipolar line $L_B = u_A^{'}F$. Given u_B, the projection of P onto I_A lies on the epipolar line $L_A = Fu_B$. The projections of camera origins O_B and O_A onto I_A and I_B are the *left* and *right epipoles* E_A and E_B, respectively.

F is completely defined by the relative geometry and intrinsic parameters of the cameras. It encodes that information up to an arbitrary projective transformation. When all parameters are known or estimated, F is computed as

$$F = K_B^{-T}[t]_x R K_A^{-1} \tag{3}$$

$$R = R_B R_A^{-1} \tag{4}$$

$$t = -R_A[O_B - O_A] \tag{5}$$

$$[t]_x = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}. \tag{6}$$

Vector t is the stereo baseline, the relative translation between O_A and O_B in coordinate system A. The matrix $[t]_x$ is a *cross product matrix* constructed such that the multiplication $[t]_x b$ yields the cross product vector $t \times b$.

Given u_A, u_B must lie on the right *epipolar line* $L_B = u_A^{'} F$, and given u_B, u_A must lie on the left *epipolar line* $L_A = F u_B$. The projections of O_B and O_A onto I_A and I_B are the *left* and *right epipoles* E_A and E_B, respectively. All epipolar lines pass through their image plane's epipole.

The epipolar constraint is the most important constraint in computational stereo. Without it computing pixel correspondences u_A and u_B, and thus all of computational stereo, would be largely intractable. Two-frame stereo algorithms exploit the epipolar constraint directly. Multi-frame stereo algorithms may exploit it indirectly, but it still shapes their formulation. A recent technique called epipolar spaces can help constrain correspondence search when precise knowledge of epipolar geometry is not known [88].

2.1.2.3 Correspondence Matching

Correspondence matching is the core of two-frame computational stereo. Given a pixel u_A in image I_A, the matching algorithm identifies the projection u_B of the same scene position into image I_B. The (u_A, u_B) matchings and camera parameters are later used by triangulation to compute 3D positions P. Matching and global optimization typically constitute the bulk of the complexity, runtime, and error in a two-frame stereo algorithm.

The analogous process in multi-frame stereo is to identify which *sets* of pixels show the same scene position. Multi-view algorithms may or may not perform explicit correspondence matching. They all perform equivalent operations, however, in that those operations can be restated in terms of identifying sets of matching pixels.

Sparse approaches estimate depth at a few pixels, which are called *features*, and which are selected to maximize the accuracy of the depth estimates. Feature selection can be driven by interest operators [89], corner detectors [109], scale-invariant features [78], oriented multi-scale Gaussian filters [62][91][100], or the popular "good features to track" approach of [108]. Further discussion and evaluations are given in [54] and [10].

Dense approaches estimate depth for most or all pixels. Some approaches use a layered *sparse-to-dense* strategy (e.g., Chapter 5).

Individual matchings (sparse or dense) are evaluated by a *local match quality function*, $Q(u_A, u_B; I_A, I_B)$, which measures the quality of a hypothetical matching in isolation. For rectified geometries (see page 23), Q can also be defined as $Q(x_A, y_A, d)$, with $u_A=(x_A, y_A)$ and d as disparity in pixels along the epipolar line. I_A and I_B are dropped for brevity.

Many different definitions of Q are used. A comparison is given in [50]. The simplest functions are the most common; simplicity is often important because Q is evaluated many times. The simplest common choice is *absolute difference* (AD),

$$AD(x,y,d) = \left| I_A(x,y) - I_B(x+d,y) \right|. \quad (7)$$

Alternatives include *squared difference* (SD) and non-linear operations [8][102] designed to mitigate the effects of large outlier errors. Complex Q functions are also used to relax assumptions, as discussed in Section 2.1.2.5 on page 22.

Q is typically aggregated over a local region. The most common choice is to aggregate AD or SD over small rectangular windows, forming the *sum of absolute differences* (SAD) and sum of squared differences (SSD), respectively. More complex techniques use multiple windows [48], shiftable windows [9], or adaptive windows [59] to improve results near object boundaries. These techniques compare favorably to competitors [103] and many can be computed efficiently. Cross-correlation of intensities across windows (XCORR) is also sometimes used for robust local aggregation of Q.

Most Q functions are defined only at integer disparities. Techniques like [110] can define simple Q functions at non-integer disparities, but these techniques are not common.

2.1.2.4 Global Optimization

Competitive techniques apply global optimization and it is the global optimization approach that differentiates them. Matching based solely on the best value of Q at each pixel is not effective because a purely local approach generates ambiguities and false matches. Figure 2-4 shows an example stereo pair and the disparity that maximizes Q at each pixel.

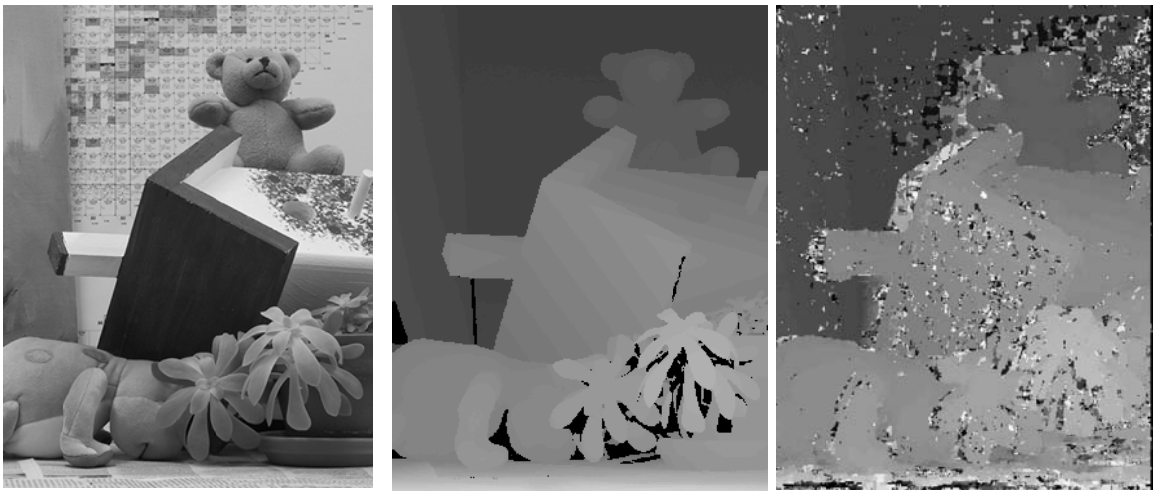


Figure 2-4: Example errors caused by local matching. (Left) One input image from an example stereo pair. (Center) ground truth disparity (pseudocolored). (Right) Disparity values maximizing *Q(x, y, d)* at each pixel for *Q* defined as cross-correlation (XCORR) over a local 5x5 window. Many errors are visible despite using even this relatively robust and complex local metric.

A survey of general and specialized optimization techniques in computational stereo is given in [103]. Many approaches formulate the optimization as an energy minimization where the objective function combines *Q* with regularization terms that capture assumptions or heuristics. Approaches based on Markov random field (MRF) models have significant interest (see 2.1.4.1). Multi-scale, coarse-to-fine, and hierarchical approaches are also common [57][68][99]. A detailed discussion of techniques directly relevant to my work is given in Section 2.1.4 below.

Global optimization almost always begins by exhaustively computing *Q* at all *x*, *y*, and *d*. This can be extremely expensive; processing NTSC video at 20 possible disparities (a coarse depth resolution for most scenes) would require 207 million *Q* computations per

second. Chapter 3 presents an efficient stochastic sampling strategy and demonstrates its benefits over exhaustive sampling.

2.1.2.5 Constraints and Assumptions

Matching and optimization algorithms use constraints and assumptions to make this under-constrained inverse computation tractable. Common constraints include:

- **Epipolar constraint:** Used by any feasible approach, and discussed in 2.1.2.2.
- **Lambertian scene model:** The Lambertian model assumes that the color and intensity of a surface is independent of the angle from which it is viewed. This model is violated by any specular reflection, translucency, and ambient lighting changes. Despite its inaccuracy, the model is very common; it is implicit in the definition of the AD and SAD metrics. Its limitations can be mitigated by feature-based matching ([31] and this dissertation's Chapter 5), or by Q functions using rank-and-census methods [123], correlation, or mutual information [49][116]. Other recent techniques address non-Lambertian scenes directly [56].
- **Uniqueness:** If all objects are opaque, at most one pixel in I_B can match a given pixel in I_A and vice versa [81].
- **Piecewise continuity:** The world is comprised of solid objects of nonzero size. Depth and disparity vary smoothly on their surfaces. Discontinuities occur only at object boundaries [81]. (This also generates $1/f$ statistics in natural images [35].)

- **Ordering:** For pixels on the same epipolar line, the ordering of corresponding pixels is the same in both images [4][93]. This constraint does not always hold; thin vertical foreground objects can cause violations.
- **Figural continuity:** Depth and disparity vary smoothly along intensity edges because depth varies smoothly along the edges of objects' surfaces and their surface markings [84].
- **Edge connectivity:** Connected edge points in one image must match to connected edge points in the other image [85]. Like the ordering constraint, it is not absolute. Unlike the ordering constraint, it fails in predictable ways.
- **Continuity of matching likelihood:** The certainty of a correspondence match being correct is continuous [65]; high-certainty matches increase the likelihood of neighboring matches.

2.1.2.6 Rectification

In a *standard stereo geometry* the left and right epipoles E_A and E_B are at infinite distance in the right and left directions (i.e., E_A=(1,0,0,0)' and E_B=(-1,0,0,0)'). Epipolar lines align with horizontal scan lines, and vertical disparity is identically zero. This simplifies computation and can improve numeric conditioning. Standard stereo geometry is only physically achievable using two cameras mounted on a common chassis. Single-camera approaches must achieve it by rectification, when it is possible at all.

A variety of rectification strategies exist. Two-frame projective rectification applies constrained projective transforms to both images to establish a standard geometry with minimal distortion at the principal camera points [44]. This allows the use of algorithms that assume a standard geometry (often enforced by a custom binocular stereo rig) on inputs such as monocular video.

Rectification can also be used to eliminate only the rotations between two cameras. Then, the direction of disparity is independent of depth. The epipolar line for a left pixel $I_L(r_i, c_i)$ will always pass through the right pixel $I_R(r_i, c_i)$. An object at infinite range would project to (r_i, c_i) in both images. This strategy is used to improve efficiency in Chapter 3.

2.1.2.7 Relationship to Other Methods

Computational stereo via disparity analysis is not the only way to estimate structure from passive imagery. *Shape from focus* estimates depth by varying a camera's focal length and identifying which parts of the image are sharply in focus at each setting [69]. The underlying theories of shape from focus and computational stereo were unified in [105]. *Shape from shading* analyzes intensity patterns in a single image to infer structure gradients [52]. *Occlusion reasoning* provides partial orderings of object ranges based on which are nearer to the camera, and can be used in combination with disparity analysis [61]. *Semantic information* can also be exploited; if the size and geometry of a

school bus is known, then the size and shape of its projection in an image can be used to estimate its distance and pose.

There is strong evidence that human vision relies strongly on cues other than disparity for the perception of depth [70]. Binocular disparity is perceived directly by the early human visual system [40]. However, the small stereo baseline of the human visual system limits its contribution to near-field perception. Distant structure perception incorporates many other depth cues, such as occlusion, eye focus, and semantic knowledge [113].

2.1.3 Structure Representations

Structure representations can be categorized as *view-centric* or *scene-centric*.

View-centric representations are interpreted with respect to one or more input viewpoints, and are most common in two-frame stereo. The *disparity image* (see Figure 2-4 on page 21) stores a disparity at each pixel for a pair of viewpoints. *Depth images* store a depth at each pixel. Given camera calibrations, depth and disparity images are equivalent. Most approaches generate uni-valued estimates; multi-valued estimates may be used to represent transparent, translucent, or occluded surfaces [31].

Scene-centric representations are defined in a *world coordinate system* or *scene coordinate system* independent of an input viewpoint (or relative to an arbitrarily chosen input viewpoint). The simplest representation is the *point cloud*, a set of unassociated points in 3D space. Point clouds are common in sparse approaches, but they can become

unwieldy for dense approaches. *Voxel models* generalize pixels to volume elements to store a Boolean occupancy value (and possibly other data) at each 3D location in the scene. Sparse, dense, or multi-scale voxel models (e.g., octrees) provide more scalable and intuitive operation than point clouds for most multi-view approaches. Voxel models are the most common representation choice for multi-view approaches.

Surface representations such as triangle meshes, splines, and others described in [122], provide the most abstract and scalable representation. They are not often used during modeling because they impose substantial overhead on correspondence matching and global optimization. Exceptions include [51] and [76], which integrate surface modeling with correspondence matching as a form of global optimization. View-centric approaches and those using point clouds or voxel models may convert to surface representations for later applications (e.g., [86]).

2.1.4 Relevant Approaches

A substantial body of related work exists. Relevant surveys of sparse stereo approaches can be found in [6] and [31], dense stereo in [14] and [103], and multi-view stereo in [106]. A few categories of techniques are of particular relevance, and are discussed in more detail below.

2.1.4.1 Stochastic Approaches

The literature on "stochastic" approaches exhibits an interesting dichotomy. Stochastic *models* are used widely to formulate the global optimization problem. In contrast, the use of stochastic *algorithms* is very rare.

Many recent two-frame approaches leverage the piecewise-smoothness constraint by modeling the disparity field as a Markov random field (MRF) [111]. They use network flow, graph cuts [66][12], or other deterministic methods to generate an optimal estimate of the MRF state given the inputs and assumptions. Recent optimizations [12] have generated new interest in MRF-based approaches. Stochastic models are also represented in recent multi-view approaches as well. One example is the work in [15] and [16], which uses a 3D volumetric MRF model to fuse depth maps computed from multiple pairings of views.

The few examples of stochastic algorithms center on *simulated annealing* and its variations. The Metropolis algorithm [87] stochastically generates a sequence of systems states whose distribution converges to that of a physical system with energy E in equilibrium with a heat source at temperature T. Simulated annealing [64] optimizes an arbitrary energy function by executing the Metropolis algorithm to convergence with decreasing T. If T approaches absolute zero slowly, the system converges provably to its minimum-energy state [38]. Microcanonical annealing extends simulated annealing to improve convergence speed by changing the physical model to a closed system [26][27]

using a *demon* (or slack) *variable*.

The seminal stochastic computational stereo algorithm is [7], which extends microcanonical annealing to perform two-frame correspondence matching. It uses a multi-scale formulation, applying microcanonical annealing at each scale. A lattice of demon variables (one per pixel) is periodically shuffled to disperse energy spatially and speed convergence further. This approach is used in Chapter 3 and is called "MCA" hereafter.

2.1.4.2 Iterative Refinement Approaches

Techniques based on iterative refinement apply repeated local or non-local operations to converge on a solution. These include *relaxation*, *diffusion*, and *cooperative* techniques, which tend to share many concepts. They do not typically define an explicit global objective function [103]. Instead, nonlinear or anisotropic (but deterministic) local operations are chosen to effect the desired global behavior.

Cooperative approaches are some of the earliest credible approaches [80][101]. They were inspired by models of early human vision that suggest cooperative processing (e.g., [30]). They can require long runtimes and may not perform well at object boundaries [103], but they generally give accurate results.

Zitnick and Kanade's cooperative approach (hereafter, ZK) [124] is used in the evaluations in Chapter 3. Figure 2-5 illustrates ZK operation on a single scan-line row,

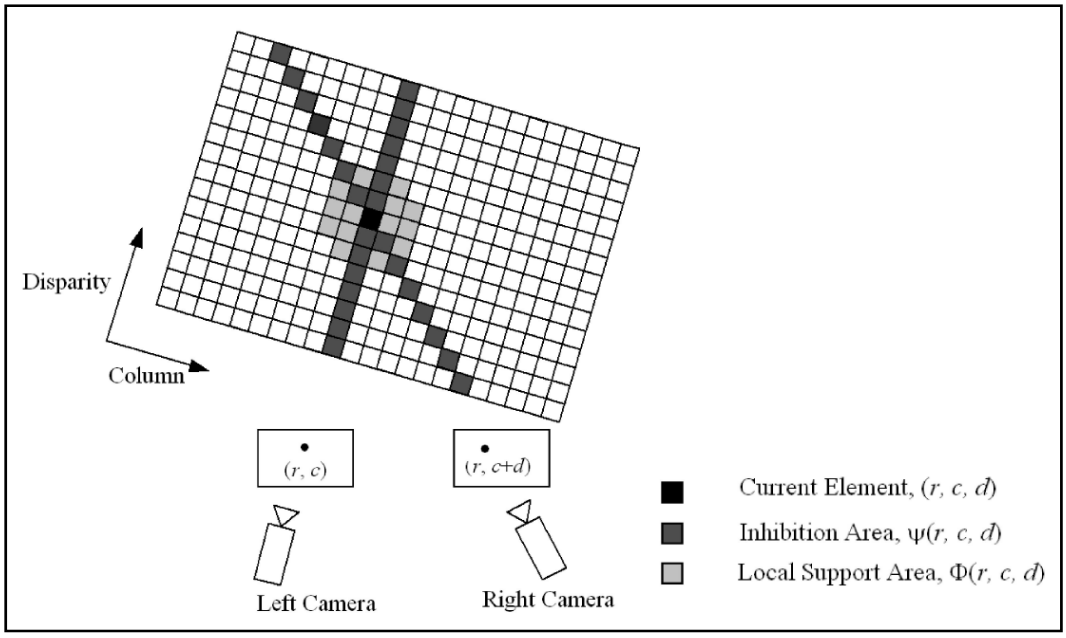


Figure 2-5: The Zitnick and Kanade (ZK) cooperative approach [124] for a single epipolar line in the standard geometry. Each possible correspondence *(r, c, d)* supports correspondences in a local region *Φ(r, c, d)* and inhibits correspondences in a non-local region *Ψ(r, c, d)*.

corresponding to a single epipolar line in the standard geometry. *Q* is computed exhaustively and used to initialize a match likelihood estimate *L(x, y, d)*. On each iteration, *support* (*Φ*) and *inhibition* (*Ψ*) are computed and aggregated from *L* according to the regions shown in Figure 2-5. Support models the piecewise continuity constraint and is a local effect. Inhibition models the uniqueness constraint and is a non-local effect. Support, inhibition, and *Q* are used to update *L(x, y, d)* for the next iteration.

After iteration, disparity estimates are selected as $D(x,y) = \arg\max_d \{L(x,y,d)\}$ and occlusions are declared for pixels with low *L(x, y, d)* for all possible *d*. A standard

stereo geometry is assumed and required for efficient operation.

2.1.4.3 Space Carving and Extensions

The Space Carving (SC) multi-view stereo approach [71] sequentially tests each voxel to determine if the input imagery and known-unoccupied voxels are consistent with its occupancy. If not, the voxel is removed and the process repeats. SC has inspired much interest and has been extended to address camera calibration error [72], characteristic holes in its reconstructions [13][121], and real-time operation [120]. The accuracy of the SC algorithm is limited by aliasing. In the words of its creators, its accuracy can be increased "only up to the point where image discretization effects (i.e., finite pixel size) become a significant source of error" [71].

2.1.4.4 Approaches Based on the Hough Transform

Other relevant algorithms are based on techniques or concepts from the Hough transform. These are discussed in Section 2.2.4 after a detailed discussion of the transform itself.

2.2 The Hough Transform

The Hough transform (HT) was designed as a technique for detecting lines in 2D images [53]. It has since been applied to general shape detection, computational stereo, and many other problems. Many variations of the transform have been developed. It is one of the most basic and widespread techniques in computer vision.

2.2.1 Standard Hough Transform

In the standard Hough transform (SHT), lines are typically parameterized as (ρ, θ) following [32], with

$$\rho = x \cos \theta + y \sin \theta , \tag{8}$$

where ρ is distance to the origin and θ is angle relative to the positive x axis. Every line maps to a (ρ, θ) pair, and every pixel maps to a sinusoidal trajectory in (ρ, θ). For every edge pixel (x, y) and every quantized $\hat{\theta}$, the SHT computes the ρ with which $(x, y, \hat{\theta})$ are consistent. Votes from all pixels are summed in an accumulator $A(\hat{\rho}, \hat{\theta})$ with $\hat{\rho}$ and $\hat{\theta}$ quantized at increments of Δ_ρ and Δ_θ, respectively. Peaks appear in A at line parameterizations supported by many pixels. Lines are detected by thresholding A. Commonly, only local peaks in A are returned.

2.2.2 Extensions and Variations

Many variations exist for accumulation and peak detection. A relevant survey is given in [98].

Gerig and Klein introduce a second accumulator $B(\hat{\rho}, \hat{\theta})$ in [39], calculated after A is complete. Each pixel (x, y) votes in B for only the θ^* with the greatest value in A along its trajectory in (ρ, θ). This improves line detection performance [98].

Palmer, Kittler, and Petrou introduce a refined voting kernel K_p in [94]. Votes in $A(\hat{\rho}, \hat{\theta})$ are weighted by the distance from the cell's quantized $\hat{\rho}$ to the ideal ρ computed from x, y, and $\hat{\theta}$, according to

$$K_P(\rho - \hat{\rho}) = \begin{Bmatrix} 1 - 2\frac{(\rho - \hat{\rho})^2}{w^2} + \frac{(\rho - \hat{\rho})^4}{w^4} & for\ |\rho - \hat{\rho}| < w \\ 0 & otherwise \end{Bmatrix}. \quad (9)$$

K_p transitions smoothly from 1.0 for $|\hat{\rho} - \rho| = 0$ to 0.0 for $|\hat{\rho} - \rho| \geq w$, where w is a free parameter that defines the half-width of the kernel in bins. The SHT kernel, for comparison, is

$$K_{SHT}(\rho - \hat{\rho}) = \begin{Bmatrix} 1 & for\ |\rho - \hat{\rho}| < 0.5 \\ 0 & otherwise \end{Bmatrix}. \quad (10)$$

Re-accumulation into B also uses K_p. Pixels are grouped into segments based on their cell in B and a maximum gap in pixels within a segment. Candidates with too few pixels are dropped. Each segment's (ρ, θ) estimate is refined using a custom three-stage cubic optimization tailored to curved sinusoidal ridges found in Hough space. The optimization exploits the fact that K_P is defined and differentiable for all (ρ, θ). Improved accuracy is demonstrated and attributed to superior spectral localization of K_P over K_{SHT} in Hough space.

2.2.3 Hough-Domain Aliasing

It is well known that aliasing hinders Hough domain line detection. Ideal Hough

domain signals are non-bandlimited, and the necessary quantization into $\hat{\rho}$ and $\hat{\theta}$ results in aliasing [67]. Most HT anti-aliasing techniques build on traditional signal processing strategies [73][92]. These are problematic in applications such as multi-view stereo because ideal HT-domain trajectories vary widely based on camera viewpoint and other factors. The inability to analytically derive the ideal HT-domain trajectory or compute it offline make existing anti-aliasing techniques expensive to the point of being infeasible.

Further, aliasing is unavoidable in HT-based stereo because is impossible to simultaneously avoid under-sampling or over-sampling Hough domain representations of non-homogeneous objects [73], which will be found in any stereo application.

2.2.4 Hough Transform Applied to Computational Stereo

The Hough transform has been previously applied to multi-view stereo. It is typically used to generate wireframe models from edge-detected input images. Wireframes may then be passed to other techniques to build dense models. Most approaches stop short of generating dense models with Hough transform concepts.

The Hough transform was applied to multi-view stereo in [23] and [24], with a focus on reconstructing scenes comprised mainly of building roofs. The approaches in [41] and [63] use it with a re-accumulation analogous to [39] but do not generate dense models. In [20] we improved wireframe modeling with DRT but also did not generate dense models.

In [112], dense models are formed from sparse models similar to those that are

generated by [41] and [63]. This is done by solving the dual problem of identifying unoccupied voxels. However, [112] uses techniques other than the Hough transform to generate the sparse models.

The Hough transform extensions and anti-aliasing techniques have generally not been applied to stereo reconstruction. The re-accumulation of [39], used in [41] and [63], is the primary exception.

2.3 Ray Tracing and Distributed Ray Tracing

Ray tracing is a computer graphics technique that renders images by simulating the visual contributions of objects in a scene to each image pixel along a ray from the camera origin, through the pixel, and into the scene [37]. Figure 2-6 illustrates the conceptual operation of a ray tracing algorithm.

Ray tracing suffers from aliasing. It attempts to represent non-bandlimited phenomena in the scenes (physical objects) on a uniformly sampled digitized medium (pixels). It uses discrete algorithms to process finite numbers of pixels, rays, voxels, and surface meshes. Further, scene representations are often textured with other sampled inputs like photographs or video. All of these factors contribute to aliasing in ray tracing.

Distributed ray tracing (DRT) [25] is an extension of ray tracing designed to mitigate aliasing in rendered images. DRT casts multiple rays through each pixel's cross-range extents, each with minor random perturbations in direction. Pixel appearance is the averaged contributions of the rays. This reduces aliasing by improving the modeling of

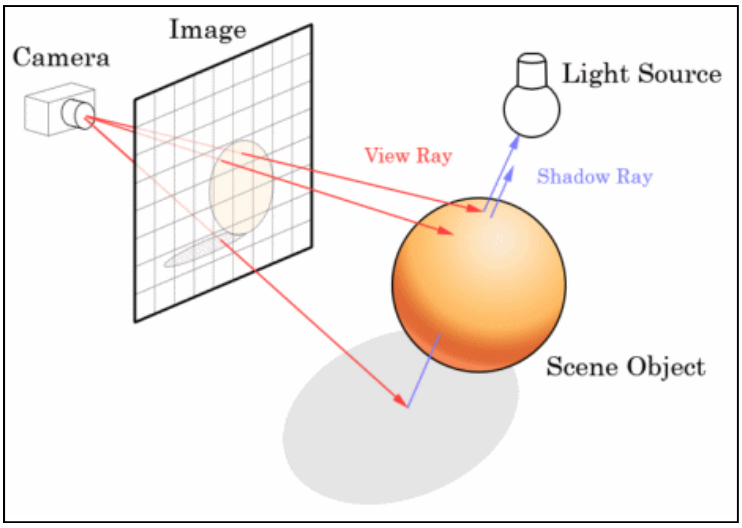


Figure 2-6: Conceptual illustration of computer graphics ray tracing (without distributed ray tracing) from [47]. Rays are cast from the origin through each pixel and into the scene being rendered. The appearance of each pixel is dictated by the intersections, reflections, refractions, and light sources encountered by the single ray (and its branches or divisions) that is cast for the pixel.

pixels, lenses, lights, and objects with non-zero spatial extents. DRT can be used to improve the rendering of gloss, translucency, shadow, motion blur, depth of field, and other soft visual effects. Figure 2-7 shows sample comparisons of ray tracing results with and without DRT.

