Surfaces with Occlusions from Layered Stereo

Michael H. Lin and Carlo Tomasi

Abstract—We propose a new binocular stereo algorithm that estimates scene structure as a collection of smooth surface patches. The disparities within each patch are modeled by a continuous-valued spline, while the extent of each patch is represented via a pixelwise partitioning of the images. Disparities and extents are alternately estimated in an iterative, energy minimization framework. Experimental results demonstrate that, for scenes consisting of smooth surfaces, the proposed algorithm significantly improves upon the state of the art.

Index Terms—Binocular stereo vision, energy minimization, graph cuts, hybrid system, smooth surfaces, surface fitting, boundary localization, sharp discontinuities, quantitative comparison.

1 Introduction

THE foundations of stereo are *correspondence* and *triangulation*. Given two images, if one can find a pair of left and right image points that correspond to the same world point, geometry readily yields the three-dimensional position of that world point. It is the search for such corresponding pairs that is the central part of the stereo problem.

There are several constraints that help to solve correspondence. Given geometric calibration, the *epipolar constraint* reduces the search for possible point matches from two dimensions to one. Given photometric calibration, *color constancy* further narrows the possibilities to points that look alike. Marr and Poggio [15] proposed two additional heuristics that mitigate the ill-posedness of stereo: *uniqueness*, which states that "each item from each image may be assigned at most one disparity value," and *continuity*, which states that "disparity varies smoothly almost everywhere." Of these four constraints, the former two are relatively straightforward, but the application of the latter two varies greatly [2], [9], [18]. We propose a three-axis categorization of binocular stereo algorithms according to their interpretation of continuity and uniqueness. In the following sections, we list last, for all three axes, that category which we consider to be the most preferable.

1.1 Continuity

The first axis describes the modeling of continuity of disparities within smooth surface patches.

Constant. Every point within any one smooth surface patch is assigned the same disparity value. This value is usually chosen from a finite, predetermined set of possible disparities, such as the set of all integers within a given range or the set of all multiples of a given fraction (e.g., $1/4$ or $1/2$) within a given range. Examples of prior work in this category include traditional sum-of-squared-differences (SSD) correlation, as well as [5], [10], [12], [13], [15].

Discrete. Disparities are again limited to discrete values, but with multiple distinct values permitted within each surface patch. Surface smoothness in this context means that, within each surface, neighboring pixels should have disparity values that are numerically as close as possible to one another. Examples of prior work in this category include [3], [11], [17], [21].

Real. Disparities within each smooth surface patch vary smoothly over the real numbers. Various interpretations of smoothness can be used; most try to minimize local first or second-order differences in disparity. Examples of prior work in this category include [1], [4], [19], [20].

1.2 Discontinuity

The second axis describes the treatment of discontinuities at the boundaries of surface patches: The penalty for a discontinuity is examined as a function of the size of the jump of the discontinuity.

Free. Discontinuities are not specifically penalized. These methods often fail to resolve the ambiguity caused by periodic textures or textureless regions. Examples of prior work in this category include traditional SSD correlation, as well as [12], [15], [21].

Infinite. Discontinuities are penalized infinitely, i.e., they are disallowed. The recovered disparity map is "smooth" everywhere. Examples of prior work in this category include [16], [19].

Convex. Discontinuities are allowed, but a penalty is imposed that is a finite, positive, convex function of the size of the jump of the discontinuity. The resulting discontinuities often tend to be somewhat blurred because the cost of two adjacent discontinuities is no more than that of a single discontinuity of the same total size. Examples of prior work in this category include [11], [17], [20].

Nonconvex. Discontinuities are allowed, but a penalty is imposed that is a nonconvex function of the size of the jump of the discontinuity. The resulting discontinuities often tend to be fairly clean because the cost of two adjacent discontinuities is generally more than that of a single discontinuity of the same total size. Examples of prior work in this category include [3], [6], [7], [10].

1.3 Uniqueness

The third axis describes the application of uniqueness, especially to occlusions.

One-Way. Uniqueness is assumed within a chosen reference image, but not considered within the other. That is, each location in one image is assigned at most one disparity, but the disparities at multiple locations in that image may point to the same location in the other image. Typically, each location in one image is assigned *exactly* one disparity, with occlusion relationships being ignored. Examples of prior work in this category include traditional SSD correlation, as well as [4], [6], [12].

Asymmetric Two-Way. Uniqueness is encouraged for both images, but the two images are treated unequally. That is, reasoning about occlusion is done and the occlusions that accompany depth discontinuities are qualitatively recovered, but there is still one chosen reference image, resulting in asymmetries in the reconstructed result. Examples of prior work in this category include [1], [5], [15], [20], [21].