DRT also reduces moiré patterns and other interference patterns caused by aliasing. Applied most broadly, DRT is a technique for reducing the effects of aliasing in digital computational geometry algorithms.

The randomness of the ray perturbations is critical to DRT's effectiveness. The sampled phenomena are non-bandlimited. Simply super-sampling rays on a regular sub-

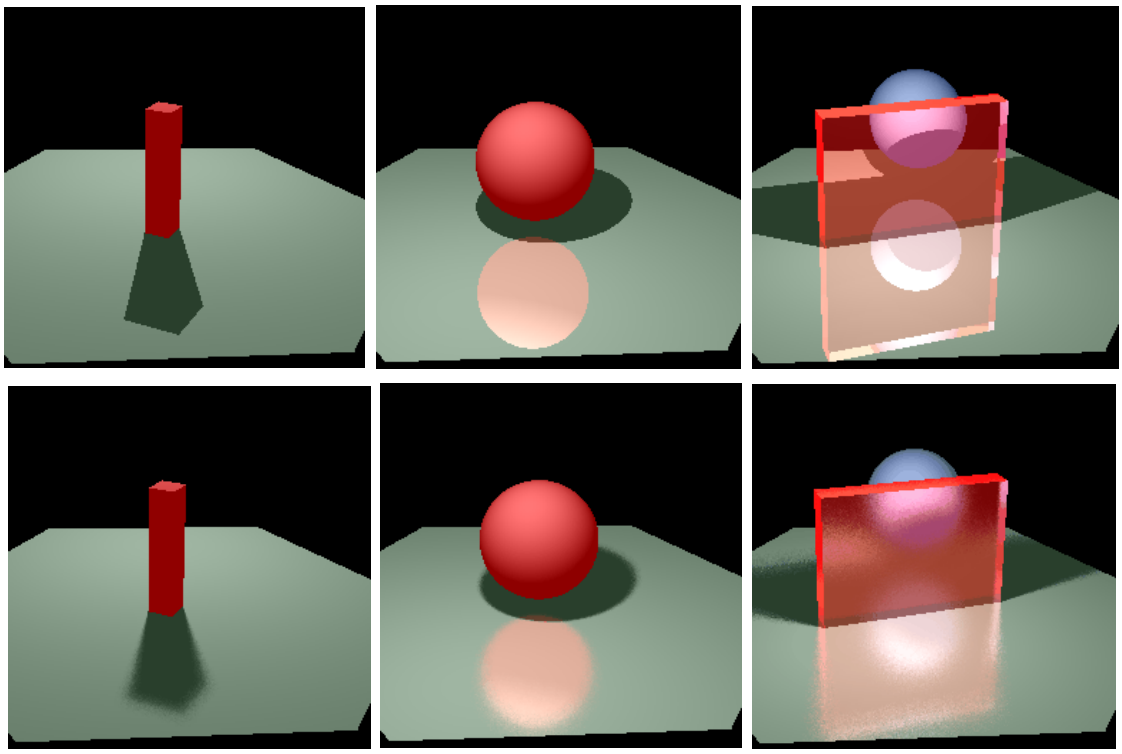


Figure 2-7: Comparison of results from standard ray tracing (top) and distributed ray tracing (DRT) (bottom), from [82]. DRT reduces aliasing in the representation of physical phenomena to generate more realistic renderings of soft visual effects including shadows, gloss, and translucency.

pixel grid is insufficient; no regular sampling rate is high enough to eliminate aliasing. This is seen most clearly with moiré patterns, which can remain strong with super-sampling but are nearly eliminated by stochastic sampling.

2.4 Limitations of the Existing State of the Art

After 40 years of active research, the state of the art in computational stereo does not provide a general and robust solution. A number of practical problems remain, particularly for long-range or wide-area modeling.

High-speed solutions typically still require specialized processing hardware and/or multi-camera rigs that enforce a standard stereo geometry (e.g., [60]). This prevents long-range modeling because establishing a sufficient stereo baseline with a rig is impractical.

Exhaustive sampling of Q is also a problem. The number of samples required is significant, causes high runtimes. Storing the samples causes high memory requirements. As discussed in Chapter 3, the number of samples for an exhaustive approach is dependent on both camera and scene geometry, and scales poorly with input resolution. These problems are exacerbated by the use of complex Q functions used to mitigate non-Lambertian surfaces or lighting changes. Exhaustive sampling is precluded altogether for Q functions defined on non-integer disparities. The few existing stochastic sampling approaches are not competitive.

Aliasing becomes an issue both in input imagery and voxel models for long-range or wide-area modeling. Aliasing limits performance broadly across multi-view approaches, including the popular Space Carving family of algorithms. Few anti-aliasing techniques exist in stereo methods, and those that do are highly specialized.

It is known that aliasing limits Hough transform (HT) performance in traditional line detection [67]. As shown in Chapter 4, this becomes an acute problem at reduced image resolutions and/or reduced accumulator resolutions. There is no way to define an optimal sampling strategy when the scene contains heterogeneous elements. Accumulators and thus memory usage must remain large to achieve accurate results.

The aliasing-induced limitations of the HT in turn limit the capabilities of HT-based stereo approaches. Heterogeneous objects cannot be avoided in stereo modeling. The body of existing HT anti-aliasing techniques do not apply well to HT-based stereo because closed-form analytic solutions cannot be easily computed or estimated.

Efficiency, accuracy, and robustness therefore all remain as practical limitations of the state of the art in computational stereo – particularly for long-range or wide-area modeling.

There is also an academic limitation in the state of the art. Stochastic algorithms (as opposed to stochastic models) are under-represented in the computational stereo literature. A stronger pursuit of stochastic approaches may help eliminate some of the practical limitations as well.

I now present and analyze two novel stochastic approaches. The first reduces the number of Q evaluations required by cooperative algorithms (Chapter 3). The second reduces aliasing in the Hough transform (Chapter 4) in a way that can be transferred to improve Hough-transform-based dense multi-view stereo modeling (Chapter 5).

Chapter 3.

Efficient Stochastic Sampling of Match Quality Functions

This chapter presents the Quality Efficient Stochastic Sampling (QUESS) computational stereo approach, which is patent-pending [18] and also described in [17] and [19]. Unlike most existing approaches, QUESS uses a non-exhaustive stochastic sampling of the local correspondence match quality function. This allows accurate depth estimation from calibrated stereo pairs or calibrated video while requiring fewer evaluations of the match quality function than exhaustive sampling. Non-exhaustive sampling facilitates the use of complex quality metrics, as well as quality metrics that take unique values at non-integer disparities, for which exhaustive sampling is impossible.

QUESS is a stochastic cooperative approach. Depth estimates are iteratively refined by stochastically perturbing the estimates, sampling match quality, and reweighting and aggregating the perturbations. This gains significant efficiencies when applied to video, where initial estimates are seeded using information from the previous stereo pair. Seeding significantly reduces the number of search iterations required to

process each stereo pair. QUESS is shown to outperform two competing approaches, and also to have more attractive memory usage and scaling properties than alternatives based on exhaustive sampling.

3.1 As Applied to Calibrated Stereo Pairs

Here I discuss the representation of the estimated quantities, the core stochastic cooperative search and the local and aggregated *influences* on which it is based.

3.1.1 Definitions and Representations

Consider two 2D images $I_A(i, j)$ and $I_B(i, j)$, with image I_A defined as the reference image. The images are assumed to be in a standard stereo geometry with known (or assumed) stereo baseline and intrinsic parameters. A scalar floating-point depth $\hat{D}(i, j)$ can be estimated at each pixel based on the *local match quality function* $Q(i, j, \hat{d}; I_A, I_B)$, which varies with depth estimate $\hat{d}$. Depth estimates can be converted to and from equivalent disparities as required (e.g., to compute Q or for evaluation in disparity-based frameworks).

Direct depth estimation contrasts with estimating integer disparity values and post-processing to recover sub-pixel disparities or depths. Estimating floating-point depth directly is preferable in situations with large depth ranges and significant spatial variations. It also avoids quantization and supports match quality metrics defined on continuous domains.

3.1.2 Stochastic Cooperative Search

An overview of QUESS is given in Table 3-1. Depth estimates are iteratively refined with a stochastic cooperative search. QUESS perturbs the depth estimates, reweights perturbations using *local influence* computed from their effects on Q, and adds *aggregated influence* to the estimates to incrementally move them towards a better solution. The search is guided by a schedule analogous to those used in simulated annealing.

Depth estimates at each pixel are initialized from the previous search stage, previous frame, or from a uniform distribution over the bounds $\hat{D}_{\min}(i,j)$ and $\hat{D}_{\max}(i,j)$. Depth bounds vary pixel by pixel and are defined using any prior knowledge about the scene (including disparity bounds).

In each iteration $n=1..N$, random noise $\Delta_{D_n}(i,j)$ is added to the previous depth estimate $\hat{D}_{n-1}(i,j)$ to form a candidate depth estimate $\tilde{D}_n(i,j)=\hat{D}_{n-1}(i,j)+\Delta_{D_n}(i,j)$. Q is evaluated at the candidate to compute a new sample, $\tilde{q}_n(i,j)=Q(i,j,\tilde{D}_n(i,j);I_A,I_B)$.

The noise $\Delta_{D_n}(i,j)$ added to each depth estimate is uniformly distributed, subject to two constraints. The first constraint is a maximum depth perturbation magnitude $\delta_{\max} \in [0,1]$ relative to the pixel-specific depth bounds,

$$\frac{\left\|\Delta_{D_n}(i,j)\right\|}{\hat{D}_{\max}(i,j)-\hat{D}_{\min}(i,j)} \le \delta_{\max}. \tag{11}$$

For each search stage

1. Initialize depth estimates $\hat{D}_0(i,j)$
2. For each iteration $n=1..N$
   a. Add noise to get perturbed estimate $\tilde{D}_n(i,j)$
   b. Sample Q at perturbed depth estimate
   c. Compute local influence $J_n^*(i,j)$ from Q samples
   d. Filter $J_n^*(i,j)$ to get aggregated influence $J_n(i,j)$
   e. $\hat{D}_n(i,j) = \hat{D}_{n-1}(i,j) + J_n(i,j)$
   f. Smooth depth estimates $\hat{D}_n(i,j)$

Table 3-1: Overview of the QUESS approach.

By gradually reducing $\delta_{\max}$, the samples of Q in later iterations are forced to be closer to current estimates. The second constraint limits $\Delta_{D_n}(i,j)$ so the perturbed estimate $\tilde{D}_n(i,j)$ remains within bounds,

$$\Delta_{D_n}(i,j) \in \left[\hat{D}_{\min}(i,j) - \hat{D}_{n-1}(i,j),\ \hat{D}_{\max}(i,j) - \hat{D}_{n-1}(i,j)\right]. \tag{12}$$

The constraint in (12) is necessary because even if $\delta_{\max}$ is small, (11) may not prevent $\tilde{D}_n(i,j)$ from falling outside the allowable range if $\hat{D}_{n-1}(i,j)$ is close to $\hat{D}_{\min}(i,j)$

or $\hat{D}_{\max}(i,j)$. The constraints are introduced before sampling, instead of sampling and then clipping, to avoid biases caused by over-sampling at the extremes of the depth range.

A *local influence* $J_n^*(i,j)$ is computed at each pixel by preferentially weighting depth perturbations that improve *Q*. Local influence is aggregated over a support region *W* to produce *aggregated influence* $J_n(i,j)$. Aggregated influence is added to the last iteration's depth estimate to incrementally improve the estimates, $\hat{D}_n(i,j) = \hat{D}_{n-1}(i,j) + J_n(i,j)$. Details of local and aggregated influence appear in Section 3.1.3.

Finally, depth estimates are smoothed at the end of each iteration, modeling the piecewise continuity constraint [81], and helping the effects of the influence function to propagate.

Search parameters vary by stages. The search schedule defines the maximum depth perturbation magnitude $\delta_{\max}$, aggregation neighborhood *W*, and number of iterations *N* for each stage. *W* is a square region with side length specified as a fraction of the average of the row and column resolutions, μ_{res}, to insulate search parameters from changes in image resolution. *W* and $\delta_{\max}$ are large in early stages to capture gross scene structure. They both shrink in later stages to capture detail and force convergence of the estimates. This is analogous to the cooling process of simulated annealing or the organization process of self-organizing maps.

The QUESS approach is heuristic. While good results are achieved and convergence is enforced by the search schedule, the estimates are not guaranteed to be optimal.

3.1.3 Match Quality, Local Influence, and Aggregated Influence

Local influence is derived from the stochastic samples of Q. Many alternative Q metrics were explored, including weighted sum of absolute differences, squared error, and normalized cross-correlation (hereafter, XCORR). Although QUESS enables the use of more complex Q, excellent results are achieved even with simple definitions. XCORR is used since it provides superior performance in the simulations owing to the divisive normalization.

Figure 3-1 shows example depth perturbations $\Delta_{D_n}(i,j)$ (expressed in grayscale), resulting changes in match quality $\Delta_{q_n}(i,j) = \tilde{q}_n(i,j) - q_{n-1}(i,j)$, local influence $J_n^*(i,j)$, and aggregated influence $J_n(i,j)$.

Local influence selectively weights depth perturbations that improve the depth estimate $\hat{D}_n(i,j)$ as inferred by improvements in Q. Random depth perturbations result in "noisy" $\tilde{q}_n(i,j)$ and $\Delta_{q_n}(i,j)$. Some perturbations increase depth while others decrease depth. Some increase quality while many decrease quality. Local influence should be positive where perturbations that increase depth also increase quality, and negative where perturbations that decrease depth increase match quality. Where perturbations *decrease*

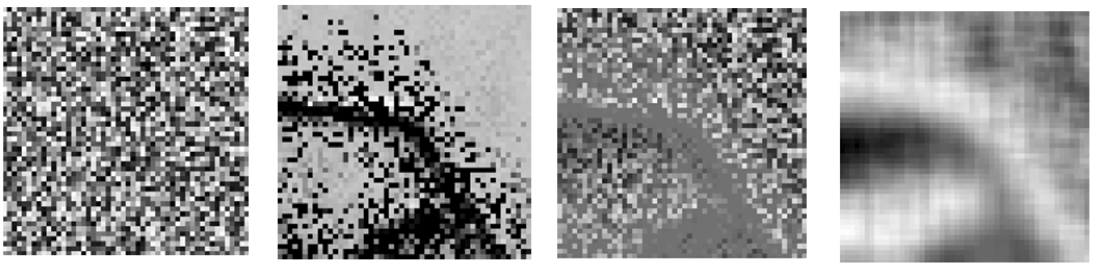


Figure 3-1: Influence formulation and aggregation. (a) Depth perturbations $\Delta_{D_n}(i,j)$, (b) changes in Q, $\Delta_{q_n}(i,j)$, (c) local influence $J_n^*(i,j)$, (d) aggregated influence $J_n(i,j)$.

match quality, local influence should be either zero or oriented away from the perturbation.

Results are improved by categorizing pixels as either *contributing* or *non-contributing*. For contributing pixels, $J_n^*(i,j)$ is the depth perturbation realizing the maximum historical sample of Q in the current search stage. For non-contributing pixels, $J_n^*(i,j) = 0$.

A pixel is contributing if it passes two tests:

1. A minimum on the standard deviation of local pixels.
2. A minimum on the range of Q samples at that pixel.

These tests inhibit local influence from pixels where Q is unreliable due to insufficient texture or other features that may cause Q values to be similar (e.g., a dominant gradient along the epipolar direction).

Local influence is defined as

$$J_n^*(i,j) = \begin{cases} d_n^*(i,j) - \hat{D}_{n-1}(i,j) & , \; if \;\; M(i,j) = true \\ 0 & , \; if \;\; M(i,j) = false \end{cases} \tag{13}$$

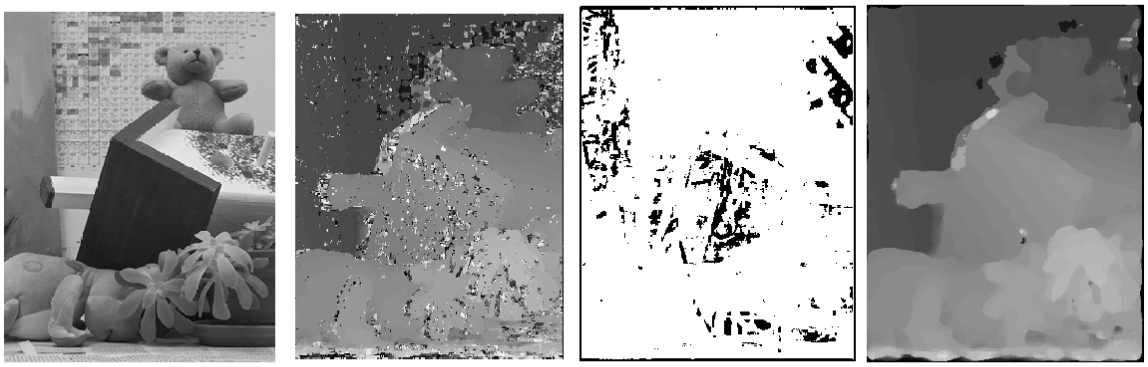
$$d_n^*(i,j) = \hat{D}_k(i,j) \text{ such that } \tilde{q}_k(i,j) = \max \tilde{q}_{1..n}(i,j) \tag{14}$$

$$M_n(i,j) = \left[std_{9x9}(i,j;I_A) > \alpha \right] \cap \left\{ \left[\max \tilde{q}(i,j) - \min \tilde{q}(i,j) \right] > \beta \right\}, \tag{15}$$

where $std_{9x9}(i,j;I_A)$ is the standard deviation in a local 9x9 square region. The mask $M_n(i,j)$ is defined relative to all historical Q samples, but the local influence of contributing pixels is defined relative to Q samples in the current search stage only (via $d_n^*(i,j)$). This exploits all knowledge of Q to identify reliable samples, but still forces the estimates to converge and capture detail in later search stages. Computing local influence requires maintaining only $d_n^*(i,j)$, $M_n(i,j)$, and two minima/maxima of Q.

Like local influence, there is flexibility in the definition of aggregated influence. It should capture consistent trends in local influence that reflect scene structure, reject spurious local influences caused by artifacts, and tend towards zero when an acceptable solution is reached.

Averaging $J_n^*(i,j)$ over W is efficient and can be effective on some scenes. However, this enforces smoothness where piecewise-smoothness is instead desired. Anisotropic smoothing prevents loss of detail along boundaries [1][95], with promising accuracy but at a significant cost. Other robust aggregation approaches, including order-statistic filtering (e.g., [11], [77]) or bilinear filtering can be applied.



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

Figure 3-2: Selective median filtering. (a) Reference image, (b) disparity maximizing XCORR, (c) contributing pixel mask $M_n(i,j)$, (d) aggregated influence $J_n(i,j)$.

A selective median filter was particularly effective. Median filters that combine partial histograms for each column are *O(n)* in pixel number *n* and *O(1)* in filter kernel size when applied to integer images [96]. Efficiency is improved by incrementally updating bin indices [55]. I implemented two novel extensions. The first only includes a value in the histograms if it passes a mask. The second applies the filter to floating point images by returning the histogram bin center containing the median – an approximation with bounded error to the true median.

Figure 3-2 illustrates the effects of the filter on a representative disparity image. Depicted are the reference image, the disparity at which XCORR is maximized, an example contribution mask, and the results of selective median filtering.

3.2 As Applied to Calibrated Video

Modifications of the search process combined with pre- and post-processing provide additional efficiencies when QUESS is applied to calibrated video input.

3.2.1 Pre-Processing

The pre-processing necessary to operate on calibrated aerial video comprises frame pairing, rectification, and input masking.

When applied to calibrated video, QUESS generates depth estimates for each frame in the video stream. The most recent frame in the stream is defined as the reference frame I_A. Following [115], stereo pairs are formed by selecting a non-adjacent frame I_B (see Figure 3-3) to maintain a target ratio τ_0 between the stereo baseline T and the minimum depth to the scene. To simplify computation, T is defined simply as the Euclidean distance between the camera origins. In addition to providing a stereo baseline sufficient to generate accurate depths (which pairing adjacent frames would not do), this provides robustness to changes in platform speed and direction, making it possible to tune other parameters to a more consistent geometry.

QUESS gains advantages from rectifying, but rectification is not strictly necessary. Image I_B is re-projected to a plane that is parallel to (but not necessarily coplanar with) the image plane of I_A, thereby creating I_{BR}. This can be described as a partial planar rectification. After rectification, scene elements at infinite depth exhibit zero disparity, and unit vectors in the epipolar direction at each pixel in I_A are computed

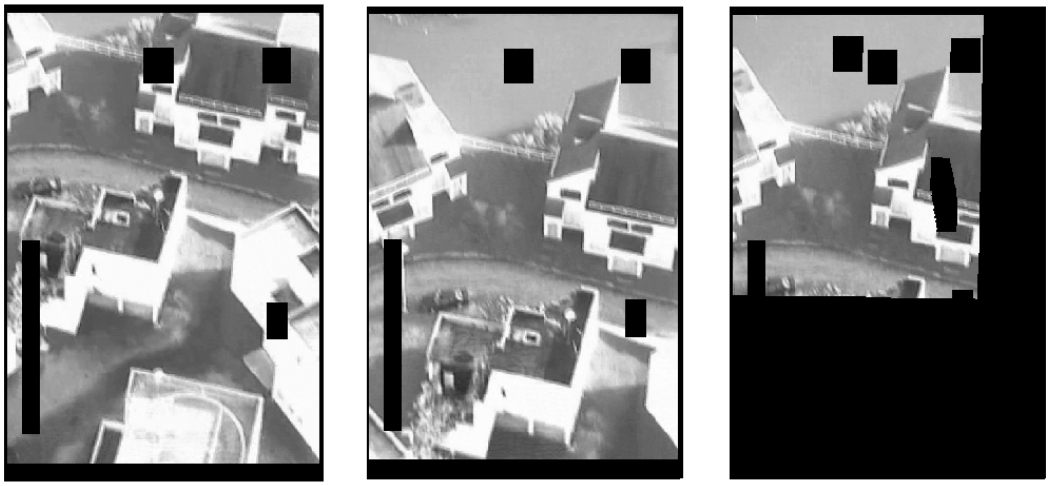


Figure 3-3: (a) Reference image I_A masked for artifacts, (b) paired image I_B masked for artifacts, (c) rectified paired image I_{BR} masked for artifacts, rectification, and scene assumptions.

and stored. Epipolar lines are not collinear or parallel as in a standard stereo geometry. However, the rectification allows quick conversion between estimated depth and equivalent disparity for computing match quality. This process also enforces correspondences to lie on epipolar lines.

Following frame pairing and rectification, a mask is computed to identify pixels satisfying various constraints on correspondences

1. That I_{BR} has the same domain as I_B,
2. That corresponding pixels lie within the images,
3. That pixels are not coincident with known artifacts, and

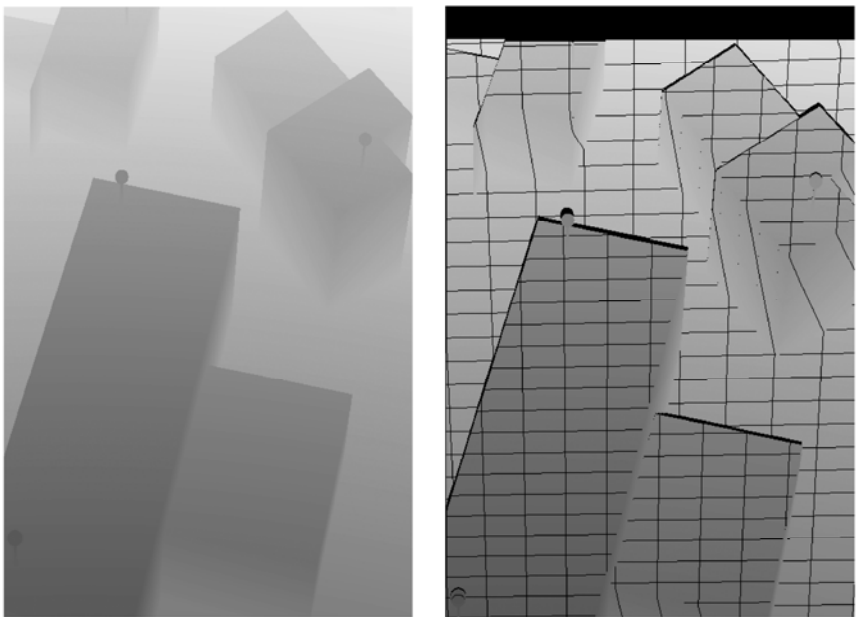
4. That depth estimates respect known scene boundaries.

Pixels failing these constraints are black in Figure 3-3(c).

3.2.2 Modifications to Stochastic Cooperative Search

QUESS stochastic search is modified in three ways. First, estimates and intermediate values are initialized using results from the previous frame pair when available. This lets estimates converge over multiple frames – existing estimates are refined instead of generating entirely new estimates. As shown in Figure 3-4, depth estimates can be seeded aggressively since camera position and orientation differ little between adjacent frames. Estimates from the last frame are re-projected to the new reference frame and adjusted for changes in camera origin, using Z-buffering [37] to address occlusion. A similar application of Z-buffering is used to initialize key quantities such as $d_n^*(i,j)$, $M_n(i,j)$, and statistics of Q, letting the search leverage results from the previous frame pair. Any small gaps can be filled by nearest-neighbor interpolation.

Second, the search schedule is modified to exploit the redundancy between frame pairs. The search schedules still require successive stages to decrease the perturbation magnitudes and neighborhood sizes, thus capturing both gross scene structure and detail. A unique schedule is used for the first frame that performs more iterations and emphasizes larger perturbations. All subsequent frames use schedules with significantly fewer iterations. These emphasize smaller neighborhoods and smaller depth perturbation in order to refine the existing solution and capture detail, although they must also capture



(a) (b)

Figure 3-4: (a) Depths (pseudocolored) in frame I_A, (b) Depths re-projected to frame I_{A+10} by Z-buffering.

larger structures in newly-visible regions. Combined with aggressive seeding, the modified search schedule lets estimates converge over many frame pairs with very few iterations (and Q evaluations) per pair.

Third, conservative assumptions constrain the depth estimates at each pixel. Application-specific assumptions can considerably improve speed and accuracy. For example, aerial modeling benefits more from bounds on elevation than bounds on disparity, and those bounds are easier to estimate reliably (e.g., from existing low-resolution elevation data).

3.2.3 Post-Processing

After computing depth estimates with respect to I_A, intrinsic and extrinsic camera parameters are used to compute equivalent 3D positions in an absolute reference frame. The final output is a 3D point cloud for each frame. These clouds can be fused and converted to surface models for further analysis using tools and techniques such as [28], [75], and [86].

3.3 Analysis

3.3.1 On a Two-Frame Stereo Benchmark

I evaluated the QUESS approach for stereo pair inputs using the data and methodology of the well-known Middlebury University stereo benchmark from [103]. The data, discussion, ground truth, and performance evaluations for numerous approaches are available online at http://vision.middlebury.edu/stereo/.

The benefits of QUESS are stronger when processing video data. However, evaluation on this benchmark allows easy comparison to many other two-frame approaches. The following descriptions of input data and metrics are largely adapted from [103].

3.3.1.1 Input Data

Figure 3-5 shows data and (estimated) ground truth disparities for the four scenes used in the evaluation. For each scene a stereo image pair is generated in a tightly

controlled image capture process, with only horizontal fronto-parallel translation between viewpoints. The images are rectified if necessary to establish a standard stereo geometry, and may then be downsampled and cropped to generate the final data.

The Tsukuba scene was originally presented in [90]. Ground truth disparities were labeled by hand at 14 distinct values. This “ground truth” is strongly quantized relative to the true disparity values.

The Venus scene was created in support of [103]. The scene is piecewise-planar by construction. Ground truth disparities were created by hand-labeling those piecewise-planar components, computing the affine motion on each patch, and defining the horizontal component of the affine motion as disparity. Ground truth disparities exhibit 20 distinct values.

The Teddy and Cones scenes were introduced in [104] and incorporated as core datasets in the performance evaluation benchmark. Ground truth disparities were generating using a semi-manual process based on structured light techniques as described in [104].

The significance of the discontinuity masks shown in the right column of Figure 3-5 is described in the next section.

Despite its shortcomings in scene variety, imaging geometry, and ground truth accuracy, these datasets comprise one of the better and most widely-used two-frame stereo benchmarks available at the time of writing.

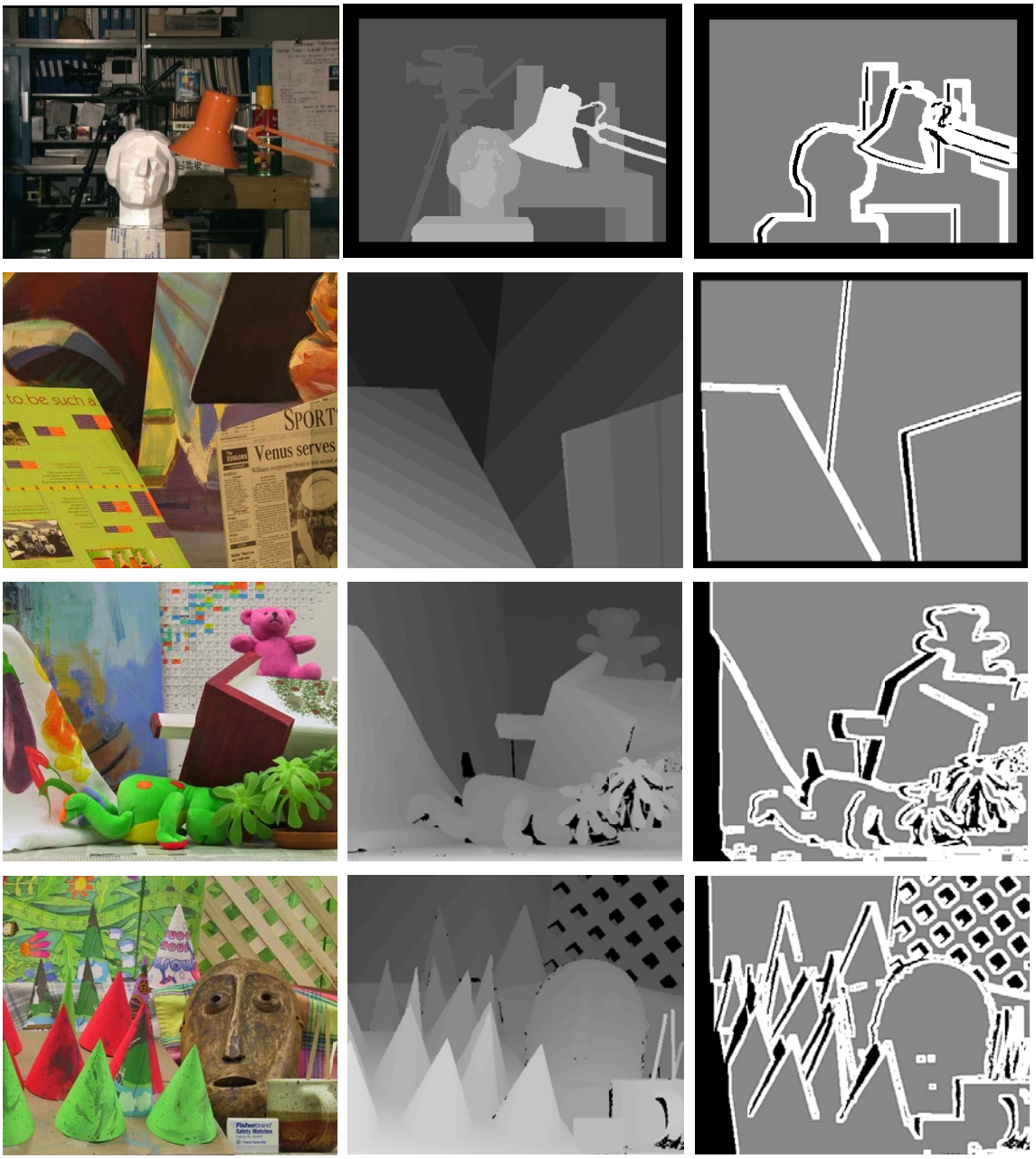


Figure 3-5: Two-frame stereo benchmark data from [103] and [104]. (From top) Tsukuba, Venus, Teddy, and Cones scenes. (Left) Reference input image. (Center) Ground truth disparity. Pseudocolored with higher disparity as lighter grayscale. Pixels without ground truth shown in black. (Right) Map of pixels near depth discontinuities (white), pixels not near discontinuities (gray), and pixels occluded in the non-reference image (black).

3.3.1.2 Performance Metrics

Performance is measured by comparing estimated disparities to ground truth disparities by way of several metrics. The same fundamental metric is measured in multiple regions in each scene, and averaged across scenes to define a final aggregate performance metric.

The fundamental metric used is the *percentage of bad matching pixels*,

$$B = \frac{1}{N} \sum_{(x,y)} \left(\left| d_C(x,y) - d_T(x,y) \right| > \delta_d \right), \tag{16}$$

where N is the number of pixels being considered, (x, y) are their indices in the reference image, d_C and d_T are the computed and true disparities at those indices, and δ_d is a disparity error tolerance. In the online evaluation and in my analysis, typically $\delta_d = 1.0$.

Each scene is divided into regions defined in the reference frame, based on the scene geometry and available ground truth data. These regions are shown in the rightmost column of Figure 3-5:

- **Non-occluded pixels ("Non"):** Pixels in the reference image for which ground truth is known and the matching pixel in the non-reference image is not occluded. Both pixels in the correspondence are visible in their respective images inputs. These are the non-black regions in Figure 3-5.
- **All pixels ("All"):** All pixels in the reference image for which ground truth is available, including those whose match is occluded in the non-reference image.

- **Pixels near discontinuities ("Disc"):** Pixels in the reference image within a small distance of a depth discontinuity that is caused by an occlusion boundary. These are of interest because these pixels typically cause difficulty for stereo algorithms. These are the white pixels in Figure 3-5.

In the evaluation methodology, an approach can tune its parameters for best performance, but all four scenes must be processed using identical values for all algorithm parameters.

3.3.1.3 Results

QUESS performance is compared to Barnard's multi-scale microcanonical annealing approach [7] (described on page 27) which is called "MCA" hereafter, and to Zitnick and Kanade's cooperative approach [124] (described on page 28) which is called "ZK" hereafter.

A variety of parameter combinations were explored and the best results are presented. QUESS used XCORR over a 5x5 window for Q, and influence thresholds of $\alpha = 0$ and $\beta = 0.10$. Its search parameters are given in Table 3-2. Relatively many Q samples are required for a single stereo pair but far fewer samples can used for video data. MCA parameters were set following [7] (minimum temperature 30, maximum temperature 300, 500 iterations per scale, and 85% of iterations for cooling). The parameter λ, which weights smoothness against match quality, was set empirically to $\lambda = 80$. Following [124], ZK used 15 iterations, occlusion threshold 0.005, and a 5x5x3

support region. The inhibition exponent was set to α = 1.25 empirically, and Q was defined as a 1x1 absolute difference (AD). Alternative quality metrics such as 5x5 sum of absolute differences, 1x1 squared differences, and 5x5 sum of squared differences did not improve results.

Numeric results are given in Table 3-3. Figure 3-6 shows reference images and computed disparity for the four stereo pairs included in the benchmark.

QUESS is not competitive with leading approaches on the Tsukuba or Venus scenes. On the Teddy and Cones scenes it performs within the range of results posted for other approaches, although it is not a top performer. This is not unexpected since many algorithms leading the Middlebury evaluation emphasize single stereo pairs of indoor scenes taken at short range, or scenes that have large planar regions, large areas of low contrast, or relatively simple geometries. These scenes allow assumptions and techniques that may be less attractive for outdoor aerial modeling or other data. QUESS performs best on the two scenes that are most representative of outdoor scenes in their complexity, disparity ranges, non-planar geometry, and higher texture. These results show that QUESS is viable even on types of data it does not emphasize

My ZK results did not repeat those achieved in [124], where the authors obtained non-occluded error rates of 1.5-3.0% on the Tsukuba scene. Results for other scenes were not given. The observed performance was significantly different on Tsukuba (27.0-28.5%) and was slightly worse than QUESS on all metrics except those which measure

Parameter	**Search Stage**			
	1	2	3	4
N, Middlebury data	30	30	30	30
N, first video frame	20	20	20	20
N, later video frames	4	6	8	8
W side length	$\frac{\mu_{res}}{30}$	$\frac{\mu_{res}}{40}$	$\frac{\mu_{res}}{60}$	$\frac{\mu_{res}}{120}$
$\delta_{\max}$	0.50	0.25	0.15	0.03

Table 3-2: QUESS search schedule parameters for reported results.

	Tsukuba			Venus			Teddy			Cones			Avg.
	Non	All	Disc	Non	All	Disc	Non	All	Disc	Non	All	Disc	
MCA	20.0	**21.8**	48.6	22.8	24.1	53.3	34.0	39.6	51.3	26.8	32.9	49.1	35.4
ZK	27.0	28.5	**30.4**	27.4	28.1	**41.7**	20.4	27.9	**33.0**	13.8	22.8	**24.6**	27.1
Q	**23.4**	25.1	44.7	**16.2**	**17.7**	44.0	**16.0**	**24.6**	36.6	**11.0**	**19.3**	27.3	**25.5**

Table 3-3: Middlebury evaluation results for Micro-Canonical Annealing (MCA), Zitnick-Kanade (ZK), and QUESS (Q). Non, All, and Disc stand for non-occluded, all, and near discontinuities.

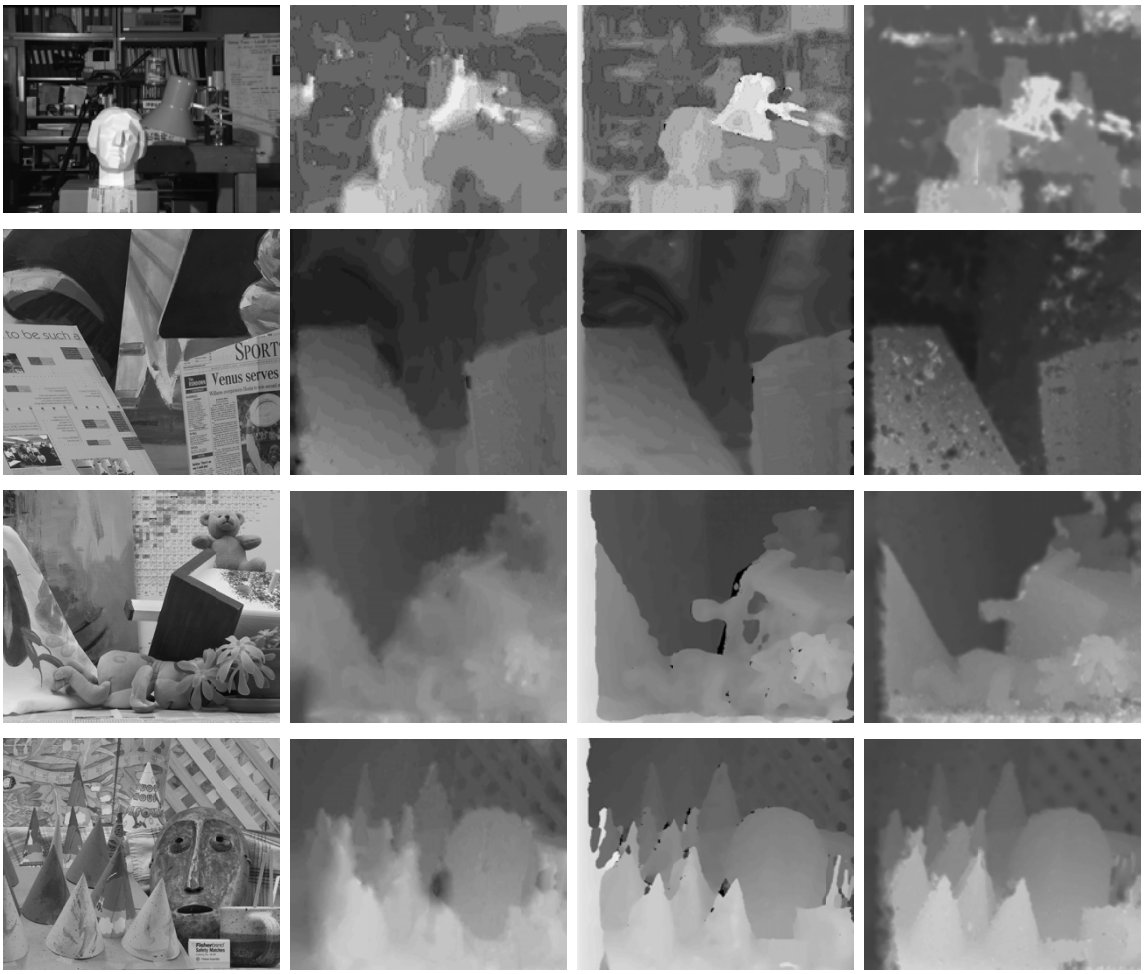


Figure 3-6: Results on Middlebury two-frame stereo benchmark. (By row from top): Tsukuba dataset, Venus dataset, Teddy dataset, Cones dataset. (By column from left) Reference image, Micro-Canonical Annealing (MCA) [7] disparities, Zitnick-Kanade (ZK) [124] disparities, QUESS disparities.

results exclusively near disparity discontinuities. ZK provides a novel method for explicitly identifying occlusions, so superior performance near discontinuities is expected. Because this ZK implementation was my own, the differences between my results and those of [124] imply that an important subtlety of the approach may have been missed in either the published description or in the implementation.

MCA performance was generally poor, both qualitatively and quantitatively. Appealing qualitative results were shown in [7] on other datasets, but earlier MCA results have not been posted for Middlebury data to my knowledge.

Approaches that stochastically sample Q are rare in the literature and none appear among the over 60 approaches with posted Middlebury results at the time of writing. While many approaches use stochastic models of the disparity field, they sample Q exhaustively and apply deterministic optimization algorithms. By combining stochastic and cooperative techniques, QUESS outperforms approaches from each category. No other approach posted among the top 60 performers on the Middlebury data uses non-exhaustive stochastic sampling and optimization. Non-exhaustive sampling of Q provides complexity, runtime, and memory benefits as discussed below.

3.3.2 On Aerial Surveillance Video

3.3.2.1 Input Data

Performance was evaluated on a calibrated monocular aerial video dataset provided by the Air Force Research Laboratory. The dataset contains 32 videos of a suburban scene captured at 60 frames per second (interlaced) and 720x480 resolution. The scene spans 220x225m in the horizontal and 17m in the vertical. Sparse ground truth positions are known for 301 locations, including building corners, fiducial markers, and ground locations. The platform traveled at 35mph at elevations around 110m, with camera declinations of -45 to -50 degrees, yielding true depths in the range of 150m to 220m. Platform position and orientation is known for each frame. Field of view and nontrivial offsets in position and orientation between the platform and camera were estimated by minimizing the re-projection error of ground truth positions. Analysis was performed on a representative 200-frame de-interlaced sequence.

3.3.2.2 Performance Metrics

Calibration inaccuracies in intrinsic and extrinsic parameters shape the definition of the accuracy metric. Sparse ground truth is projected to a 2D pixel location. Depth estimates for all pixels within a radius r are considered and absolute error (AE) is defined as the minimum Euclidean distance between the ground truth position and the estimated 3D positions. Mean absolute error (MAE) averages AE over all visible ground truth

points and all frames. This defines a family of MAE metrics E_r with error values decreasing monotonically with r.

Results are presented for E_5, which was selected by inspection based on residual re-projection error. Between 1800 and 2200 ground truth comparisons contributed to each MAE value. Results are presented for downsampled 360x240 video due to the memory limitations of the ZK approach, discussed in Section 3.3.5.

3.3.2.3 Results

A variety of parameter combinations were explored and the best results are presented. QUESS used XCORR over a 5x5 window for the match quality metric, and influence thresholds of $\alpha = 1.5$ and $\beta = 0.15$. Its search parameters are given in Table 3-2 on page 58. QUESS requires significantly fewer Q evaluations per frame when processing video than for stereo pairs. MCA parameters were identical to the evaluation on Middlebury data. ZK used 10 iterations, occlusion threshold 0.02, a 5x5x3 support region, $\alpha = 1.5$ and 5x5 sum of absolute differences (SAD) for the match quality metric.

The target stereo baseline was varied over $0.01 \leq \tau_0 \leq 0.20$ for MCA and QUESS, and up to 0.25 for ZK to capture all important trends. Intentionally loose elevation assumptions simulated imprecise *a priori* scene knowledge (45m vertical range versus the actual 17m range) and defined the disparity ranges for each approach. MCA and ZK require a standard stereo geometry, so a planar rectification following [44] was applied.

Disparity estimates were converted to depth estimates and transformed to the reference frame for evaluation.

Figure 3-7 shows a reference image and example reconstructions for MCA, ZK, and QUESS. Results are shown for $\tau_0 = 0.07$, at which QUESS achieves its best accuracy. Figure 3-8 plots reconstruction error E_5 against target stereo baseline ratio τ_0 for the three approaches.

As seen in Figure 3-8, QUESS outperforms both MCA and ZK at the sparse evaluation positions. QUESS achieves $E_5 = 1.15$m at $\tau_0 = 0.07$. This equates to 0.62% estimate error relative to absolute depth, which is 0.29 pixels average disparity magnitude error at that baseline. ZK performance is somewhat competitive, and achieves $E_5 = 1.88$m at $\tau_0 = 0.18$, resulting in 1.01% depth estimate error and 1.50 pixels average disparity magnitude error. MCA performance is not competitive, achieving a minimum $E_5 = 3.39$ at $\tau_0 = 0.08$. These relative results are consistent with evaluations on the Middlebury data.

A few general trends are evident. The accuracy of all approaches degrades for small τ_0, where the reduced disparity range makes disparity estimates sharply quantized, and depth estimates are simultaneously more sensitive to disparity error. Metrics defined at sub-pixel disparities ([110]) would help, but increased sensitivity will remain. QUESS and MCA accuracy degrade at higher τ_0 where viewpoint changes make correspondence

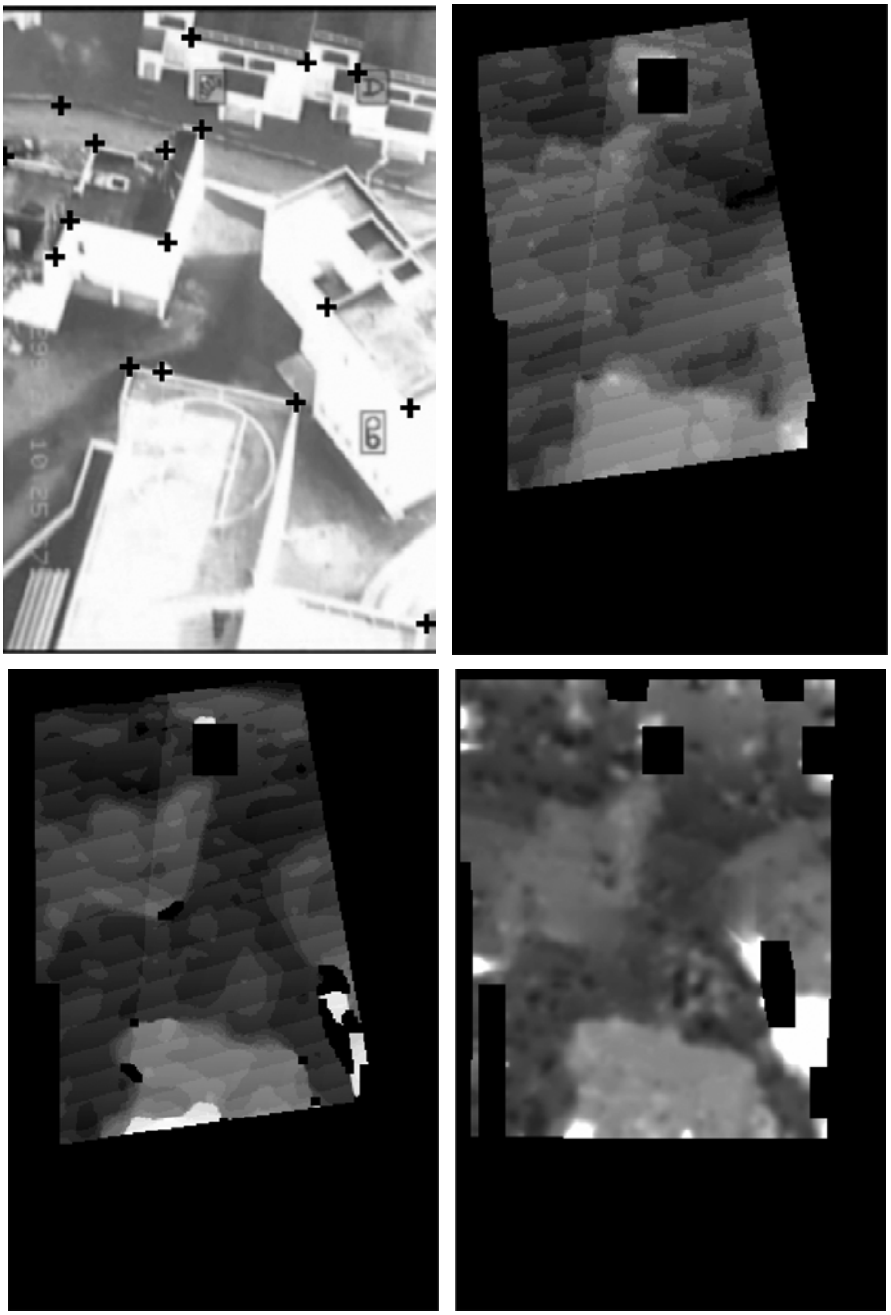


Figure 3-7: Example reconstructions from aerial video. (Top left) Reference image with sparse evaluation positions marked, (Top right) MCA estimated elevations, (Bottom left) ZK estimated elevations, (Bottom right) QUESS estimated elevations.

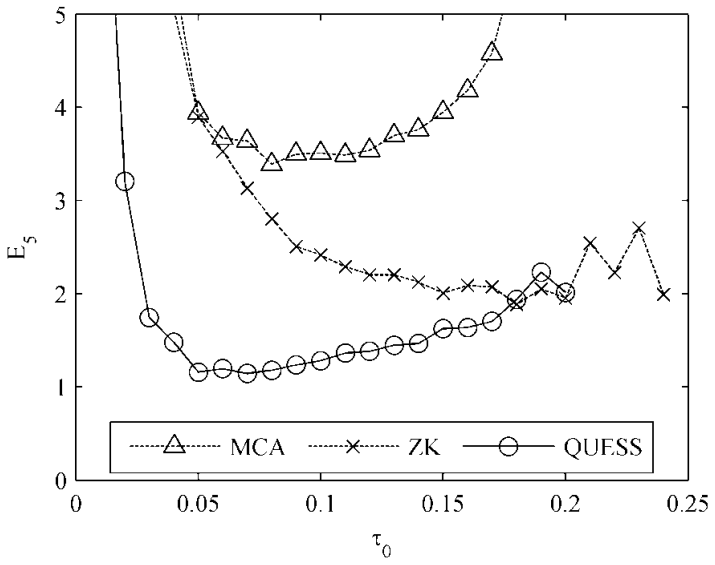


Figure 3-8: Reconstruction error E_5 versus stereo baseline τ_0 for MCA, ZK, and QUESS approaches.

matching more difficult. ZK accuracy gradually improves as τ_0 grows and the effects of disparity quantization are reduced. ZK accuracy becomes unstable as τ_0 grows further, likely because few frames are successfully paired and few depth estimates are generated per frame. QUESS outperforms ZK and MCA, but it can produce inaccurate results in large regions of low texture or contrast (as do most other approaches).

These results demonstrate that by combining stochastic and cooperative techniques, QUESS outperforms both a stochastic approach and an exhaustive cooperative approach on a realistic and complex dataset. Accurate range estimates are generated from high-range calibrated aerial video. QUESS has a variety of additional advantages which are discussed next.