Symmetric Two-Way. Uniqueness is enforced in both images symmetrically; detected occlusion regions are marked as being without correspondence. Examples of prior work in this category include [3], [10], [11], [13].

1.4 Overview

In this paper, we propose an algorithm (described more fully in [14]) that lies in the most preferable category along all three axes. To the authors' knowledge, ours is the first such algorithm for binocular stereo. We contend that, for scenes consisting of smooth surfaces, our algorithm improves upon the state of the art, achieving both more accurate localization in depth of surface interiors via subpixel disparity estimation and more accurate localization in the image plane of surface boundaries via the symmetric treatment of images with proper handling of occlusions.

- *M.H. Lin is with Acuity Technologies, 3475 Edison Way, Building P, Menlo Park, CA 94025. E-mail: michelin@cs.stanford.edu.*
- *C. Tomasi is with the Department of Computer Science, Levine Science Research Center, Section D, Duke University, PO Box 90129, Durham, NC 27708. E-mail: tomasi@cs.duke.edu.*

Manuscript received 11 June 2002; revised 6 May 2003; accepted 15 Sept. 2003.
Recommended for acceptance by L. Quan.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 116735.

Section 2 describes our mathematical model of the stereo problem. Sections 3 and 4 describe surface fitting and boundary localization, while Section 5 summarizes the overall algorithm. Finally, Sections 6 and 7 present some experimental results and a few concluding remarks.

2 Preliminaries

In this section, we develop a mathematical abstraction of the stereo problem in a continuous domain. Discretization for computational feasiblity will be discussed in subsequent sections.

2.1 Mathematical Abstraction

We use a layered model [8] to represent possible solutions to the stereo problem. We follow the common practice of assuming that input images have been normalized for both photometric and geometric calibration. In particular, we assume that the images are rectified. Let

$$\mathcal{I} = \{p = (x, y, t)\} = (\mathcal{R} \times \mathcal{R} \times \{\text{"LEFT"}, \text{"RIGHT"}\})$$

be the space of image locations and let

$$I \ : \ \mathcal{I} \mapsto \mathcal{R}^m$$

be the given input image pair. Typically, $m = 3$ for color images, and $m = 1$ for grayscale images. Note that I is defined on a continuous domain; in practice, it is interpolated from discrete pixels.

Our abstract model of a hypothesized solution consists of a labeling (or segmentation) f, which assigns each point of the two input images to zero or one of n surfaces, plus n disparity maps $d[k]$, each of which assigns a disparity value to each point of the two input images:

$$\begin{array}{lll} \text{[segmentation]} & f & : \quad \mathcal{I} \mapsto \{0, 1, \ldots, N\} \\ \text{[disparity map]} & d[k] & : \quad \mathcal{I} \mapsto \mathcal{R} \text{ for } k \text{ in } \{1, 2, \ldots, \mathrm{N}\}. \end{array}$$

The segmentation function f specifies to which one of n surfaces, if any, each image location "belongs." We take belonging to mean the existence of a world point which 1) projects to the image location in question and 2) is visible in both images. For each surface, the signed disparity function $d[k]$ defines the correspondence (or matching) function $m[k]$ between image locations:

$$\begin{aligned} m[k] &: \mathcal{I} \mapsto \mathcal{I} \\ m[k](x, y, t) \ &= \ \big(x + d[k](x, y, t), \ y, \ \neg t\big), \end{aligned}$$

where $\neg$ "LEFT" = "RIGHT" and vice versa. The interpretation of this model is:

$$\begin{aligned} &\text{for all } p: \quad f(p) = k \text{ with } k > 0 \Rightarrow p \text{ corresponds to } m[k](p) \\ &f(p) = 0 \Rightarrow p \text{ corresponds to no location in the other image.} \end{aligned}$$

2.2 Desired Properties

Using this abstraction, how can we evaluate a hypothesized solution? We propose three properties that together characterize a "good" solution: consistency, smoothness, and nontriviality.

Consistency. Correspondence should be bidirectional: If p and q are images of the same world point, then each corresponds to the other; otherwise, neither corresponds to the other. It cannot be that p corresponds to q but that q does not correspond to p. This translates into a constraint on each $m[k]$, equivalent to a constraint on each $d[k]$, and additionally into a constraint on f:

$$\text{for all } k, p: \quad m[k](m[k](p)) = p \tag{1}$$

$$\text{for all } p: \quad f(p) = k \text{ with } k > 0 \ \Rightarrow \ f(m[k](p)) = k. \tag{2}$$

Ideally, consistency should be exact, but computationally, it is merely maximized.

Smoothness. Continuity dictates that a recovered disparity map should consist of smooth surface patches separated by cleanly defined, smooth boundaries. Thus, both $d[k]$ and f should be smooth. The disparity maps $d[k]$ are continuous-valued functions, so we take the smoothness of $d[k]$ to mean differentiability, with the magnitude of higher derivatives being relatively small. The segmentation function f is piecewise constant, so we take the smoothness of f to mean simplicity of the boundaries separating those pieces, with the total boundary length being relatively small.