3.3.3 Number and Scalability of Match Quality Evaluations

The primary advantages of QUESS are the generation of depth estimates using fewer evaluations of Q, and its attractive scaling properties with respect to video resolution, stereo baseline ratio, and scene bound assumptions.

Analysis of Q evaluations is focused on comparing to ZK because ZK can be used as a representative for other approaches. For example, the current MCA implementation is not competitive with respect to Q evaluations because Q is recomputed on each iteration which results in 500 Q samples per pixel *per scale*. MCA could be re-implemented to exhaustively sample Q at each scale and use a lookup table, but then MCA would still be no better than ZK in the number of Q samples. Any approach that exhaustively samples Q will encounter the same scalability issues as ZK.

QUESS requires

$$K_Q = 2RC\alpha_Q(\vec{p},\tau_0)N_{tot} \tag{17}$$

evaluations of Q for R x C images, where $\alpha_Q(\vec{p},\tau_0)$ is the fraction of overlapping pixels in the stereo pair (a function of τ_0 and camera path $\vec{p}$), and N_{tot} is the total number of iterations in the schedule. The value $\alpha_Q \in [0, 1]$, so $K_Q \le 2RCN_{tot}$. QUESS generates $RC\,\alpha_Q(\vec{p},\tau_0)$ depth estimates per frame pair.

Exhaustive approaches are more difficult to characterize. ZK requires

$$K_{ZK} = R^*(\vec{p},\tau_0)C^*(\vec{p},\tau_0)\alpha_{ZK}(\vec{p},\tau_0)D(\vec{p},\tau_0;R,C,B) \tag{18}$$

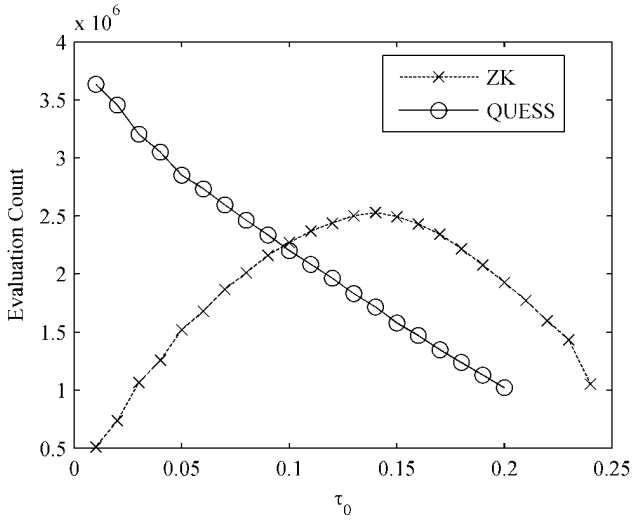


Figure 3-9: Average number of Q function evaluations per 720x480 frame (345,600 pixels) for the QUESS and ZK approaches. ZK is used in this analysis as a representative of all exhaustive sampling approaches because all will share similar characteristics in the number of Q evaluations.

evaluations, where R^* and C^* are the number of rows and columns after projective rectification, α_{ZK} is analogous to α_Q, and D is the *range* of potential disparities for resolution *RxC* and scene bounds B. D is a complex function of camera path and stereo baseline that grows with increasing baseline. An upper bound on D is not easily determined. ZK generates $RC\alpha_{ZK}(\vec{p},\tau_0)$ depth estimates per frame pair.

Figure 3-9 plots the average Q evaluations per frame for the aerial video test data, as a function of τ_0. QUESS requires fewer evaluations per frame only in some ranges. QUESS shows a steady decline in K_Q as α_Q shrinks with increasing τ_0 and other factors remain constant. For ZK, the number of evaluations grows with τ_0 for low τ_0 until a

decrease in α_{ZK} dominates and K_{ZK} follows. The values of R^*, C^*, D, and α_{ZK} are all complicated functions of camera path and baseline. QUESS achieves its best accuracy at 2.6×10^6 Q evaluations per frame and ZK achieves its best accuracy at 2.2×10^6 per frame. On the surface, ZK may appear superior in number of Q evaluations (although its accuracy is worse), but this is not the complete story.

Figure 3-10 plots the number of depth estimates generated per frame. Increasing the stereo baseline decreases the number of estimates per frame as overlap decreases. Qualitatively different behavior is seen in the number of evaluations of Q *per depth estimate* in Figure 3-11. QUESS uses a constant number of match quality evaluations per depth estimate, independent of stereo baseline. The number of ZK evaluations per estimate is dominated by the growth of D with increasing τ_0. QUESS achieves its best performance at 52 evaluations per estimate, but ZK requires an average of 203 per estimate. QUESS generates more accurate results using 75% fewer Q evaluations per estimate.

The specifics of the analysis will differ between exhaustive approaches, but the themes generalize. For roughly linear camera paths, all factors α_* will decrease with increasing τ_0 (here $\alpha_Q > \alpha_{ZK}$ by coincidence only). However, α_*, R^*, and C^* are determined by frame pairing and rectification. Any exhaustive algorithm that needs a standard stereo geometry and uses the same projective rectification will share these same

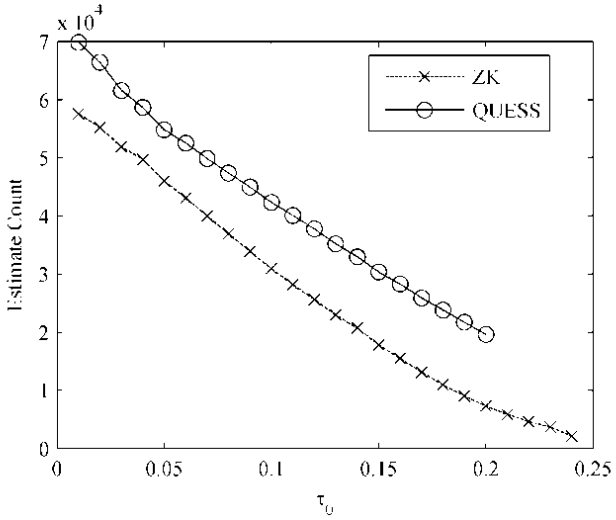


Figure 3-10: Average number of depth estimates per 720x480 frame (345,600 pixels) for the QUESS and ZK approaches. The number of depth estimates is dictated in large part by rectification and input bounding. ZK is used as a representative of all approaches requiring planar rectification to standard epipolar geometry, because all will share similar characteristics in the number of depth estimates produced.

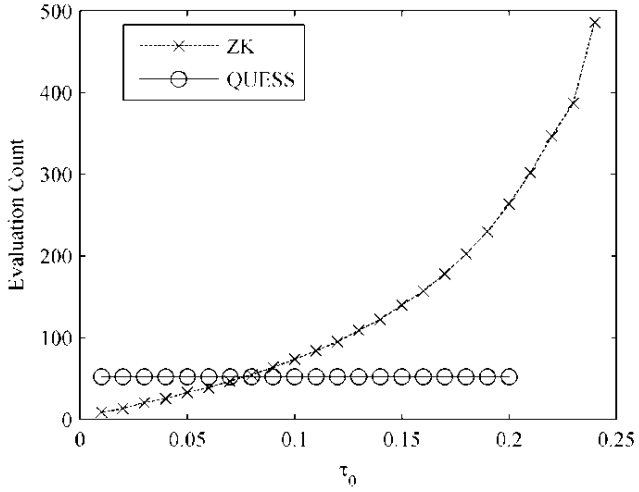


Figure 3-11: Average number of Q function evaluations per depth estimate for the QUESS and ZK approaches. ZK is used as a representative of exhaustive sampling approaches that require rectification to a standard stereo geometry. All of these approaches, which constitute the majority of two-frame approaches in the literature, will share similar characteristics in the number of Q evaluations per depth estimate.

values. The value D is also shared. For any exhaustive approach, α_* will decrease with τ_0 and D will increase, resulting in more evaluations of Q per estimate. By contrast, QUESS can freely optimize τ_0 without changing the number of Q evaluations.

QUESS also has more attractive scaling properties with respect to resolution and other factors. For QUESS, Q evaluations scale directly with $O(n)$ (n = number of pixels). For ZK, doubling R and C also doubles D, so K_{ZK} scales with $O(n^{3/2})$. This applies to any method that exhaustively computes Q at integer disparity magnitudes in a search range based on camera and scene geometry – doubling resolution causes an unavoidable $O(n^{3/2})$ scaling in evaluations of Q.

ZK inhibition computations scale with $O(n^2)$ in the number of pixels, as opposed to the $O(n^{3/2})$ scaling described in [124]. For each row, column, and disparity, inhibition is summed over a second disparity index whose range is also linear in resolution. For simple Q metrics, computing ZK inhibition may dominate computing Q, but that is not necessarily true for complex Q metrics or for other approaches.

Scene geometry and camera path have complex and significant effects on D for any exhaustive approach, further complicating their use under uncontrolled scene and camera geometries. By contrast, the number of Q evaluations in QUESS is independent of scene geometry and camera path once the number of iterations is chosen.

	Middlebury (average)			Aerial Video		
Resolution	FR	1/2	1/4	1/2	1/4	1/8
MCA	95.7	21.4	5.7	10.9-27.5	3.8-8.1	1.6-2.5
ZK	95.9	12.3	1.3	15.6-23.0	1.8-2.8	0.3-0.4
QUESS	359.1	72.5	19.0	41.1	9.5	2.9

Table 3-4: Runtime (seconds per frame) for Micro-Canonical Annealing (MCA), Zitnick-Kanade (ZK), and QUESS processing of Middlebury stereo pairs and aerial video data. Results are given for full-resolution (FR) inputs of 720x480, half-resolution (1/2), 1/4 resolution, and 1/8 resolution.

3.3.4 Runtime

Runtimes are given in Table 3-4 for each dataset at three different resolutions. Runtime was measured with a single 2.5 GHz dual-core CPU with 3.5 GB RAM. While QUESS was not the fastest of the three approaches, a direct comparison does not reflect all the relevant issues. Algorithms were implemented in Matlab and were vectorized, but with no effort to optimize the implementations. As a result, QUESS did not capitalize on opportunities to reduce the number of match quality evaluations by exploiting increases in τ_0. MCA and ZK, however, do benefit because lower frame overlap shrinks the size of the data cube created for rectified stereo pairs.

QUESS runtimes were significantly lower on video data than on the single stereo pairs because the initialization techniques described in Section 3.2.2 allow its search

schedules to be shortened. Runtime scales with number of pixels, increasing by a factor of about four for every doubling in resolution. QUESS runtimes are independent of τ_0, and for an optimized implementation would actually decrease with increasing τ_0. This insulation of runtimes also applies to stereo geometry and relative scene orientation. Neither characteristic holds for approaches that exhaustively sample *Q*.

ZK runtimes follow the number of *Q* evaluations shown in Figure 3-9. Runtime ranges are given because ZK runtime varies significantly with τ_0. As expected, each doubling of resolution creates an 8-fold increase *per frame* in *Q* evaluations, an approximate 8-fold increase in total runtime, and doubling of *Q* evaluations and runtime *per depth estimate*. ZK thus scales with $O(n^{3/2})$ in the number of pixels.

MCA runtimes also follow the number of *Q* evaluations, and also vary significantly with τ_0. *Q* evaluations scale linearly in pixels in the current implementation because *Q* is recomputed each iteration. If the disparity space was instead quantized and sampled exhaustively, MCA would scale with $O(n^{3/2})$ like ZK but with a much lower hidden constant that it currently has.

The use of simple *Q* functions actually minimizes the runtime advantages of QUESS over other approaches. As more complex metrics are used, evaluation of *Q* becomes a larger percentage of runtime and the advantages of quality-efficient stochastic sampling become more pronounced.

3.3.5 Memory Usage

QUESS has memory advantages over approaches that exhaustively sample Q and retain all samples in memory, and it is thus more attractive than exhaustive approaches on memory-constrained devices or platforms. QUESS requires storing approximately 20 floating-point values for each pixel, independent of τ_0. All aspects of memory usage scale with $O(n)$ and are independent of stereo baseline, frame overlap, camera motion, and scene structure.

ZK memory usage follows directly from the number of Q evaluations analyzed in Section 3.3.3, since all Q samples are stored in a 3D data cube. As a result, ZK memory also scales with $O(n^{3/2})$. ZK requires two data cubes of this size. At its best accuracy, ZK requires the simultaneous storage of over 400 floating point values per depth estimate.

As currently implemented, MCA memory requirements are similar to QUESS because Q is recomputed in each iteration. If Q values were instead computed once at all disparities and stored, MCA memory usage would become nearly identical to ZK. Both alternatives have significant disadvantages for MCA.

Similar scaling properties are shared by other exhaustive approaches. The size of any exhaustive cube of Q samples is affected by stereo baseline, frame overlap, camera motion, scene orientation, and scene structure. Few of these factors are easily controlled so exhaustive approaches can cause problems on limited memory devices. Aerial video comparisons were performed at 360x240 resolution because even on a modern machine

with significant memory, ZK generated Matlab out-of-memory errors on larger imagery.

3.4 Conclusions

This chapter presents and analyzes Quality-Efficient Stochastic Sampling (QUESS), a new stochastic approach for dense stereo correspondence matching that uses fewer local match quality metric evaluations than exhaustive approaches. It is based on a set of general techniques that are easily applied to a variety of applications; its strengths are maximized when operating on calibrated monocular video. QUESS is both stochastic and cooperative, with advantages from both.

QUESS exploits the piecewise continuity and continuity of matching likelihood constraints [65] to skip portions of the disparity search space. Estimates are initialized from the previous frame pair's results in a novel application of the Z-buffering algorithm. This allows convergence across multiple frame pairs. A relatively simple formulation of *local influence* selectively re-weights random perturbations injected into the solution, and *aggregated influence* extracts consistent trends from the stochastic sampling of match quality.

QUESS has a number of advantages over exhaustive approaches. It was shown to outperform both Barnard's stochastic approach [7] and Zitnick and Kanade's cooperative approach [124] on a complex and representative video dataset. It requires fewer match quality metric evaluations per depth estimate, with corresponding gains in efficiency. It reduces memory requirements and provides better scaling in both runtime and memory.

Runtime and memory usage are insulated from a variety of factors that cannot be easily controlled, including stereo baseline, camera path, and scene orientation and structure. Iterative cooperative processing allows straightforward control of runtimes. Its advantages are demonstrated using simple quality metrics, but will become even more pronounced as metric complexity increases. QUESS facilitates the use of complex and robust metrics [50], as well as metrics defined on non-integer disparities [110].

QUESS is particularly beneficial for 3D modeling from small resource-constrained platforms. In these scenarios, computational complexity and memory must be tightly controlled to remain within the limits of the platform. QUESS reduces these requirements directly, and also insulates them from outside factors that are difficult to control in practical use.

Chapter 4.

Distributed Ray Tracing Applied to Hough Transform

This chapter presents two novel extensions to the Hough transform (HT) for line detection based on applications of distributed ray tracing (DRT). Their performance is analyzed and compared to other Hough transform variations on line detection tasks.

This work (also submitted for publication as [22]) is relevant because DRT will be used in Chapter 5 to improve the performance of a HT-based multi-view stereo algorithm. Understanding the benefits of DRT to traditional HT applications informs analogous extensions in HT-based multi-view stereo.

4.1 Approach

Distributed ray tracing can be applied in two distinct ways, resulting in two new HT variations.

4.1.1 The Distributed Ray Hough Transform

The distributed ray Hough transform (DRHT) extends the re-accumulation approach of [39]. In DRHT, N trajectories are accumulated in Hough space for each edge pixel. Each trajectory is randomly perturbed according to

$$\rho = (x + \Delta_x)\cos\theta + (y + \Delta_y)\sin\theta\,, \tag{19}$$

where Δ_x and Δ_y are drawn from densities $P(\Delta_x)$ and $P(\Delta_y)$. The experiments used uniform distributions over [-0.5, 0.5] such that the perturbed values "cover" the 2D extents of the pixel.

Instead of sampling the trajectory at increments of size Δ_θ as in the SHT, the DRHT samples using finer increments $s\Delta_\theta$, where $0 < s < 1$. Although the trajectories are sampled more finely, the accumulators A and B are not enlarged. I show below that DRT (hence DRHT) in fact allows accurate results for *reduced* accumulator sizes.

Re-accumulation can be augmented similarly. Each pixel votes in B only for θ^* at the maximum of the *sum* of N trajectories in A, which are also perturbed by $P(\Delta_x)$ and $P(\Delta_y)$ and sampled at increments of size $s\Delta_\theta$. Each pixel casts N votes in B at locations defined by

$$\rho = (x + \Delta_x)\cos\theta^* + (y + \Delta_y)\sin\theta^*\,. \tag{20}$$

4.1.2 Extensions to Palmer et al.'s Peak Refinement Approach

The second variation extends the soft kernel approach of [94]. Processing follows [94], but with the original kernel, K_P, replaced by

$$K_{DRT}(x, y, \theta, \hat{\rho}) = \frac{1}{N}\sum_{i=1}^{N} K_{SHT}((x + \Delta_{xi})\cos\theta + (y + \Delta_{yi})\sin\theta - \hat{\rho})\,. \tag{21}$$

K_{DRT} averages the contributions of N evaluations of K_{SHT} with pixel perturbations following $P(\Delta_x)$ and $P(\Delta_y)$. In practice Δ_{xi} and Δ_{yi} are computed once for each pixel and stored, for efficiency and repeatability.

The three-stage cubic peak parameter optimization algorithm of [94] can be effectively replaced by a simpler brute-force search through the sub-cell values of ρ and θ. The original optimization is not appropriate for K_{DRT} since K_{DRT} is not continuous. K_{DRT} is, however, defined at all (ρ, θ) and varies within accumulator bins, so other optimization algorithms can be used. Brute-force search was used to minimize implementation issues, but it is not germane to the approach.

The definition of K_{DRT} makes it possible to compute an approximate "equivalent kernel" defined in terms of $(\rho - \hat{\rho})$ for various $P(\Delta_x)$, $P(\Delta_y)$, and w, to compare to K_P and K_{SHT}. Comparison is done by evaluating K_{DRT} with $x = y = \theta = 0$. Figure 4-1 compares K_{SHT}, K_P, and $K_{DRT}(0,0,0,\hat{\rho})$ for $w = 1.25$ (a nominal value used in [94]) and

$$P(\Delta_x) = P(\Delta_y) \approx \mathbf{U}(-0.65, 0.65) + \mathbf{N}(\mu = 0, \sigma^2 = 0.25), \tag{22}$$

where $\mathbf{U}(a,b)$ denotes a uniform distribution on $[a,b]$. $P(\Delta_x)$ and $P(\Delta_y)$ are chosen to generate a K_{DRT} close to K_P, although the shape of $K_{DRT}(\rho - \hat{\rho})$ will also vary with x, y, and θ.

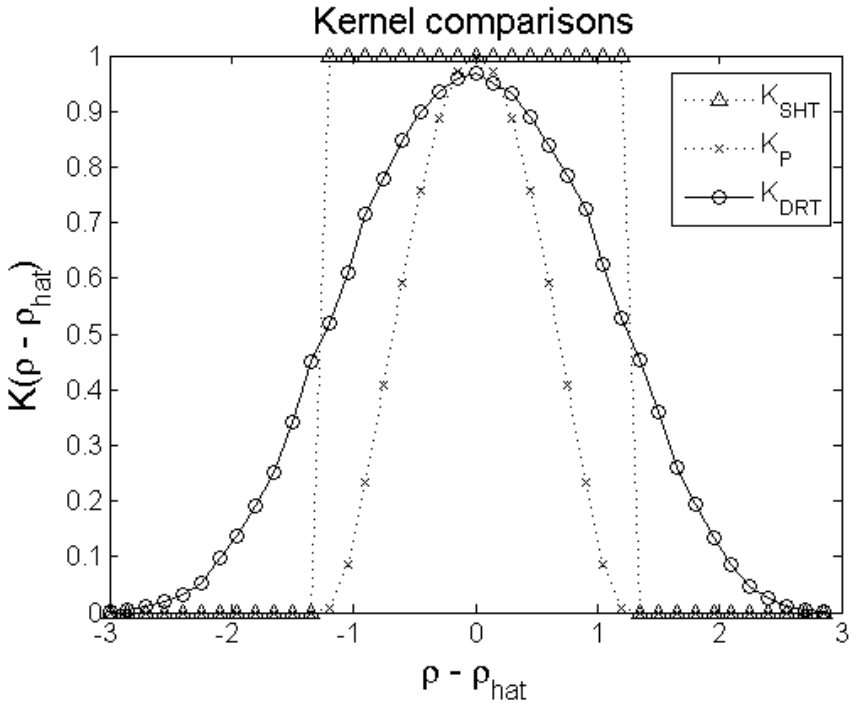


Figure 4-1: Comparison of Hough transform accumulation kernels for the standard Hough transform (K_{SHT}), Palmer's extension (K_P) for w=1.25, and distributed ray Hough transform (K_{DRT}) for $P(\Delta_x) = P(\Delta_y) \approx \mathbf{U}(-0.65, 0.65) + \mathbf{N}(\mu = 0, \sigma^2 = 0.25)$.

4.2 Analysis on Line Detection and Parameterization Experiments

4.2.1 Experiment Design

Line detection and localization performance was measured on synthetic data. Each scene contains 10 lines with randomly placed endpoints on the ideal range [0,1]x[0,1]. Endpoints were not quantized, thus the true HT parameters are also not quantized. Lines are rendered using [118] with anti-aliasing turned off to simulate binary edge images. Unlike the Bresenhem algorithm, [118] correctly reflects sub-pixel endpoint

positions. The same ideal scenes are rendered at resolutions from 300x300 to 30x30. Figure 4-2 shows an example scene at different resolutions.

Each approach accepts a binary input image and returns a set of estimated peak locations $\left(\hat{\rho}_i, \hat{\theta}_i\right)$. The true number of lines in each image is not specified *a priori*. The estimated HT parameters are compared to ground truth on 10 or 25 scenes and the results averaged. Explicit segment endpoint estimates are ignored as the primary interest is HT parameter estimation accuracy. Each experiment passes the same scenes to every approach and parameterization.

Six HT variations are evaluated in two groups of three to compare the proposed DRT-enabled approaches against alternatives. For each approach, all parameters but one are fixed at empirically chosen values. A detection threshold T is allowed to vary freely as would be done to generate a receiver operator characteristic (ROC) curve [33].

Group 1 uses no explicit sub-cell peak refinement. The standard Hough transform ("SHT") is applied without re-accumulation by applying threshold T to A and returning only local maxima. The SHT is not competitive but it is used to provide context for performance improvements. The approach of [39] (hereafter "Gerig") is applied with T operating on B and only local maxima returned. DRHT is applied without sub-cell peak refinement and with $N = 25$, $s = 1/3$, and pixel indices perturbed with $\mathbf{U}(-0.5,0.5)$. The threshold T is applied to B and only local maxima are returned.

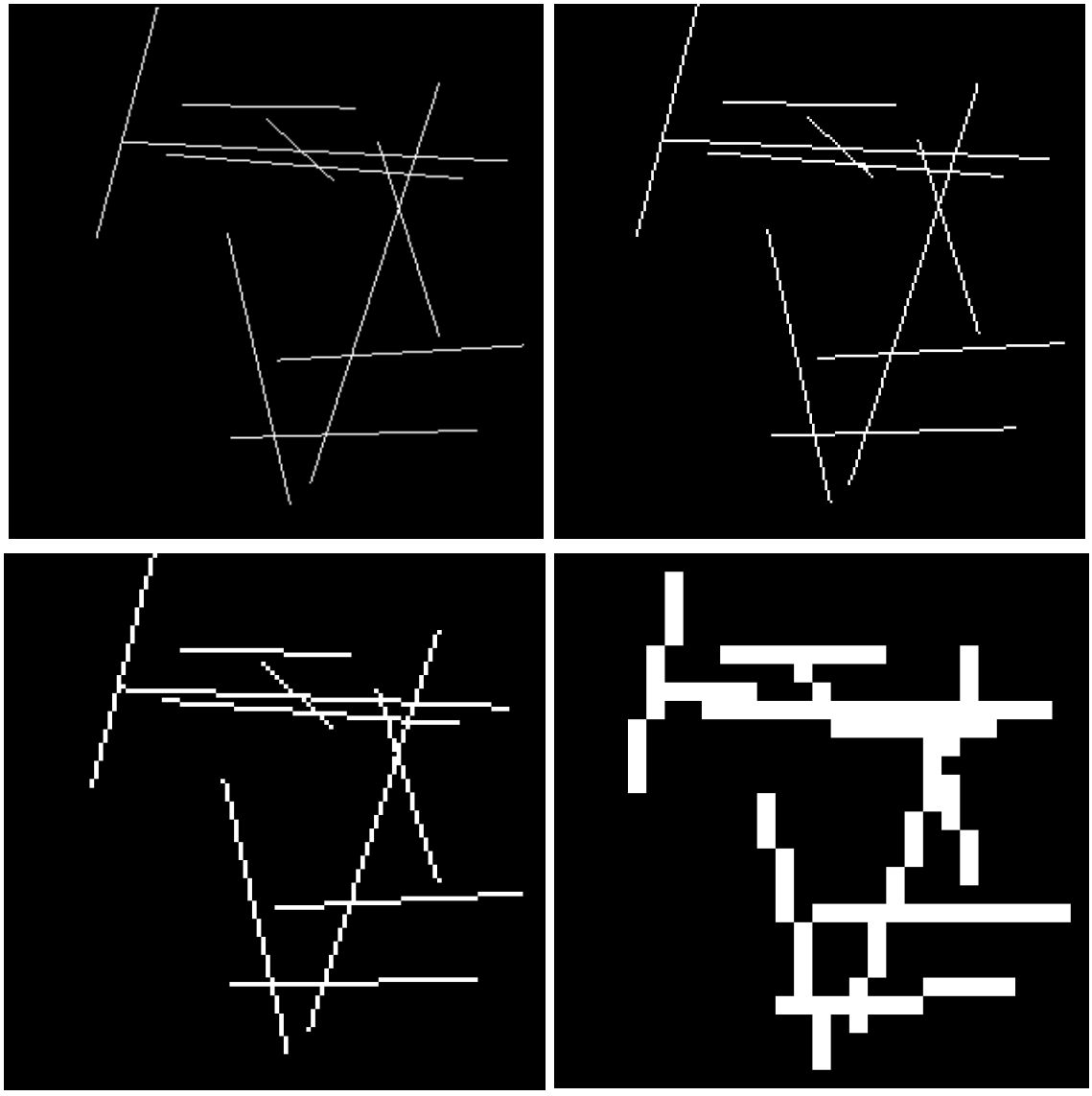


Figure 4-2: (Raster order from top left) An example Hough transform line detection test scene rendered at 300x300, 210x210, 120x120, and 30x30 resolution. Experiments render the same field of view and set of ideal scenes (each defined by the ideal line endpoints) at different resolutions.

Group 2 uses sub-cell peak refinement. The approach of [94] (hereafter "Palmer-K_P") is applied with $w = 1.25$ and T applied to B. The tests for maximum gap and minimum pixels per segment are disabled because endpoint locations are not of interest. The original peak optimization algorithm is replaced by an exhaustive 20x20 search within the extent of the winning cell in B. The approach is also applied with the DRT-based kernel ("Palmer-K_{DRT}") and $w = 1.25$, $N = 50$, and the perturbations of (22). The threshold T and peak optimization are applied as in Palmer-K_P. DRHT is tested in Group 2 with peak refinement added ("DRHT+refine").

4.2.2 Performance Metrics

Performance is measured by the F_1 score and root-mean-squared (RMSE) error in the HT parameters, after a greedy bipartite matching process assigns estimated segments to true segments.

The F_1 score is the geometric mean of recall and precision. Recall is the percentage of true segments that are detected by the approach. Precision is the percentage of segments returned by the approach that are true segments.

The RMSE is computed between the true and estimated HT parameters of only true positives. There is an unavoidable and often arbitrary tradeoff in any test measuring both detection and localization. It is defined by the criteria that distinguish a poorly localized true positive from a false positive. False positives in a 300x300 image are

defined by inspection as error in ρ over 10 pixels or error in θ over $\frac{10 \cdot 2\pi}{360}$ radians (10°), with no errors in the other dimension.

A scalar distance is needed within the HT domain. This is problematic because the units of *pixels* and *radians* have no joint meaning. Further, [73] implies that the optimal aspect ratio between ρ and θ depends on image size, since precision in θ increases with image size but precision in ρ does not. Ten pixels in a 300x300 image (the false positive threshold) cover a linear distance of $\frac{10}{300}$. That distance is equated with 10° in the θ dimension to define the resolution-independent Hough domain distance,

$$E_{HT}(\rho_1, \theta_1, \rho_2, \theta_2) = \left[\left(\frac{300}{R_1}\rho_1 - \frac{300}{R_2}\rho_2 \right)^2 + \left(\frac{360}{2\pi}\theta_2 - \frac{360}{2\pi}\theta_1 \right)^2 \right]^{1/2}, \quad (23)$$

where R_1 and R_2 are the resolutions of the first and second image, respectively.

Any estimated peaks with $E_{HT} > 10$ are considered false positives, and RMSE is computed for true positives with respect to E_{HT}.

4.2.3 Results and Discussion

I conducted experiments to measure performance when the image resolution R_I, HT accumulator resolution R_{HT}, or both, are reduced. Reducing the values of R_I and/or R_{HT} may be necessary to satisfy memory or complexity requirements. Or super-resolution

accuracy may be desired (e.g., [20]). Full resolution imagery and accumulators are defined as R_I^* = (300, 300) and R_{HT}^* = (2401, 1885), following [73].

The values of R_I, R_{HT}, or both are reduced to 10% of their ideal sizes. T is varied to cover the performance envelope from high recall and low precision, and low recall and high precision. Maximum F_1 is achieved between these two extremes. The *maximum* F_1 achieved by each approach is reported for each R_I and R_{HT}. The RMSE over E_{HT} is also reported for each approach at a *selected* common F_1 score – typically the highest F_1 achieved by all approaches in the group.

Figure 4-3 shows F_1 and RMSE results when R_I and R_{HT} are reduced in tandem. For Group 1, size ratios of less than 0.5 degrade both F_1 and RMSE across all approaches. SHT is not competitive on F_1 but is competitive with both Gerig and DRHT on RMSE. Gerig's re-voting technique significantly improves F_1. DRHT shows marginal improvement in F_1 relative to Gerig but outperforms the other approaches in terms of RMSE at all sizes. All Group 1 approaches show steadily improving F_1 and asymptotically improving RMSE with increasing size ratio.

Group 2 generally outperforms Group 1 in terms of both F_1 and RMSE. Peak refinement improves DRHT's F_1 slightly, but the Palmer variations consistently achieve higher F_1. RMSE is similar across Group 2. The RMSE of the Palmer variation is affected by a reduced ratio. The F_1 score for the Palmer variation is only affected at small ratios (< 0.3).

Figure 4-4 shows results when only R_I is reduced. In Group 1, SHT is again noncompetitive using F_1. Gerig provides superior F_1 but no improvement to RMSE. Both F_1 and RMSE improve as R_I is increased for both SHT and Gerig. DRHT achieves similar F_1 score as Gerig (likely due to re-voting). DRHT significantly outperforms Gerig in terms of RMSE for all R_I, and particularly for small R_I. At ratios > 0.4, DRHT RMSE is no more than 2x its performance at R_I^*.

Palmer-K_{DRT} significantly outperforms other Group 2 approaches on F_1, followed by Palmer-K_P and finally DRHT. DRHT peak refinement does not improve F_1 but slightly improves RMSE. Palmer-K_{DRT} and DRHT significantly outperform Palmer-K_P on RMSE for ratios $\leq$ 0.7.

Figure 4-5 shows results when only R_{HT} is reduced. In Group 1, Gerig and DRHT again outperform SHT on F_1. Against expectations, F_1 generally increases at lower R_{HT}. This may be due to reduced *peak extension* across θ bins [73]; Lam *et al.*'s optimal quantization may not be optimal for all approaches. RMSE is similar across Group 1 with DRHT having slightly superior performance. Reducing R_{HT} significantly degrades performance at low ratios (< 0.3).

Group 2 exhibits high F_1 throughout, with DRHT and Palmer-K_{DRT} (for small RHT only) slightly lower. RMSE is generally very low for all approaches and outperforms Group 1, except for Palmer-K_{DRT} for small R_{HT} (< 0.3). These results imply that peak refinement effectively mitigates moderate reductions in the size of R_{HT}.

SHT was non-competitive across all experiments. Group 2 generally outperforms Group 1. DRHT improves on Gerig but is not always competitive with the Palmer variations. Palmer-K_P and Palmer-K_{DRT} are the top performers. Their performance is often similar, but Palmer-K_P is superior at very low R_{HT} (< 0.3) and Palmer-K_{DRT} is superior for most reduced values of R_I (0.2 to 0.8). DRT-enabled variations tend to show improvements to F_1, RMSE, or both, when measured against the performance of analogous non-DRT variations.

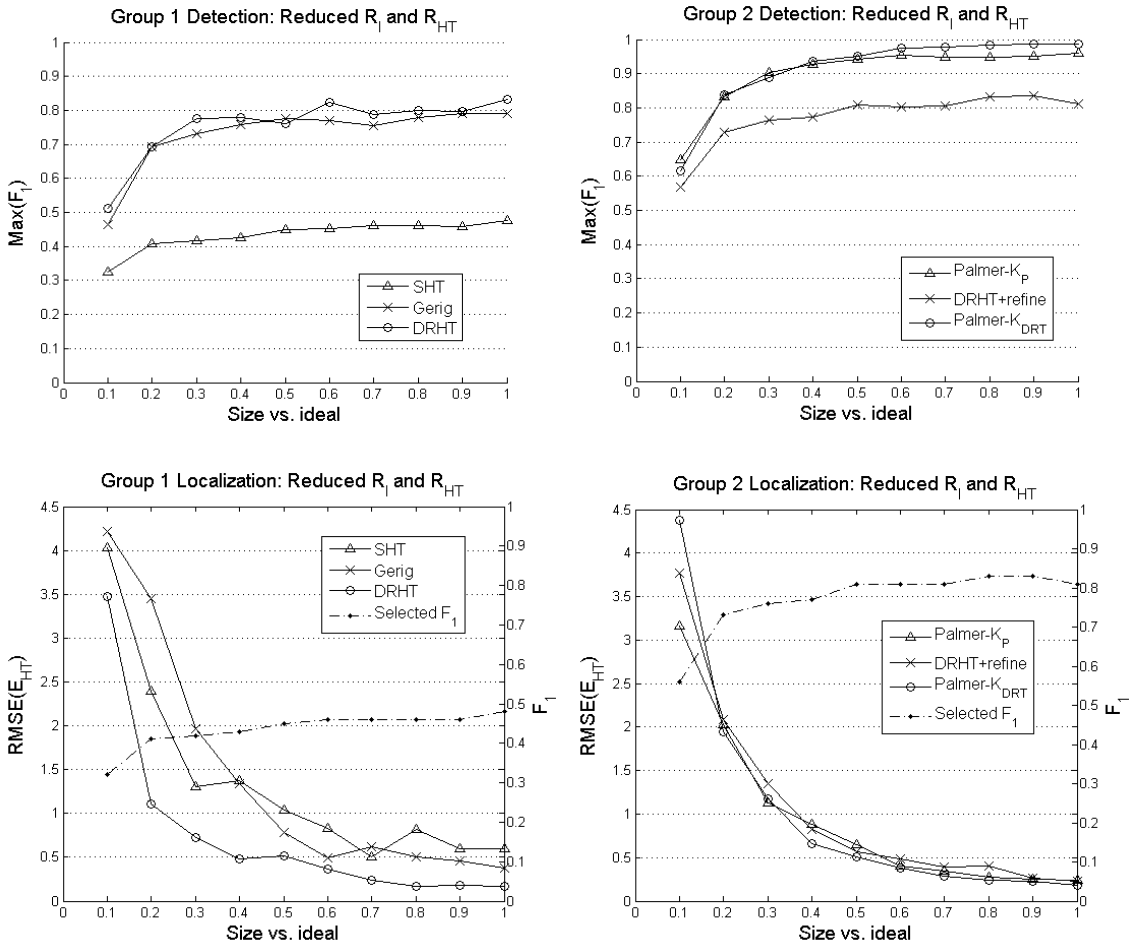


Figure 4-3: Hough transform line detection and localization results when image size R_I and accumulator size R_{HT} are reduced in tandem.

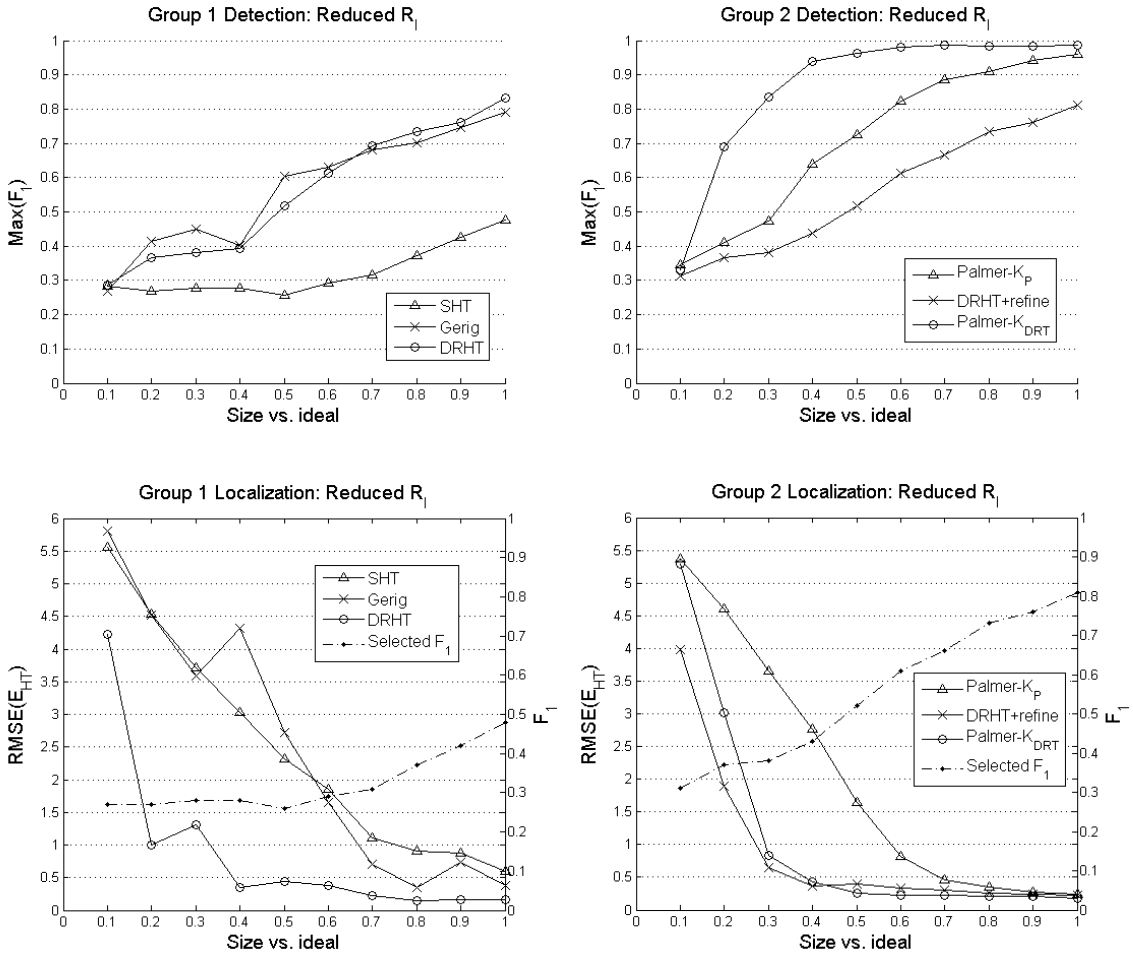


Figure 4-4: Hough transform line detection and localization results when image size R_I is reduced and accumulator size R_{HT} is maintained at full resolution.

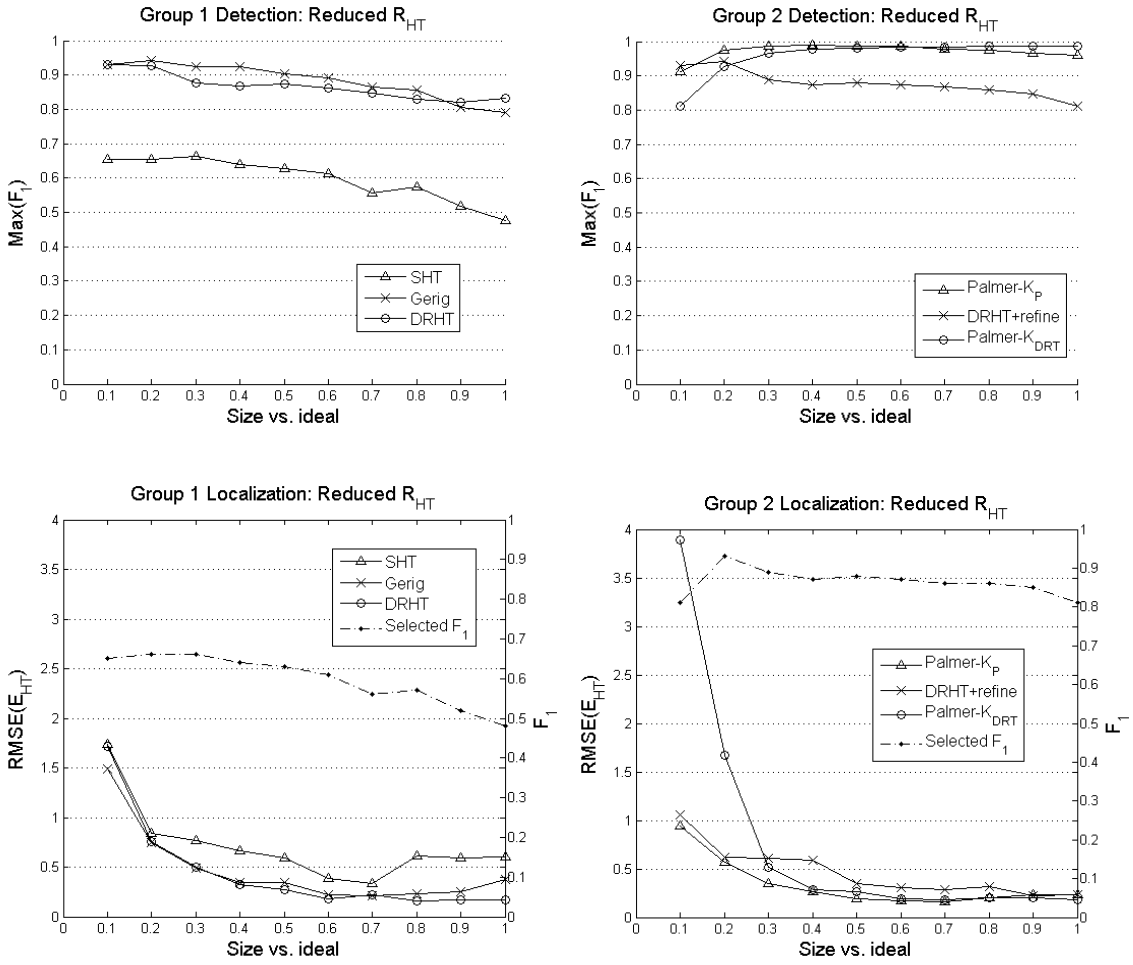


Figure 4-5: Hough transform line detection and localization results when accumulator size R_{HT} is reduced and image size R_I is maintained at full resolution.

4.3 Conclusions

DRT improves line detection performance when used to extend the Hough transform. It improves peak localization in Gerig and Klein's re-voting scheme when image and accumulator size are simultaneously reduced and (more significantly) when only the image size is reduced. The new DRHT achieves localization error of less than twice its full-resolution performance on images as small as 40% of full resolution. DRT also improves detection and localization performance at reduced resolutions when it replaces the voting kernel used in the extension of Palmer, Kittler, and Petrou. This is seen particularly in images at 70% resolution or less.

Chapter 5.

Distributed Ray Tracing Applied to Multi-View Stereo

In this chapter I present a dense multi-view stereo reconstruction algorithm based on Hough transform (HT) concepts and techniques. The algorithm is then extended with distributed ray tracing as in the DRHT from Chapter 4. I then analyze the performance of the DRT-enhanced algorithm on fundamental and complex scenes. This work is also submitted for publication as [22].

Distributed ray tracing (DRT) is used often as an anti-aliasing technique in computer graphics, but it has not been widely used in computational stereo algorithms that also suffer from aliasing. I demonstrate that DRT can improve multi-view stereo results, particularly when performance is limited by the sampling intervals of input imagery or internal representations. This example provides insight into how DRT could be applied to improve other computational stereo algorithms.

5.1 Approach

5.1.1 SHT-Based Dense Multi-View Stereo

I begin by applying SHT in a dense multi-view stereo reconstruction approach, in

three stages: wireframe, sparse-to-dense, and hole filling. Each stage fuses and interprets evidence using HT principles, sometimes augmented with other techniques.

The input is a set of images $I_m(i, j)$ with camera parameters C_m, and an optional Boolean voxel model $H(x, y, z)$ that defines a priori bounds on the scene. H is typically the visual hull, constructed from foreground masks as in [106].

The first stage estimates a wireframe model. It is illustrated in Figure 5-1. Edges are detected in each $I_m(i, j)$ to form binary feature images $F_m(i, j)$. Each feature pixel (i', j') in F_m contributes evidence that the scene is occupied at some location that projects to (i', j') at viewpoint C_m. SHT traverses a ray $\vec{r}(i', j')$ from the origin of C_m through the center of (i', j') to accumulate counts in a voxel model A_w. The step length along $\vec{r}$ is equal to the voxel side length v_{HT}. A_w is the Hough accumulator and the rays $\vec{r}$ are trajectories in Hough space. As in other applications, trajectories converge on correct Hough parameterizations. Finally, A_w is thresholded at a level t_w to define the wireframe model S_w.

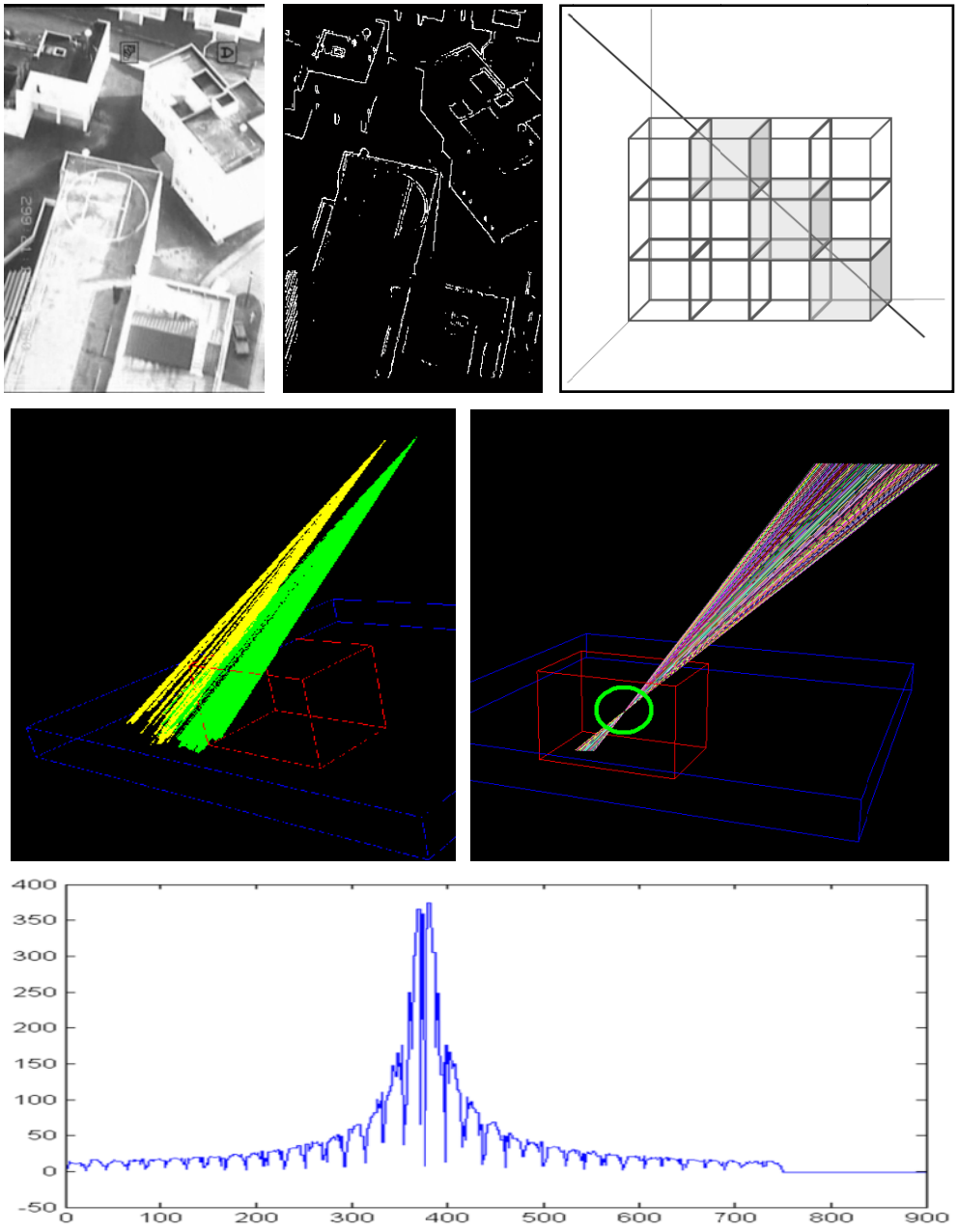


Figure 5-1: Wireframe modeling stage of SHT-based dense multi-view stereo. (Top row) (Left) Sample input image. (Center) Input images are edge-detected. (Right) Each edge pixel defines a ray through the accumulator voxel model A_w. (Middle row) (Left) Rays are cast for edge pixels through the accumulator from multiple viewpoints (shown here in yellow and green). (Right) Edge rays converge from many viewpoints at each occupied voxel. (Bottom row) Convergence causes peaks in A_w along each ray's trajectory.

The second stage uses S_w and a set of F_m to accumulate and interpret dense evidence, in the same spirit that stage one did for sparse evidence. S_w is rendered from each C_m to generate a wireframe range image $D_m^{S_w}(i,j)$. A Delaunay triangulation is computed from $\left\{D_m^{S_w}\right\}$, the set of all pixel indices in $D_m^{S_w}$ with assigned ranges. A dense range estimate is computed for each pixel in the convex hull of $\left\{D_m^{S_w}\right\}$ by interpolating the sparse ranges at its enclosing triangle's vertices. This defines the dense range image $D_m(i,j)$.

It is important that only feature pixels in F_m are given ranges in $D_m^{S_w}$ in order to prevent occluded objects from contributing false values. In practice, F_m is dilated by one pixel to account for calibration error.

Dense ranges are refined towards the camera following [112] to mitigate feature detection failures in, F_m which distinguishes ranges in $D_m^{S_w}$ that reflect concavities from ranges that should be occluded in a dense model. $D_m^{S_w}$ and D_m are iteratively updated with voxels in S_w that are closer to the camera than D_m. These voxels indicate feature detection failures in F_m. Well-known incremental Delaunay algorithms can be used, and a required ordering is described in [112].

A dense evidence accumulator A_d is incremented for each range estimate in each D_m. Stage two ends by thresholding A_d at t_d to define a dense occupancy model S_d.

The third stage reduces holes in S_d. Stage three largely repeats stage two, but starting from S_d instead of S_w. Range images $D_m^{S_d}$ are rendered from each C_m. *Hole pixels* are defined as interior foreground pixels without ranges in $D_m^{S_d}$. I then mask $D_m^{S_d}$ to define ranges only at pixels surrounding hole pixels, which improves runtime significantly without changing the results. $D_m^{S_d}$ is interpolated as in stage two to define ranges $D_m(i,j)$ at the hole pixels. An accumulator A_{hf} is incremented for each range in each D_m. Since the goal is to fill small holes, A_{hf} is always thresholded at t_{hf}=1. Stage three thresholds A_{hf} to define S_{hf} and combines it with S_d to form the final model S.

5.1.2 Extension with Re-accumulation

The SHT stereo approach is extended using re-accumulations analogous to those of [39]. Each of the three SHT stages is augmented by re-accumulation after A_w, A_d, or A_{hf} – collectively referred to as A – is computed. The rays $\vec{r}(i,j)$ from each C_m, first used to create A, are traversed again. The evidence in A at each step on $\vec{r}$ defines $e_{\vec{r}}(d)$ as a function of the distance d along $\vec{r}$. Figure 5-2 shows an example $e_{\vec{r}}(d)$ along a ray passing through A_w, relative to the ideal $e_{\vec{r}}^*(d)$ (discussed in Section 5.2).

The largest peak in $e_{\vec{r}}(d)$ is identified and the voxel corresponding to its distance d^* is incremented in a second accumulator B_w, B_d, or B_{hf}. After all viewpoints are processed, the thresholds, t_w, t_d, and t_{hf}, are applied to B instead of A to define occupancy models S_w, S_d, and S_{hf}.

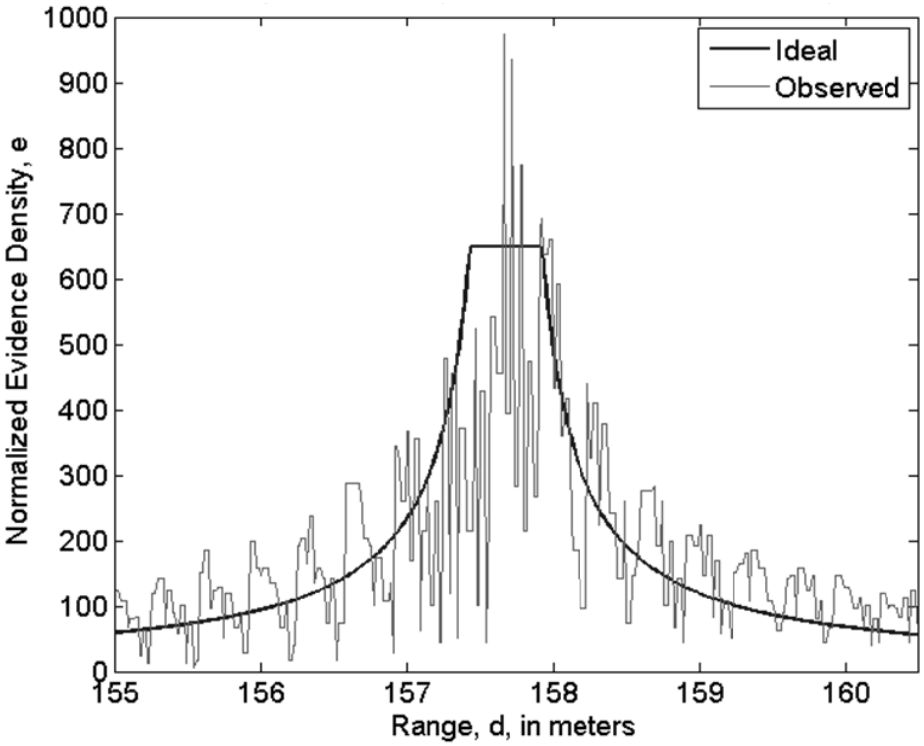


Figure 5-2: Accumulated sparse evidence signal extracted from A_w along an example ray, after ray casting with the standard Hough transform (SHT).

The peak distance d^* is computed from the envelope of $e_{\bar{r}}(d)$. Following a common technique, the analytic signal $s_{\bar{r}}(d) = e_{\bar{r}}(d) + iH\{e_{\bar{r}}(d)\}$ is defined using the Hilbert transform $H\{\cdot\}$, and the envelope is estimated from a low-pass filtered $s_{\bar{r}}(d)$. Choosing d^* from the envelope helps mitigate the aliasing that is clearly visible in Figure 5-2.

This is different from the re-accumulations from [39], [41], and [63], which simply select

$$d^* = \arg\max e_{\vec{r}}(d)\,. \tag{24}$$

It is also unlike our work in [20], which used application-specific morphological operators and restrictions on the shape of $e_{\vec{r}}(d)$ (e.g., minimum peak-to-mean ratio). Excluding application-specific techniques helps to isolate and quantify the benefits of the re-accumulation.

5.1.3 Extension with Distributed Ray Tracing

The SHT stereo approach is now extended with DRT instead of re-accumulation. Each stage uses only one accumulation, and thresholds are applied to A, as in SHT. Accumulation is modified in three ways.

First, multiple rays are cast for each feature in F_m (stage one) or dense range estimate in D_m (stages two and three). Each $\vec{r}(i, j)$ is replaced with rays $\vec{r}_k(i + \Delta_i, j + \Delta_j)$ with slightly perturbed trajectories. The experiments use **U**(-0.5, 0.5) for $P(\Delta_x)$ and $P(\Delta_y)$, so the rays for a feature or dense range estimate cover the cross-range span of its pixel, as depicted in Figure 5-3.

Second, the rays in stage one are traversed through A_w in steps smaller than voxel side length v_{HT}. The experiments used a step length of $\frac{1}{3} v_{HT}$. This yields larger accumulations for rays passing through the center of a voxel than for slight intersections.

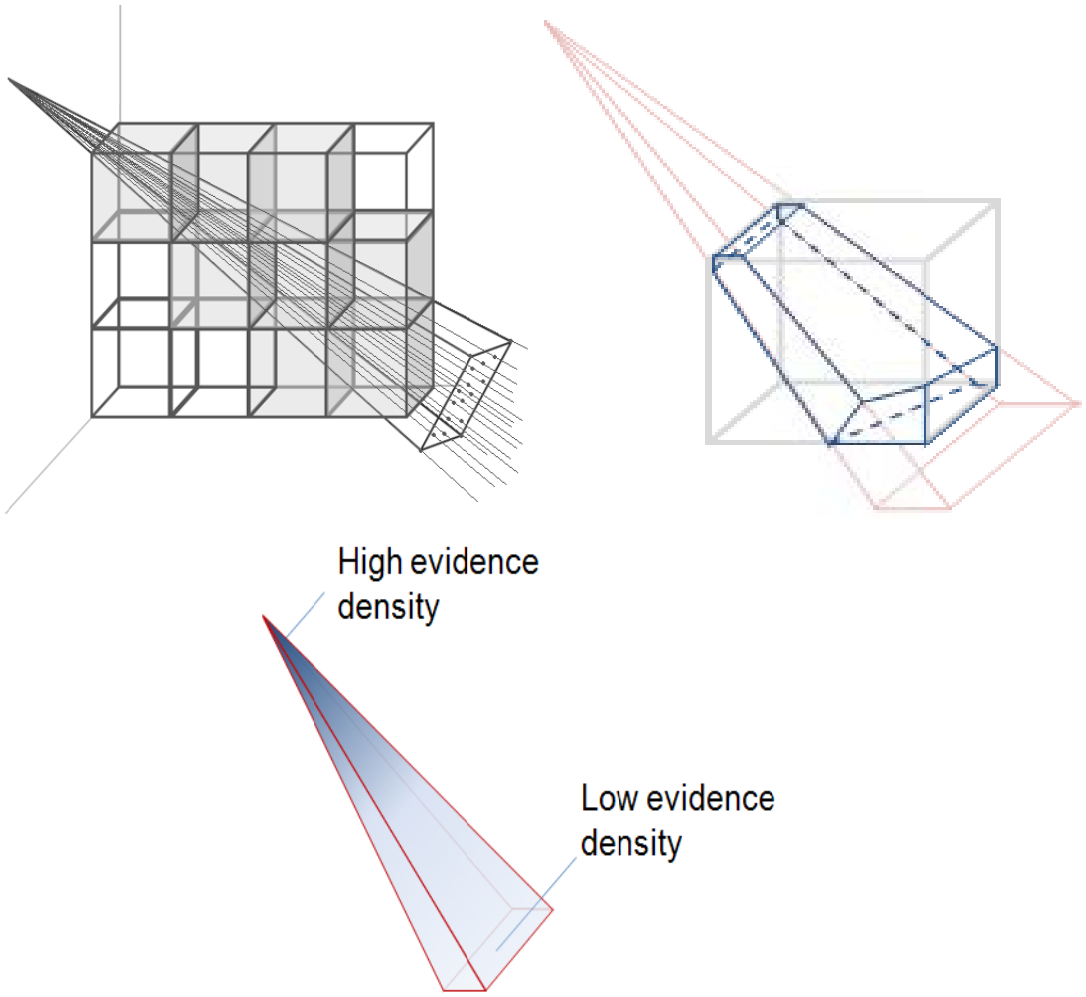


Figure 5-3: Sparse evidence accumulation for object-centric multi-view stereo in the style of the distributed ray Hough transform. DRT is used to approximate evidence density and evidence mass intersection calculations. (Top left) The spatial extents of a pixel's evidence are modeled by casting multiple perturbed rays. Voxels with which the pixel intersects are highlighted. (Top right) Complex intersection boundary between a pixel's spatial extents and a single voxel. (Bottom) Decrease of evidence density at increasing range under the model that each pixel cross-section contains equal evidence.

Finally, distances along the rays are slightly perturbed. The i^{th} accumulation on sparse feature ray $\vec{r}_k$ occurs at distance

$$d_{ki} = \frac{1}{3} v_{HT} \left(i + \Delta_{ki} \right), \tag{25}$$

with Δ_{ki} drawn from **U**(-0.5, 0.5). The locations of accumulations into *B* towards finding dense range estimates are perturbed similarly. This prevents accumulations from forming plane waves (constrained to the cross-range extents of their pixel) within the accumulators.

Figure 5-3 also illustrates the conceptual changes in evidence accumulation resulting from the use of DRT. DRT represents each pixel as a pyramid and each dense range estimate as a frustum. Cross-range extent grows with increasing range and each cross section is modeled to contain equal evidence. As a result, *evidence density* decreases with increasing range. Accumulation approximates the integral of evidence density over the intersection of pyramid (or frustum) and voxel. Accumulator values approximate *evidence mass*.

Analytical computations of these values would be complex and expensive, requiring solid geometry intersections, boundary value calculations, and 3D integration over irregular volumes. The values are instead approximated efficiently and simply using DRT. Selecting the number of rays and step length provides an explicit tradeoff between approximation quality and runtime.

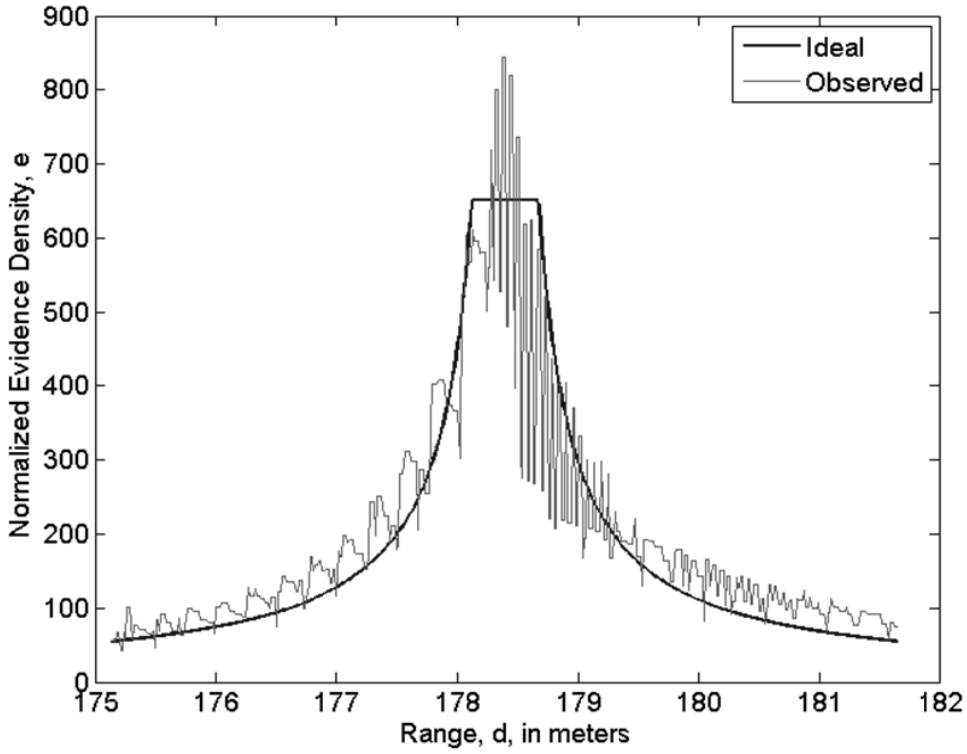


Figure 5-4: Accumulated sparse evidence signal extracted from A_w along an example ray, after ray casting with the distributed ray Hough transform (DRHT), reduced step length, and perturbed step lengths. The techniques mitigate aliasing in the signal, as evidenced by the improved match between the observed and the ideal signal (discussed in Section 5.2), and more accurate peak location.

Figure 5-4 shows an example evidence signal from a DRT accumulator along the same trajectory shown for SHT accumulation in Figure 5-2. DRT clearly mitigates aliasing in the observed signal. As demonstrated below, this enables more accurate HT-based stereo reconstruction.

5.2 Analysis on a Fundamental Scene

To examine the benefits of DRT, an ideal sparse evidence signal $e^*_{\vec{r}}(d)$ is defined for an abstract scene and compared with observed signals $e_{\vec{r}}(d)$.

Consider a scene with a single point object at location *p* that is at a distance d_p from a linear camera path of length L_0 that is symmetric about *p*. This scenario is illustrated in Figure 5-5. The ideal evidence density after accumulation, $e^*_{\vec{r}}(d)$, can be defined at distance *d* on ray $\vec{r}$.

Density on the camera path is defined arbitrarily as 1.0 m^{-1}. Density grows with $(d_p\text{-}d)^{-1}$ as the total evidence L_0 is compressed over a shorter linear distance *L(d)*. As *d* nears d_p, $e^*_{\vec{r}}(d)$ increases to infinity. This is undesirable, so $e^*_{\vec{r}}(d)$ is limited to e_{max}. The ideal density $e^*_{\vec{r}}(d)$ at distance *d* on $\vec{r}$ is thus

$$e^*_{\vec{r}}(d) = \min\left\{\frac{L_0}{L(d)}, e_{\max}\right\} = \min\left\{\frac{d_p}{d_p - d}, e_{\max}\right\}. \quad (26)$$

A characteristic ideal evidence density signal is shown in Figure 5-6. Ideal density signals are also shown in Figure 5-2 and Figure 5-4 for their respective scenes. The voxel size, L_0, and other factors affect the correct choice for e_{max}. Here, e_{max} was set empirically.

The quality of an observed $e_{\vec{r}}(d)$ with and without DRT is measured relative to $e^*_{\vec{r}}(d)$ using signal-to-noise ratio (SNR). SNR is defined in decibels as the ratio between

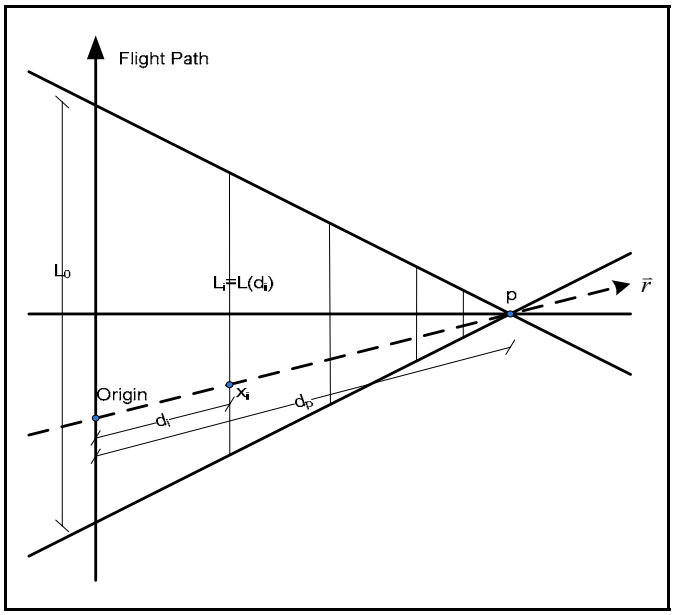


Figure 5-5: Idealized modeling scenario in which a point object p is viewed from a symmetric and linear flight path. Ideal evidence density signal $e^*_{\vec{r}}(d)$ along ray $\vec{r}$ is defined for the purpose of measuring observed signal quality.

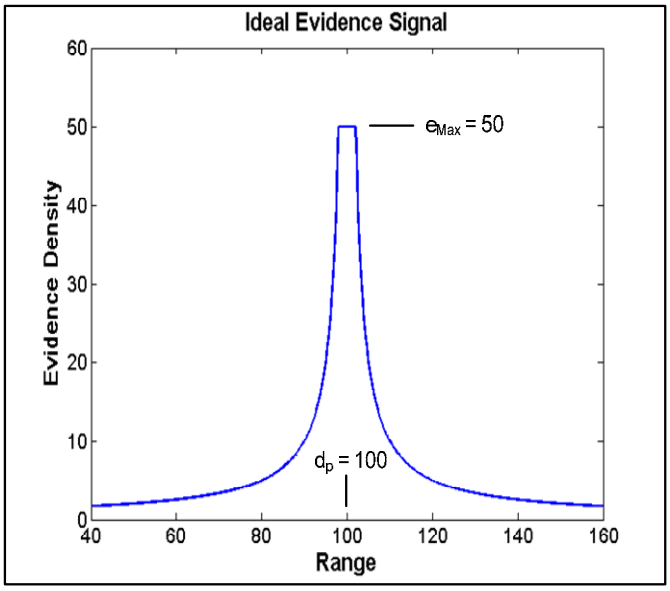


Figure 5-6: Example ideal evidence signal for point object p at distance d_p = 100m from the flight path, and e_{max} = 50.

the power in the ideal signal and the power in the noise (observed signal minus ideal signal), as

$$SNR(e) = 10\log 10\left(\frac{\sum_i \left(e^*(d)\right)^2}{\sum_i \left(e(d) - e^*(d)\right)^2}\right). \quad (27)$$

The subscripts on $e_{\vec{r}}(d)$ and $e^*_{\vec{r}}(d)$ are removed in (27) for clarity. An error-free signal has $SNR = \infty$.

To approximate a remote sensing task, evidence from 200 images at a resolution R_I = (720,480) was fused from viewpoints along a 75 m camera path at ranges of 150-220 m to the point object, to approximate a remote sensing task. The sensor's nominal ground sample distance (GSD) was 0.08 m. The region around the object was divided into voxels with v_{HT} = 0.04 m. This resulted in significant aliasing as the voxel size is smaller than the GSD.

Results are shown in Figure 5-7 for Δ_i and Δ_j drawn from **U**(-0.5,0.5). For SHT, the measured evidence signals have an average SNR of 7.7dB relative to $e^*_{\vec{r}}(d)$. Using DRT for accumulation into A_w increases the SNR to 10.0dB. Using DRT on both A_w and B_w increases the SNR to as much as 13.0dB. Most benefits are realized using only 50-100 rays.

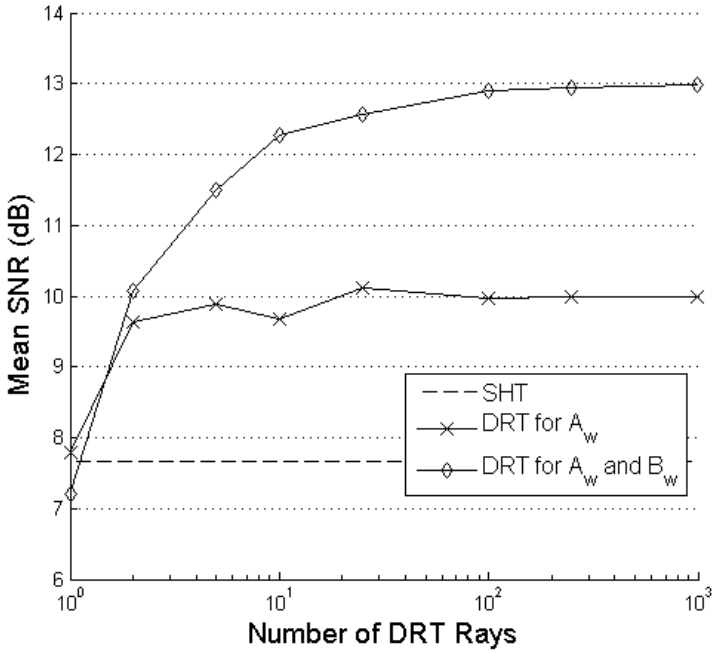


Figure 5-7: Comparisons of observed evidence signals $e_{\vec{r}}(d)$ to ideal signal $e^*_{\vec{r}}(d)$ under different applications of DRT, measured via signal to noise ratio. The use of DRT improves the match of observed to ideal signal over accumulation with SHT, when applied to accumulation (A_w), and improves it further when applied to both accumulation and re-accumulation (B_w).

Other perturbation distributions were tested in [21]. Most choices improved performance relative to SHT. The best performance occurs with modest perturbations, but does require parameter tuning. Performance degrades as perturbation magnitude becomes large.

5.3 Analysis on a Multi-View Stereo Benchmark

DRT was shown above to improve evidence fusion quality on trivial scenes. I now analyze its benefits on more complex and realistic scenes using a well-known benchmark dataset from [106]. Performance is compared to a baseline approach and also to HT-based multi-view modeling without DRT.

5.3.1 Experiment Design

Performance is measured on the 'dinoRing' dataset from [106]. The dataset contains 48 color images of a dinosaur figurine in a controlled environment, taken at R_I = (480,640) from viewpoints on a circle. Examples are shown in Figure 5-8. Camera calibrations are provided with the data.

The datasets of [106] show only small near-field foreground objects in tightly controlled and well-lit scenes. They do not provide the variety required to evaluate an algorithm's effectiveness on uncontrolled natural scenes. However, the data is widely used and accurate calibrations are available, making it one of the best benchmarks at this time. Many of the top performers in [106] compute and then refine a visual hull of the scenes. This strategy is also used in the experiments below.

Thirty-six (36) images are used for modeling and 12 are withheld for evaluation. Evaluation images are interleaved among the modeling images. Algorithms process the modeling images I_m, their camera parameters C_m, and a hull estimate H to produce dense

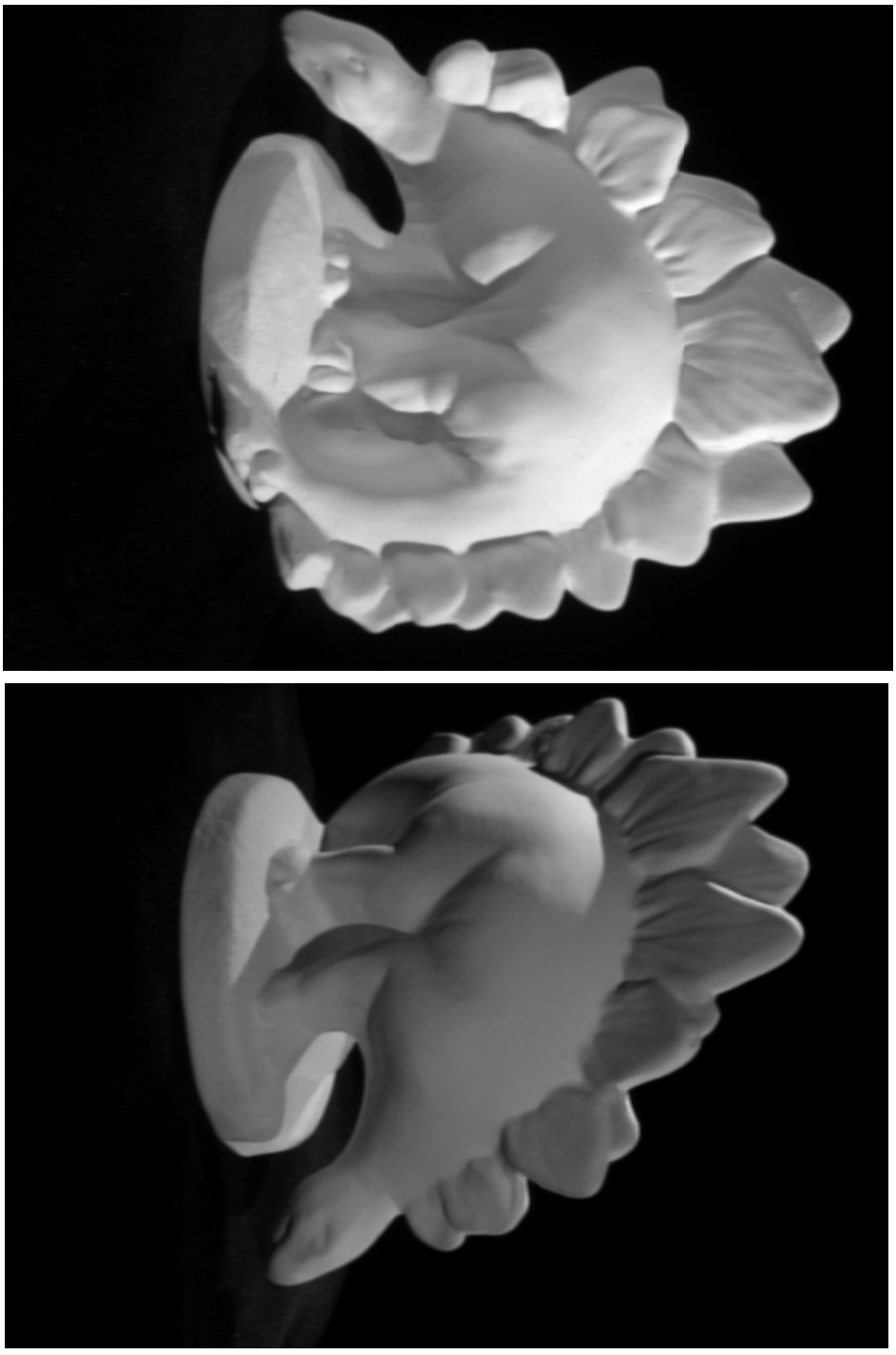


Figure 5-8: Sample input images from the ‘dinoRing’ dataset from [106].

structure estimate *S*. Performance is measured via re-projection error on the evaluation images.

Performance is tested on full- and reduced-resolution inputs, from 100% to 10% of the original resolution R_I. Images are reduced using Matlab's *imresize* function with its default bicubic interpolation and anti-aliasing. C_m are constructed for reduced R_I by holding all parameters constant except resolution and principal point; resolution is manually reduced and principal point is converted to its equivalent at reduced R_I.

The hull estimate H is the intersection of the projections of all foreground masks through the modeling region. Following [106], foreground masks are defined as

$$FG_m(i,j) = ERODE\ (DILATE\ (THRESHOLD\ \ (I_m(i,j), t_{FG}), D_{FG}), E_{FG}). \qquad (28)$$

At full resolution, D_{FG} = 10 pixels and E_{FG} = 7 pixels. The value t_{FG} is selected empirically. Importantly, H is built using only the modeling images, and it is built from reduced-resolution images in reduced-resolution tests. D_{FG} and E_{FG} are scaled linearly with resolution and then rounded up and down, respectively. Their values are given in Table 5-1.

5.3.2 Performance Metrics

The metrics used in [106] require unpublished ground truth and manual steps, so results must be submitted to the authors for evaluation. This makes it difficult to compare multiple approaches that are not already published on their website.

R_I (%)	100	90	80	70	60	50	45	40	35	30	25	20	10
D_{FG}	10	9	8	7	6	5	5	4	4	3	3	2	1
E_{FG}	7	6	5	4	4	3	3	2	2	2	1	1	0

Table 5-1: Morphology parameters applied to foreground masks when constructing visual hull H at different input image resolutions. Resolution R_I is specified as a percentage of original (480x640 pixels). Dilation (D_{FG}) and erosion (E_{FG}) parameters are given as side length in pixels for a square kernel.

Re-projection error does not have this limitation. Re-projection error is the difference between a captured image and an image generated by rendering the model S from the same viewpoint. It has been used in many previous evaluations. Re-projection error is characterized with an F_1 metric and a modified Structural SIMilarity (SSIM) index designed to assess the quality of range images, known as Range SSIM or R-SSIM.

The re-projection error is defined for captured image I_m and rendered image $\hat{I}_m$ via an *ideal* foreground mask FG_m^*. FG_m^* is the projection of the ideal visual hull H^* onto viewpoint C_m. Unlike the hull estimate H used in modeling, the ideal hull H^* is created from all available full-resolution images (modeling and evaluation) and is the same for every experiment.

F_1 is the geometric mean of *recall* and *precision*. *Recall* is the percentage of pixels in FG_m^* that are given estimated grayscale values in $\hat{I}_m$. *Precision* is the percentage of estimated pixels that are considered good. Estimated pixels within FG_m^*

are good if their value is within a threshold t_P of the reference value in I_m, and bad otherwise. All estimated pixels outside FG_m^* are bad. The experiments used t_P=15, which was set by inspection. The F_1 results are averaged over all views.

The SSIM index is a perceptual image quality index that correlates well with human subjective evaluation of image quality [117]. It combines factors for luminance, contrast, and local structure into a scalar quality value at each pixel. It has been extended in numerous ways [107]. This analysis uses the original index with the R-SSIM extension from [79] that is intended to assess the quality of range images, and that addresses missing pixels in I_m or $\hat{I}_m$. R-SSIM results are averaged over all pixels within the masks FG_m^* or with estimated values in $\hat{I}_m$.

Rendering $\hat{I}_m$ requires mapping a grayscale value (hereafter a "color") to every visible voxel in *S*. Because the goal is to measure the accuracy of the structure of *S*, a common coloring algorithm is shared among all the approaches. Colors are assigned by projecting the center of each voxel face onto to each captured image I_m. A face is ignored if it is occluded or has a backwards-facing normal. Voxels are assigned the average grayscale of all their faces' unoccluded projections. Nearby voxels without colors (distance 2 or less) are assigned the average value of their local neighbors.

Models are always colored using the 36 full resolution input images. Metrics and indices are always computed using the 12 full resolution evaluation images. This follows

from the underlying goal of evaluating structure accuracy rather than view synthesis for its own sake.

5.3.3 Results and Discussion

Four approaches are compared: the reduced-R_I hull H ("Hull"), HT-based modeling with the SHT as in Section 5.1.1 ("SHT"), HT-based modeling with re-accumulation as in Section 5.1.2 ("Gerig"), and HT-based modeling with DRT as in Section 5.1.3 ("DRT"). Comparing to Hull gives context, and comparing to SHT clarifies the effects of the extensions.

All approaches use $v_{HT} = 2\text{x}10^{-4}$m. No parameters exist to vary in the Hull approach. In the other approaches, the thresholds t_w and t_d vary freely; results are reported for the best-performing values. DRT uses 100 rays for wireframes, 50 for sparse-to-dense, and 40 for hole filling.

F_1 and R-SSIM results are given in Figure 5-9. DRT is the top performer in almost all cases. It captures more structural detail and is more robust to reduced R_I. At resolutions of 80% and above, the difference between DRT, Gerig, and SHT is small in terms of F_1 but significant with respect to R-SSIM. Figure 5-10 and Figure 5-11 show example models at 100% resolution. DRT is more robust than SHT and Gerig at 50% resolution and below, so the gap widens significantly using both F_1 and R-SSIM. Figure 5-12 and Figure 5-13 show example models from the four approaches at 40% resolution.

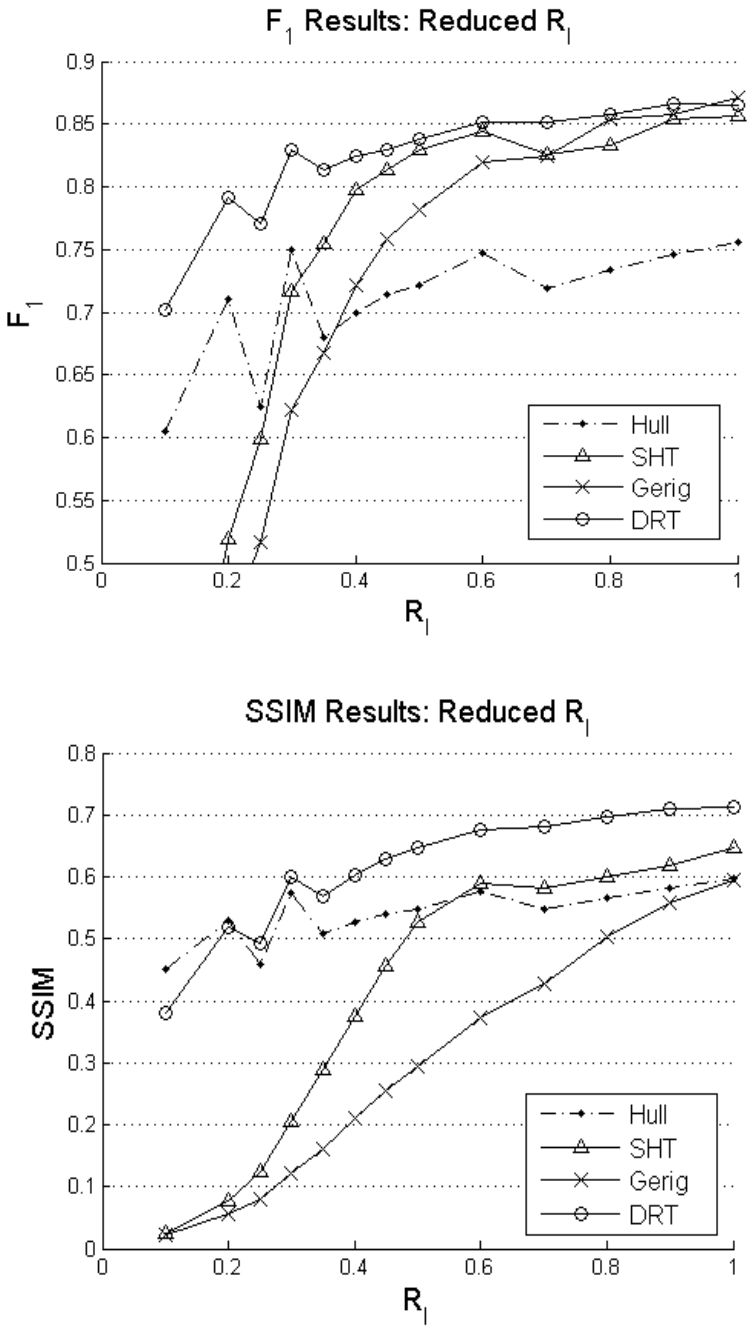


Figure 5-9: Reconstruction performance on ‘dinoRing’ for the visual hull (‘Hull’), HT-based reconstruction with standard Hough Transform (‘SHT’), with re-accumulations as in [39] (‘Gerig’), and with distributed ray tracing (‘DRT’). Results are presented for re-projection error measured by F_1 score (top) and a modified SSIM metric (bottom).

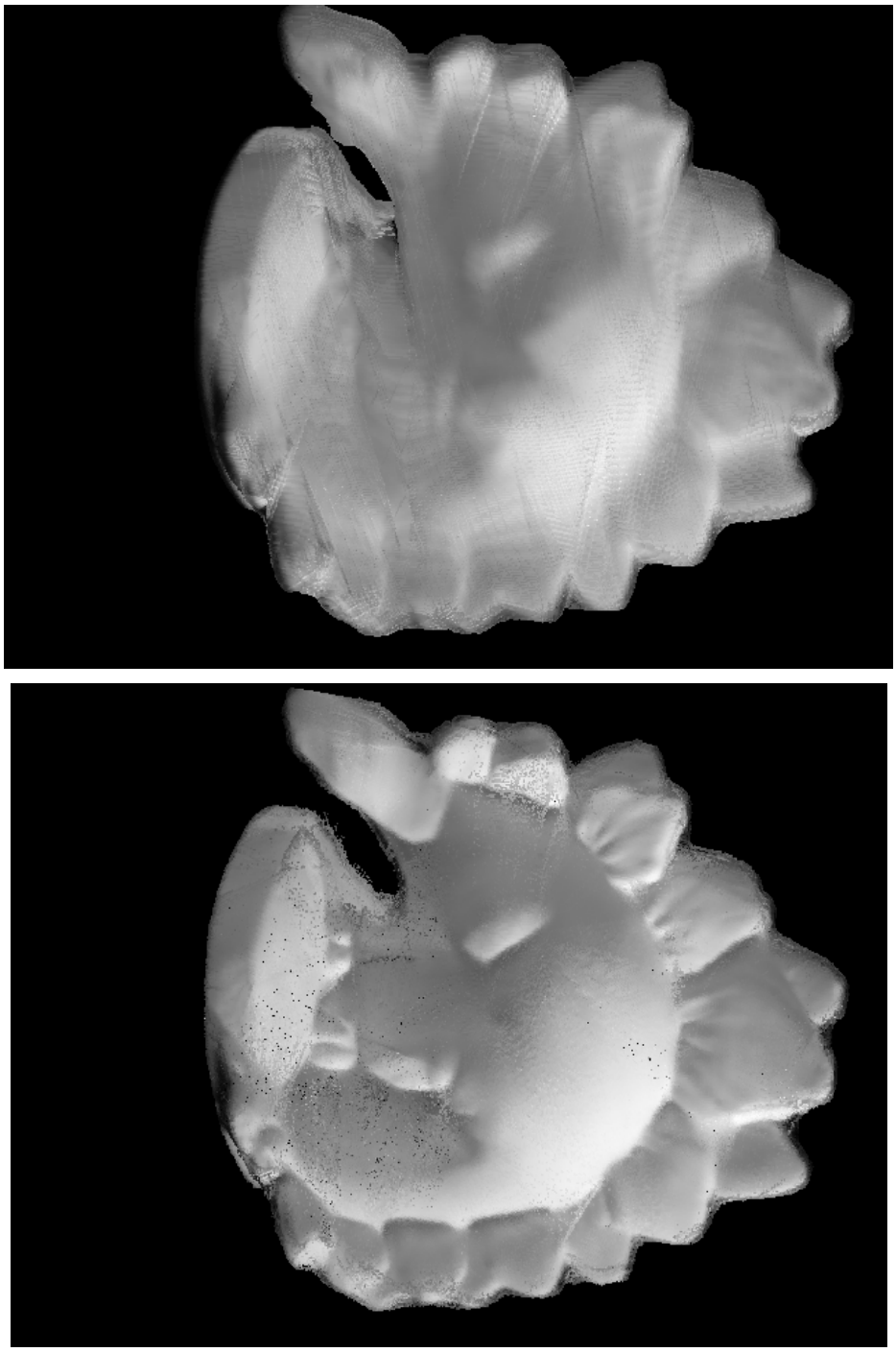


Figure 5-10: Reconstruction from 100% resolution input by Hull (top) and SHT (bottom).

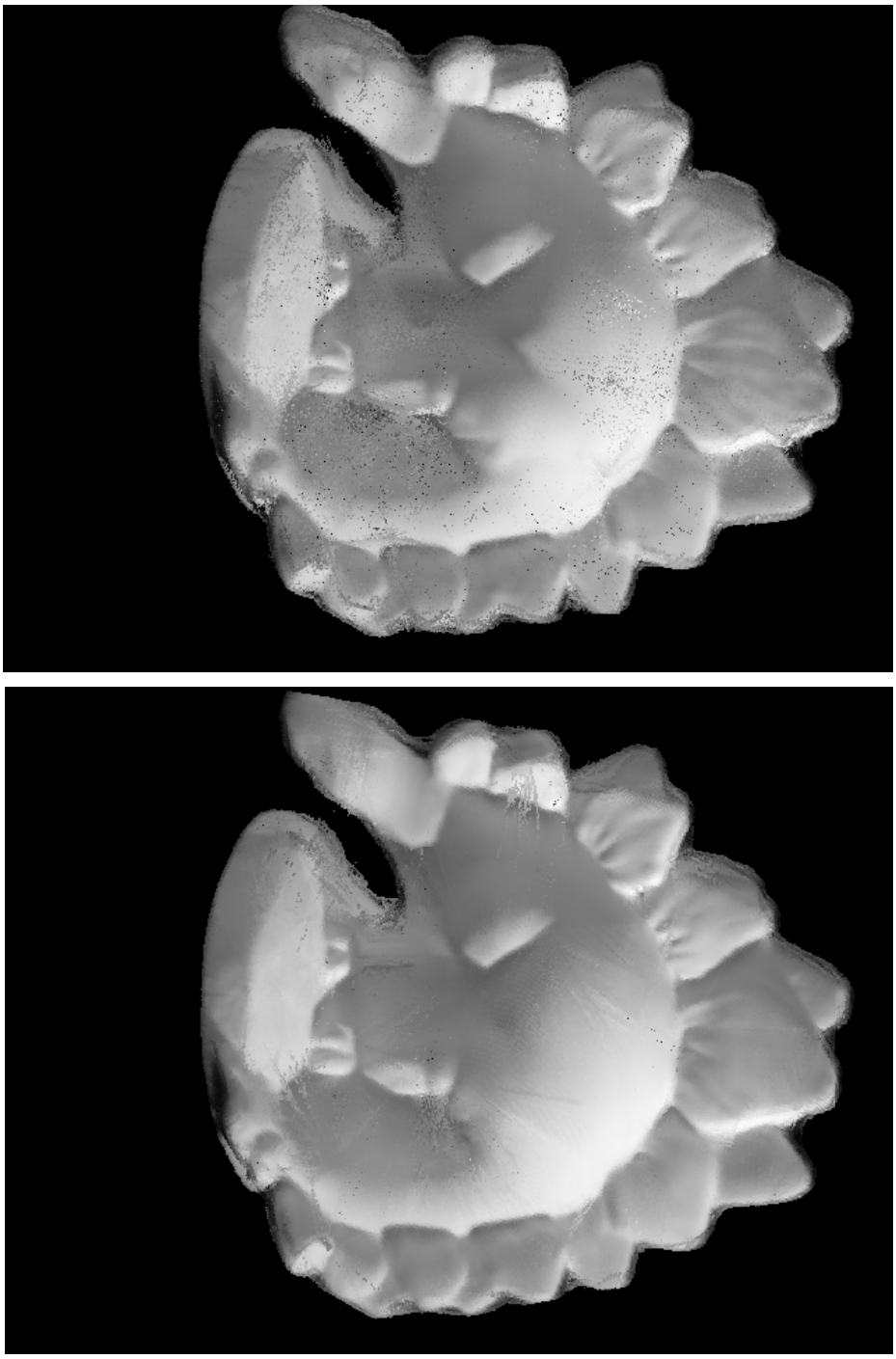


Figure 5-11: Reconstruction from 100% resolution input by Gerig (top) and DRT (bottom).

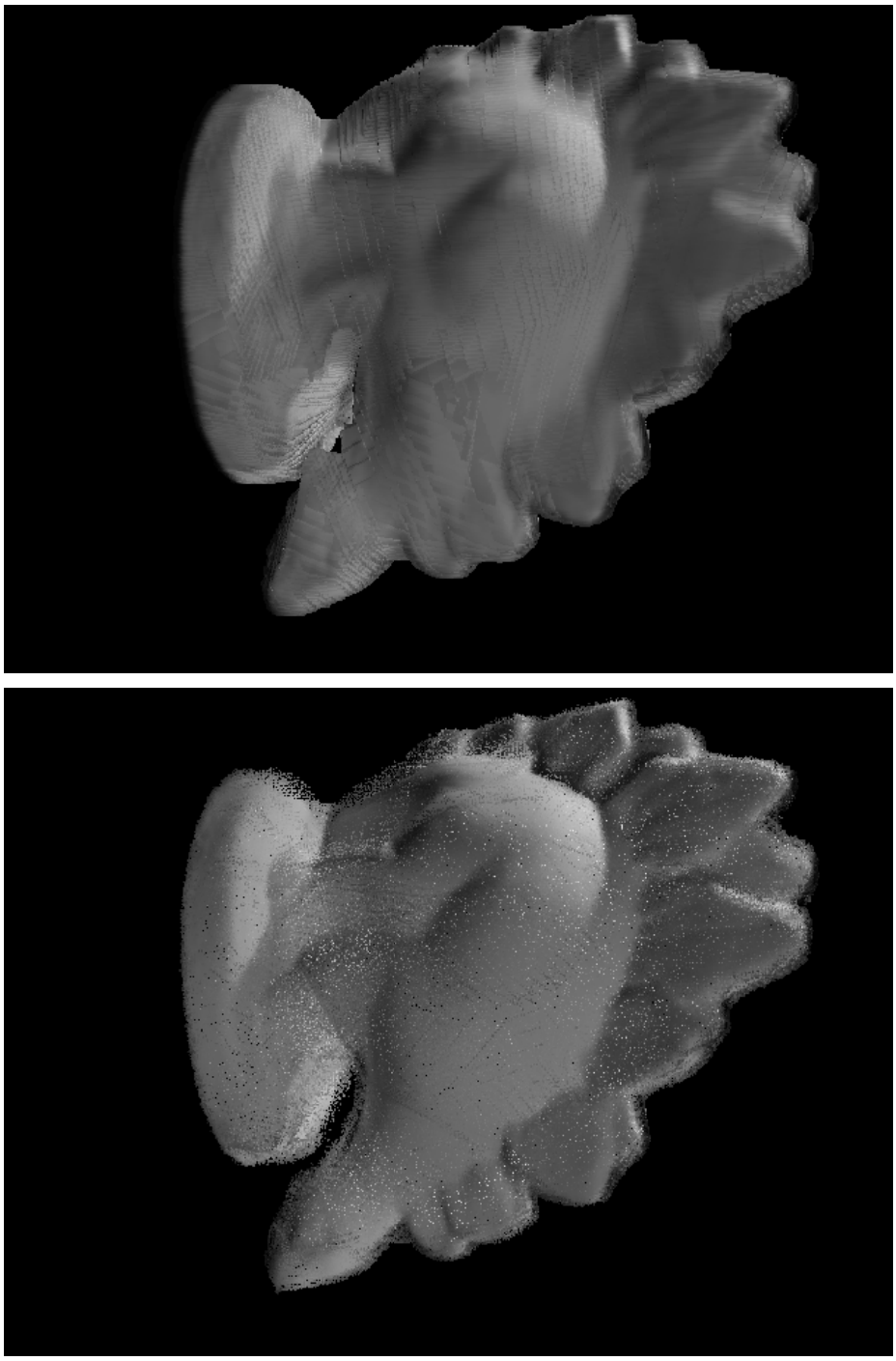


Figure 5-12: Reconstruction from 40% resolution input by Hull (top) and SHT (bottom).

Figure 5-13: Reconstruction from 40% resolution input by Gerig (top) and DRT (bottom).

There are two minor exceptions to DRT's dominance. At 100% resolution, Gerig slightly outperforms DRT using F_1. Hull outperforms DRT slightly on R-SSIM at 20% resolution, and more significantly at 10% resolution.

SHT and Gerig results are mixed relative to Hull and to each other. Both perform well at high R_I but degrade significantly at low R_I. SHT and Gerig outperform Hull in terms of F_1 only at resolutions above 35%. SHT outperforms Hull using R-SSIM at resolutions of 60% and above. Gerig never outperforms Hull using R-SSIM, and reaches equivalence only at 100% resolution. SHT outperforms Gerig in terms of F_1 for resolutions 60% and below. SHT outperforms Gerig with R-SSIM at all resolutions, but most significantly at 35-70%.

Figure 5-13 shows that at low R_I, DRT maintains detail in S and prevents holes. It preserves detail by mitigating aliasing when R_I and v_{HT} are mismatched. SHT and Gerig are less effective at mitigating aliasing, as shown in Figure 5-12 and Figure 5-13, respectively. Reduced aliasing also allows DRT to outperform at high R_I, albeit less significantly. DRT avoids holes by improving sparse-to-dense modeling and hole filling. Holes appear in SHT and Gerig at low R_I despite the explicit hole filling stage.

Hull avoids holes by construction, but its performance degrades at low R_I because coarser foreground masks cause a loss of detail. Its performance varies widely at 35% resolution and below, where the accuracy of H is strongly affected by quantization of D_{FG} and E_{FG}. Because the HT variations start from H, DRT performance also gyrates at low

R_I. SHT and Gerig perform poorly enough that the effect is masked, although it can be seen slightly in their results near 70%.

R-SSIM and F_1 are both sensitive to the level of detail in S, but R-SSIM is more sensitive to small holes than is F_1. Voxels from the back surface of S often show through the holes, and in this scene the surfaces are lit differently. As a result, holes manifest as isolated bright pixels in a smooth dark region, or isolated dark pixels in a smooth bright region. As shown in Figure 5-14, this causes a single bad pixel for F_1 but it depresses the R-SSIM results over a larger local region.

DRT dominance is thus more universal in the sense of R-SSIM than F_1 since it reduces holes relative to SHT and Gerig. Similarly, relative Hull performance is better with R-SSIM than F_1 because it never has holes. Relative to SHT, Gerig increases detail but does not reduce holes. This helps explain why Gerig outperforms SHT in some cases using F_1 but never using R-SSIM.

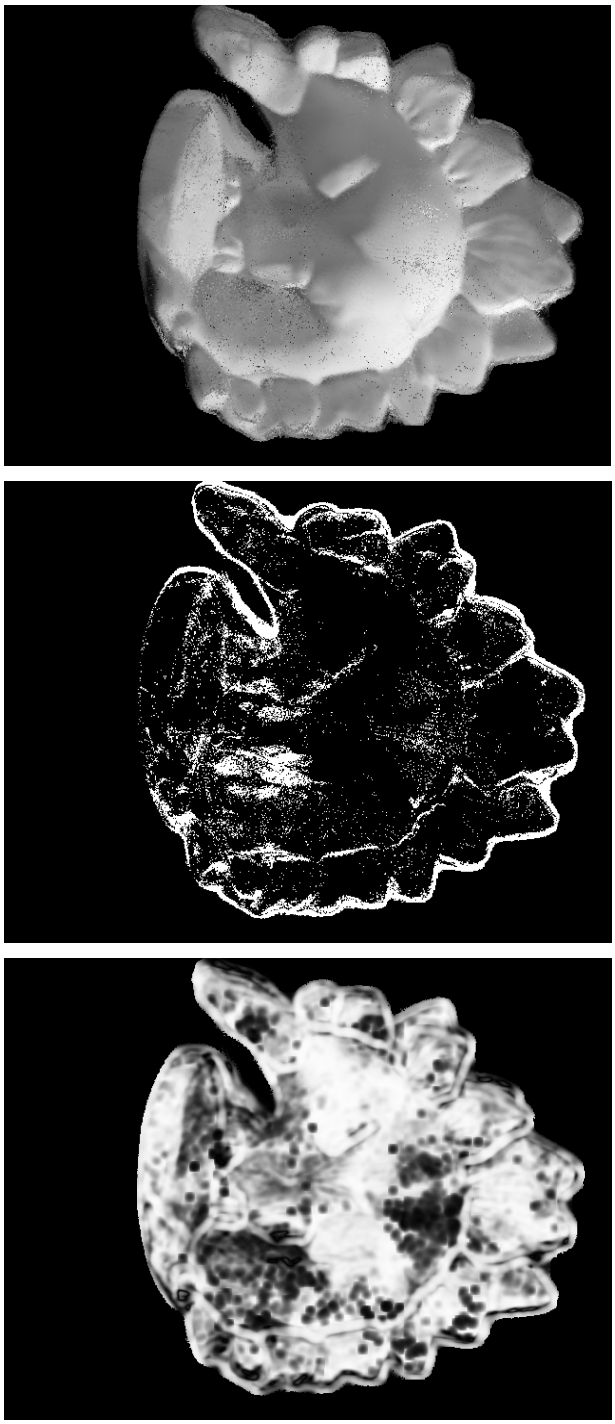


Figure 5-14: (From top) Sample DRT reconstruction, F_1 results (bad pixels), and SSIM results.

5.4 Conclusions

DRT can also improve reconstruction performance in multi-view stereo. In a wireframe-to-dense approach based on the Hough transform, it improves performance over both the standard Hough transform and extensions to the approach based on re-accumulation.

In a fundamental modeling task, using DRT improves the match between the ideal evidence density signal and the computed evidence density signal along a trajectory in the Hough domain. Aliasing is mitigated and a measurable improvement in signal-to-noise ratio is observed.

In a complex modeling task, HT-based modeling using DRT outperforms other HT variations. DRT outperforms at all resolutions because it mitigates HT-domain aliasing to capture more structural detail. It outperforms more strongly at lower image resolutions where it also prevents holes from forming in the reconstructed model.

Although achievable camera resolutions are increasing rapidly, the desire to generate models with voxel sizes smaller than the sample distance of the camera will remain. Scenarios in which this is required include urban modeling from aerial platforms at “standoff” distances (e.g., 20,000 feet or more), and wide-area modeling from many small platforms with low-resolution cameras. These are the scenarios in which DRT provides the most significant benefits.

Chapter 6.

Conclusions and Future Work

6.1 Conclusions

The experimental results and analysis above support conclusions ranging from very specific to very general.

Quality-efficient stochastic sampling (QUESS) is shown to be an effective stereo reconstruction algorithm. It outperforms the best-known existing example of a stochastic stereo algorithm [7] and also a popular cooperative approach [124] on two-frame and monocular video reconstruction tasks. Accurate depth estimates are generated with fewer evaluations of the local match quality function through the use of stochastic sampling.

Many of the conclusions and comparisons from the QUESS experiments generalize beyond just that evaluation. ZK is representative of many approaches that begin by exhaustively sampling the local match quality function, which will have similar requirements for computing and storing those values. In comparison, stochastic sampling gives advantages to runtime and memory, and to the insulation of runtime and memory from stereo baseline, camera path, scene orientation, and scene structure. It allows more attractive scaling with respect to image and depth resolution than approaches that rely on

exhaustive sampling. Stochastic sampling also facilitates the use of complex quality metrics and metrics defined on non-integer disparities.

In traditional line segment detection tasks, the distributed ray Hough transform (DRHT) and the extension of [94] with distributed ray tracing (Palmer-K_{DRT}) compare favorably to alternative approaches. Line parameter estimation accuracy is improved, particularly when operating on reduced input image resolutions or reduced accumulator matrix sizes.

A novel application of distributed (stochastic) ray tracing (DRT) improves the performance of a Hough transform-based dense multi-view stereo algorithm. Aliasing is reduced, resulting in improved recovery of detail and fewer holes or defects in the recovered models. The use of DRT (or equivalently, DRHT) outperforms alternative extensions of the approach based on other Hough transform variations – specifically the re-accumulation approach of [39].

The potential applications of DRT to computational stereo go beyond the Hough transform-based approach considered in the experiments. DRT principles could be used to mitigate aliasing in many stereo algorithms that sample and quantize values on a regular lattice. DRT also provides a flexible and tractable way to approximate complex modeling, fusion, or other calculations that may be too complex to compute analytically, and to incorporate prior knowledge into those approximations. It is most effective in the face of strong aliasing, which could be caused by low input resolution or quantization of

internal data. Despite recent advances in sensor resolution, these will continue to be important motivating scenarios for the foreseeable future.

Stochastic computational stereo algorithms can offer advantages over deterministic algorithms. My work demonstrates two examples of this and suggests several more. Stochastic algorithms are severely under-represented in the literature (in contrast with stochastic models). The broadest conclusion to be drawn from this work, therefore, is that both practical and academic gaps in the current state of the art could be reduced by the continued exploration of stochastic computational stereo algorithms.

6.2 Future Work

As with conclusions, both specific and general future work can be identified. QUESS can be refined in a variety of ways. The Hough transform (HT) based stereo algorithm can also be refined, Additional work would extend new stereo algorithms with DRT and also pursue totally new stochastic algorithms.

The current formulation of QUESS influence is heuristic. It would be valuable to identify a formal energy function that the approach optimizes, prove its convergence, and identify the pre-requisites to its convergence. This would remove the heuristic categorization and allow a comparison to other methods that define an explicit energy function. Alternative influence definitions could be explored, based on either heuristic or theoretic foundations. It may be possible to relate the approach to gradient ascent search, with which it shares some similarities.

The relative benefits of stochastic sampling in QUESS become stronger as the match quality function becomes more complex. Further work with more complex match quality metrics is of interest. Of particular interest are metrics that mitigate illumination changes and calibration errors in the input data. Both issues are significant in the video dataset from Section 3.3.2, and would be expected from future ubiquitous low-cost autonomous systems.

More tactical and implementation-specific improvements can undoubtedly still be made. QUESS search schedules can be refined and tuned. Anisotropic filtering showed promise for influence aggregation and depth estimate smoothing, but was abandoned because of its high runtime. Fast anisotropic filtering methods, e.g. [1], should be pursued, as well as relevant generalizations of median filtering [11]. Finally, there has been no significant effort to optimize the QUESS implementation.

Many avenues exist for future work on the use of DRT in the HT-based algorithm of Chapter 5. A very interesting and achievable improvement would be to tailor DRT ray perturbation distributions to reflect prior knowledge about camera parameter inaccuracies or other sources of input or geometry error. This has been explored in unpublished work but should be investigated more thoroughly [21]. It has the potential to approximate optimal fusion of information given all prior knowledge on camera viewpoints and parameters. Another logical extension would be to combine the use of DRT with the re-voting approach of [39] for potential synergistic benefits.

The HT-based stereo modeling approach could be extended (independent of the use of DRT) to incorporate custom processing in each of its stages or after the current set of stages. This was partially explored in [20]. Additional 1D signal processing operations can augment evidence signal interpretation (e.g., predicates on which trajectories are allowed to re-vote for peaks). Additional 2D operations can augment wireframe-to-dense processing (e.g., filtering dense range images to remove noise). Additional 3D operations can remove spurious results in intermediate voxel models to improve the final results (e.g., morphological filtering of the wireframe models). Finally, in other unpublished work, the use of a technique based on random sample consensus (RANSAC) [36] improved the quality of wireframe models [21].

As with QUESS, optimized implementations of HT-based modeling with DRT would be of interest. The HT-based approach seems to be amenable to parallelized implementation on a graphics processing unit (GPU) or other non-traditional architecture, but additional consideration is needed to confirm that.

Future work should apply DRT to computational geometry and computational stereo beyond the approaches explored in this dissertation. It would be straightforward to generalize the distributed ray Hough transform (DRHT) for arbitrary shapes and higher-dimensions, as was done with the standard HT. The use of DRT in HT-based multi-view stereo can be seen as one such extension, but many other applications could also benefit.

The Space Carving (SC) algorithm and its variants (see Section 2.1.4.3) are particularly interesting for extension with DRT. Existing SC algorithms provide high-quality results. SC could be extended using DRT to better model the many-to-many mappings between pixels and voxels during the carving process. This could help overcome the current limitations on low-resolution imagery noted in [71].

It would be desirable to augment both QUESS and HT-based stereo, with processing before the current operations, to process uncalibrated video. This can be done by pre-pending sparse feature selection [108], tracking [58], and bundle adjustment [114]. This has been reduced to practice in [3] and others.

Long-term future work should pursue separate and additional stochastic algorithms for computational stereo. A larger study of stochastic algorithms, across many research teams, is required to reduce the gap of representation in the literature between deterministic and stochastic approaches. This effort may eliminate many of the existing practical problems in computational stereo as well.

Bibliography

[1] S. T. Acton, A. C. Bovik, and M. M. Crawford, “Anisotropic diffusion pyramids for image segmentation,” *Proc. IEEE Int. Conf. on Image Processing*, 1994.

[2] N. Ayache and F. Lustman, “Fast and reliable trinocular stereovision,” *Proc. Int. Conf. on Computer Vision*, pp. 422-427, 1987.

[3] A. Azarbayejani and A. Pentland, “Recursive estimation of motion, structure, and focal length,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 6, pp. 562-575, 1995.

[4] H. Baker and T. Binford, “Depth from edge and intensity based stereo,” *Proc. Int. Joint Conf. on Artificial Intelligence*, pp. 631-636, 1981.

[5] R. Balter, P. Gioia, and L. Morin, “Scalable and efficient video coding using 3-D modeling,” *IEEE Trans. on Multimedia*, vol. 8, no. 6, pp. 1147-1155, 2006.

[6] S. Barnard and M. Fischler, “Computational Stereo,” *ACM Computing Surveys*, vol. 14, pp. 553-572, 1982.

[7] S. Barnard, “Stochastic stereo matching over scale,” *Int. Journal of Computer Vision*, vol. 3, no. 1, pp. 17-32, 1989.

[8] M. J. Black and A. Rangarajan, “On the unification of line processes, outlier rejection, and robust statistics with applications in early vision,” *Int. Journal of Computer Vision*, vol. 19, no. 1, pp. 57-91, 1996.

[9] A. F. Bobick and S. S. Intille, “Large occlusion stereo,” *Int. Journal of Computer Vision*, vol. 33, no. 3, pp. 181-200, 1999.

[10] R.C. Bolles, H.H. Baker, and M.J. Hannah, “The JISCT stereo evaluation,” *Proc. DARPA Image Understanding Wksp.*, pp. 263-274, 1993.

[11] A. C. Bovik, T. S. Huang, and D. C. Munson, “A generalization of median filtering using linear combinations of order statistics,” *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 31, no. 6, pp. 1342-1350, 1983.

[12] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222-1239, 2001.

[13] A. Broadhurst, T. W. Drummond, and R. Cipolla, “A probabilistic framework for space carving,” *Proc. Int. Conf. on Computer Vision*, 2001.

[14] M. Z. Brown, D. Burschka, and G. D. Hager, “Advances in computational stereo,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993-1008, 2003.

[15] N. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla, “Automatic 3D object segmentation in multiple views using volumetric graph-cuts,” *Proc. British Machine Vision Conf.*, 2007.

[16] N. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla, “Using multiple hypotheses to improve depth-maps for multi-view stereo,” *Proc. European Conf. on Computer Vision*, 2008.

[17] T. R. Coffman and A. C. Bovik, “Fast computation of dense stereo correspondences by stochastic sampling of match quality,” *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2008.

[18] T. R. Coffman, *System and method for fast computation of dense stereo correspondences by stochastic sampling of match quality,* international patent application PCT/US2008/005271, filed 2008.

[19] T. R. Coffman and A. C. Bovik, “Efficient stereoscopic ranging via stochastic sampling of match quality,” *IEEE Trans. on Image Processing*, vol. 19, no. 2, pp. 451-460, 2010.

[20] T. R. Coffman and A. C. Bovik, “Multi-view stereo ranging via distributed ray tracing,” *Proc. IEEE Southwest Symp. on Image Analysis and Interpretation*, 2010.

[21] T. R. Coffman, T. M. Bowles, and S. F. Dubuque, “3DWAR: 3D Wide Area Reconstruction,” *Final Technical Report for U.S. Army Aviation and Missile Command contract W31P4Q-09-C-0464*, 2010.

[22] T. R. Coffman and A. C. Bovik, “Applications of distributed ray tracing to Hough transform and multi-view stereo,” *submitted to IEEE Trans. on Image Processing*, 2011.

[23] R. T. Collins, “A space-sweep approach to true multi-image matching,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1996.

[24] R. T. Collins, C. Jaynes, Y. Cheng, X. Wang, F. Stolle, H. Schultz, E. Riseman, and A. Hanson, “The Ascender System: Automated Site Modeling from Multiple Aerial Images,” *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 143-162, 1998.

[25] R. L. Cook, T. Porter, and L. Carpenter, “Distributed ray tracing,” *Computer Graphics*, vol. 18, pp. 165-174, 1984.

[26] M. Creutz, “Microcanonical Monte Carlo Simulation,” *Physical Review Letters*, vol. 50, no. 9, pp. 1411-1414, 1983.

[27] M. Creutz, “Microcanonical Cluster Monte Carlo Simulation,” *Physical Review Letters*, vol. 69, no. 7, pp. 1002-1005, 1992.

[28] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," *Proc. ACM Int. Conf. on Computer Graphics and Interactive Techniques*, 1996.

[29] DARPA Grand Challenge 2005 website, http://www.darpa.mil/grandchallenge05/.

[30] P. Dev, "Segmentation processes in visual perception: A cooperative neural model," *COINS Technical Report 74C-5*, University of Massachusetts at Amherst, 1974.

[31] U. R. Dhond and J. K. Aggarwal, "Structure from stereo – a review," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1489-1510, 1989.

[32] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, pp. 11-15, 1972.

[33] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification: 2nd Edition*, Hoboken, NJ: Wiley & Sons, 2001.

[34] O. Faugeras, "What can be seen in three dimensions with an uncalibrated stereo rig?" *Proc. European Conf. on Computer Vision,* pp. 563-578, 1992.

[35] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *Journal of the Optical Society of America – A*, vol. 4, no. 12, pp. 2379-2394, 1987.

[36] M. Fischler and R. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 1981.

[37] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice: 2nd Edition*, Boston, MA: Addison-Wesley, 1996.

[38] S. Geman and D. Geman, “Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721-741, 1984.

[39] G. Gerig and F. Klein, “Fast contour identification through efficient Hough transform and simplified interpretation strategy,” *Proc. Int. Joint Conf. on Pattern Recognition*, 1986.

[40] B. Gulyás and P. E. Roland, “Binocular disparity discrimination in human cerebral cortex: Functional anatomy by positron emission tomography,” *Proc. National Academy of Sciences*, vol. 91, no. 4, pp. 1239-1243, 1994.

[41] T. Hamano, T. Yasuno, and K. Ishii, “Direct estimation of structure from non-linear motion by voting algorithm without tracking,” *Proc. Int. Conf. on Pattern Recognition*, 1992.

[42] R. Hartley, R. Gupta, and T. Chang, "Stereo from Uncalibrated Cameras," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 761-764, 1992.

[43] R. Hartley and P. Sturm, "Triangulation," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146-157, 1997.

[44] R. Hartley, "Theory and practice of projective rectification," *Int. Journal of Computer Vision*, vol. 35, no. 2, pp. 115-127, 1998.

[45] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge, UK: Cambridge University Press, 2000.

[46] E. E. Hemayed, "A Survey of Camera Self-Calibration," *Proc. IEEE Conf. on Advanced Video and Signal Based Surveillance*, pp. 351-357, 2003.

[47] Online image from http://en.wikipedia.org/wiki/Ray_tracing_(graphics), attributed to user Henrik (http://commons.wikimedia.org/wiki/User:Henrik). Published under GNU Free Documentation License, April 12, 2008.

[48] H. Hirschmüller, "Improvements in real-time correlation-based stereo vision," *Proc. IEEE Wksp. on Stereo and Multi-Baseline Vision*, 2001.

[49] H. Hirschmüller, "Accurate and efficient stereo processing by semi-global matching and mutual information," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 807-814, 2005.

[50] H. Hirschmüller and D. Scharstein, “Evaluation of cost functions for stereo matching,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007.

[51] W. Hoff and N. Ahuja, “Surfaces from stereo: Integrating feature matching, disparity estimation, and contour detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, no. 2, pp. 121-136, 1989.

[52] B. K. P. Horn and M. J. Brooks, *Shape From Shading*, Cambridge, MA: MIT Press, 1989.

[53] P. Hough, *Method and means for recognizing complex patterns*, United States patent 3,069,654, issued December 18, 1962.

[54] Y. C. Hsieh, D. McKeown, and F. P. Perlant, “Performance evaluation of scene registration and stereo matching for cartographic feature extraction,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 214-238, 1992.

[55] T. Huang, G. Yang, and G. Tang, "A fast two-dimensional median filtering algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 27, pp. 13-18, 1979.

[56] H. Jin, S. Soatto, and A. Yezzi, “Multi-view stereo beyond Lambert,” *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 171-178, 2003.

[57] D. G. Jones and J. Malik, “A computational framework for determining stereo correspondence form a set of linear spatial filters,” *Proc. European Conf. on Computer Vision*, pp. 395-410, 1992.

[58] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *Transactions of the ASME – Journal of Basic Engineering*, vol. 82, pp. 33-45, 1960.

[59] T. Kanade and M. Okutomi, “A stereo matching algorithm with an adaptive window: Theory and experiment,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920-932, 1994.

[60] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka, “A stereo machine for video-rate dense depth mapping and its new applications,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1996.

[61] S. B. Kang, R. Szeliski, and J. Chai, “Handling occlusions in dense multi-view stereo,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[62] M. Kass, “Linear image features in stereopsis,” *Int. Journal of Computer Vision*, vol. 1, no. 4, pp. 357-368, 1988.

[63] S. Kawato, “Hough transform to extract 3D Information from images of different view points,” *Proc. Int. Conf. on Computer analysis of Images and Patterns*, 1993.

[64] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by Simulated Annealing,” *Science*, vol. 220, no. 4598, pp. 671-680, 1983.

[65] Y. Kim and J. K. Aggarwal, “Positioning 3-D objects using stereo images,” *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 361-373, 1987.

[66] J. Kim, V. Kolmogorov, and R. Zabih, “Visual correspondence using energy minimization and mutual information,” *Proc. Int. Conf. on Computer Vision*, 2003.

[67] N. Kiryati and A. M. Bruckstein, “Antialiasing the Hough transform,” *Computer Vision, Graphics, and Image Processing*, vol. 53, pp. 213-222, 1991.

[68] W. Klarquist and A. C. Bovik, “FOVEA: A foveated, multi-fixation, vergent active stereo system for dynamic three-dimensional scene recovery,” *IEEE Trans. on Robotics and Automation*, vol. 14, no. 5, pp. 755-770, 1998.

[69] E. Krotkov, “Focusing,” *Int. Journal of Computer Vision*, vol. 1, pp. 223-237, 1987.

[70] T. Kumar, D. A. Glaser, “Shape analysis and stereopsis for human depth perception,” *Vision Research*, vol. 32, no. 3, pp. 499-512, 1992.

[71] K. N. Kutulakos and S. M. Seitz, “A theory of shape by space carving,” *Int. Journal of Computer Vision*, vol. 38, pp. 199-218, 2000.

[72] K. N. Kutulakos, “Approximate N-view stereo,” *Proc. European Conf. on Computer Vision*, 2000.

[73] W. C. Y. Lam, L. T. S. Lam, K. S. Y. Yuen, and D. N. K. Leung, “An analysis on quantizing the Hough space,” *Pattern Recognition Letters*, vol. 15, pp. 1127-1135, 1994.

[74] R. Larcom and T. R. Coffman, *System and method for visually tracking with occlusions*, international patent application PCT/US2009/085233, filed 2009.

[75] L. Levkovich-Maslyuk et al., “Depth Image-Based Representation and Compression for Static and Animated 3-D Objects,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 7, pp. 1032-1045, 2004.

[76] H. S. Lim and T. O. Binford, “Stereo correspondence: A hierarchical approach,” *Proc. DARPA Image Understanding Wksp.*, pp. 234-241, 1987.

[77] H. G. Longbotham and A. C. Bovik, “Theory of order statistic filters and their relationship to linear FIR filters,” *IEEE Trans. on Acoustics, Speech, and Signal Processing,* vol. 37, no. 2, pp. 275-287, Feb. 1989.

[78] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[79] W. Malpica and A. C. Bovik, “Range image quality assessment by structural similarity,” *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2009.

[80] D. C. Marr and T. Poggio, “Cooperative computation of stereo disparity,” *Science*, vol. 194, pp. 283-287, 1976.

[81] D. C. Marr and T. Poggio, “A computational theory of human stereo vision,” *Proc. Royal Society of London*, vol. B204, pp. 301-328, 1979.

[82] A. Martin, “Distributed Ray Tracing,” online course material available at http://web.cs.wpi.edu/~matt/courses/cs563/talks/dist_ray/dist.html, 1999.

[83] L. Matthies, A. Kelly, T. Litwin, and G. Tharp, “Obstacle detection for unmanned ground vehicles: a progress report,” *Proc. IEEE Intelligent Vehicles Conf.*, pp. 66-71, 1995.

[84] J. Mayhew and J. Frisby, “Psychophysical and computational studies towards a theory of human stereopsis,” *Artificial Intelligence*, vol. 17, pp. 349-385, 1981.

[85] G. Medioni and R. Nevatia, “Segment-based stereo matching,” *Computer Vision, Graphics, and Image Processing*, vol. 31, pp. 2-18, 1985.

[86] P. Merrell et al., “Real-time visibility-based fusion of depth maps,” *Proc. IEEE Int. Conf. on Computer Vision*, 2007.

[87] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, “Equations of state calculations by fast computing machines,” *Journal of Chemical Physics*, vol. 21, pp. 1087-1091, 1953.

[88] J. Monaco, A. C. Bovik, and L. K. Cormack, “Epipolar Spaces and Optimal Sampling Strategies,” *Proc. IEEE Int. Conf. on Image Processing*, 2007.

[89] H. P. Moravec, “Towards automatic visual obstacle avoidance,” *Proc. Int. Joint Conf. on Artificial Intelligence*, p. 584, 1977.

[90] Y. Nakamura, T. Matsuura, K. Satoh, and Y. Ohta, “Occlusion detectable stereo – occlusion patterns in camera matrix,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 371-378, 1996.

[91] R. Nevatia and K. Babu, “Linear feature extraction and description,” *Computer Graphics and Image Processing*, vol. 13, pp. 257-269, 1980.

[92] W. Niblack and D. Petkovic, “On improving the accuracy of the Hough transform,” *Proc. IEEE Conf, on Computer Vision and Pattern Recognition*, 1988.

[93] Y. Ohta and T. Kanade, “Stereo by intra- and inter-scanline search using dynamic programming,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 7, no. 2, pp. 139-154, 1985.

[94] P. L. Palmer, J. Kittler, and M. Petrou, “An optimizing line finder using a Hough transform algorithm,” *Computer Vision and Image Understanding*, vol. 67, no. 1, pp. 1-23, 1997.

[95] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, no. 7, pp. 629-639, 1990.

[96] S. Perreault and P. Hebert, "Median filtering in constant time," *IEEE Trans. on Image Processing*, vol. 18, no. 9, pp. 2389-2394, 2007.

[97] M. Pollefeys, R. Koch, M. Vergauwen, and L. Van Gool, “Automated reconstruction of 3D scenes from sequences of images,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 55, pp. 251-267, 2000.

[98] J. Princen, H. K. Yuen, J. Illingworth, and J. Kittler, “A comparison of Hough transform methods,” *Proc. Int. Conf on Image Processing and its Applications*, 1989.

[99] L. Quam, “Hierarchical warp stereo,” *Proc. Image Understanding Wksp.*, pp. 149-155, 1984.

[100] R. P. N. Rao and D. H. Ballard, “An active vision architecture based on iconic representations,” *Artificial Intelligence*, vol. 78, pp. 461-505, 1995.

[101] A. Rosenfeld, R. Hummel, and S. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. on Man, Systems, and Cybernetics*, vol. 6, pp. 420-423, 1976.

[102] D. Scharstein and R. Szeliski, "Stereo matching with nonlinear diffusion," *Int. Journal of Computer Vision*, vol. 28, no. 2, pp. 155-174, 1998.

[103] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *Int. Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7-42, 2002. Also *Technical Report MSR-TR-2001-81, Microsoft Research*, 2001.

[104] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 195-202, 2003.

[105] Y. Y. Schechner and N. Kiryati, "Depth from Defocus vs. Stereo: How Different Really Are They?" *Int. Journal of Computer Vision*, vol. 39, no. 2, pp. 141-162, 2000.

[106] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2006.

[107] H. R. Sheikh and A. C. Bovik, “An evaluation of recent full reference image quality assessment algorithms,” *IEEE Trans. on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[108] J. Shi and C. Tomasi, “Good features to track,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1994.

[109] S. M. Smith and J. M. Bradley, “SUSAN – a new approach to low-level image processing,” *Int. Journal of Computer Vision*, vol. 23, no. 1, pp. 45-78, 1997.

[110] R. Szeliski and D. Scharstein, “Sampling the disparity space image,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 3, pp. 419-425, 2004.

[111] R. S. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, “A comparative study of energy minimization methods for Markov random fields,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1068-1080, 2008.

[112] C. J. Taylor, “Surface reconstruction from feature based stereo,” *Proc. Int. Conf. on Computer Vision*, 2003.

[113] M. J. Tovée, *An Introduction to the Visual System*, Cambridge, UK: Cambridge University Press, 1996.

[114] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle Adjustment – A Modern Synthesis," *Proc. Int. Wksp. on Vision Algorithms: Theory and Practice*, pp. 298-372, 1999.

[115] R. Vidal and J. Oliensis, "Structure from Planar Motions with Small Baselines," *Proc. European Conf. on Computer Vision*, pp. 383-398, 2002.

[116] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *Int. Journal of Computer Vision*, vol. 24, no. 2, pp. 137-154, 1997.

[117] Z. Wang, A. C. Bovik, and H. Sheikh, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Trans. on Image Processing*, vol. 13, no. 4, pp. 600-612, 2004.

[118] X. Wu, "An efficient antialiasing technique," *Computer Graphics*, vol. 25, no. 4, pp. 143-152, 1991.

[119] Y. Xiong, C. F. Olsen, and L. H. Matthies, "Computing depth maps from descent images," *Machine Vision and Applications*, vol. 16, no. 3, pp. 139-147, 2005.

[120] R. Yang and M. Pollefeys, "Multi-Resolution Real-Time Stereo on Commodity Graphics Hardware," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2003.

[121] A. Yao and A. Calway, "Dense 3-D structure from image sequences using probabilistic depth carving," *Proc. British Machine Vision Conference*, 2003.

[122] T. Z. Young, ed., *Handbook of Pattern Recognition and Image Processing: Computer Vision*, San Diego, CA: Academic Press, 1994.

[123] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," *Proc. European Conf. on Computer Vision*, vol. 2, pp. 151-158, 1994.

[124] C. L. Zitnick and T. Kanade, "A Cooperative Algorithm for Stereo Matching and Occlusion Detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 7, pp. 675-684, 2000.

Vita

Thayne Richard Coffman attended Colonel Zadok Magruder High School in Rockville, Maryland. He earned the Bachelor of Science degree in Computer Science and Engineering and the Master of Engineering degree in Electrical Engineering and Computer Science from The Massachusetts Institute of Technology (M.I.T.), both in Spring 1996. Following graduation he began his professional career, which has included positions at Trilogy Software, Question Technologies, and 21st Century Technologies.

He attended the Ph.D. program at The University of Texas at Austin (UT-Austin) as a part-time student from Fall 2002 to Spring 2011, under the supervision of Professor Alan C. Bovik. While at UT-Austin he was a member of the Laboratory for Image and Video Engineering (LIVE).

His research interests include image and video processing, pattern classification, autonomous systems, graph analytics, and cyber network operations.

Permanent contact information (email): thayne@alum.mit.edu

This dissertation was typeset with Microsoft Word by the author.