Nontriviality. Good solutions should explain the input as much as possible. For example, any two images could be interpreted as views of two distinct surfaces, each shown to one camera; such a trivial interpretation would be valid but undesirable. Moreover, although color constancy can be violated, a solution that does so excessively would also be undesirable. That is, we expect that a correspondence exists, and that color constancy holds, for "most" image locations:

$$\begin{aligned} \text{for most } p: \quad & f(p) > 0 \\ \text{for most } p \text{ where } f(p) > 0: \quad & I\big(m[f(p)](p)\big) \approx I(p). \end{aligned}$$

2.3 Energy Minimization

We formalize the stereo problem in an energy minimization framework. We formulate six energy terms, corresponding to each of the three desired properties, applied to both disparity maps over surface interiors and segmentation via surface boundaries; total energy is the sum thereof.

3 Surface Fitting

In this section, we consider the subproblem of estimating the disparity maps $d[k]$, supposing that the segmentation f is known. Using this context, we formulate and discuss the minimization of the three energy terms that encourage surface nontriviality, smoothness, and consistency.

3.1 Surface Model

We model the disparity map of each surface as a bicubic B-spline. The control points of the spline are placed in each image on a fixed, uniform rectangular grid (5×5 in our experiments). The resulting spline surface can be thought of as a linear combination of shifted basis functions, with shifts constrained to the grid. Mathematically, we restrict each $d[k]$ as follows:

$$d[k](x, y, t) = \sum_{i,j} \big(D[k][i, j, t] \cdot b(x - in, y - jn)\big), \tag{3}$$

where b is the bicubic basis function, D is the lattice of control points, and n is the spacing thereof.

3.2 Surface Nontriviality

This energy term, often called the "data term" in other literature, penalizes any deviation from color constancy. We quantify such deviation using a scaled sum of squared differences:

$$E\_match\_I = \sum_p \begin{cases} g(I(m[k](p)) - I(p); A(p)) & \text{if } f(p) = k \text{ with } k > 0, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $g(v; A) = v^T \cdot A \cdot v$, and where $A(p)$ is a measure of certainty, defined as follows:

Let $\mathbf{I}$ be the $m \times m$ identity matrix and $\mathbf{x^2}$ be shorthand for the outer product $\mathbf{x}\mathbf{x}^T$. Let $G_\sigma * I$ represent the convolution of I with a Gaussian of width σ. Then, for all p, we define

$$A = \left[\epsilon \mathbf{I} + G_\sigma * (I^2) - (G_\sigma * I)^2\right]^{-1},$$

where ϵ and σ are small constants. Intuitively, $A(p)$ serves to normalize the local contrast of I around p. Note that this and the next two energy terms should be defined as integrals over all $p \in \mathcal{I}$, but, for computational convenience, we merely take a finite sum over discrete pixel positions.

3.3 Surface Smoothness

Although our spline model already ensures some degree of surface smoothness, this inherent smoothness is limited to a spatial scale not much larger than that of the spline control point grid. Because we would also like to have smoothness on a more global scale, we impose an additional energy term which, loosely speaking, is proportional to the global "variance" of the surface slope:

$$E\_smooth\_d[k] = \lambda\_smooth\_d \cdot \sum_p \left\| \nabla d[k](p) - \mathrm{mean}(\nabla d[k]) \right\|^2,$$

where the summation and the mean are both taken over all discrete pixel positions p, independent of the segmentation f. This energy term attempts to quantify deviations from global planarity.

3.4 Surface Consistency

For perfect consistency, a surface should have left and right views that coincide exactly with one another, as specified in (1). However, with left and right views simultaneously constrained each to have the form of (3), exact coincidence is generally no longer possible. Therefore, to allow but discourage any noncoincidence, we propose the energy term

$$E\_match\_d[k] = \lambda\_match\_d \cdot \sum_p \big(m[k](m[k](p)) - p\big)^2,$$

which, intuitively, measures the distance between the surfaces defined by the left and right views. Again, the sum is taken over all discrete pixel positions p, independent of the segmentation f.

3.5 Surface Optimization

Given a particular k, this section's subproblem is to minimize total energy by varying $d[k]$ while holding f and the remaining $d[j]$ constant. Total energy is a sum of six terms, three of which were shown in this section to depend smoothly on $d[k]$. In Section 4, the remaining three terms are shown either to depend only on f or to depend smoothly on $d[k]$. Hence, the total energy as a function of $d[k]$ is differentiable and can be minimized with standard gradient-based numerical methods. For convenience, we use MATLAB's optimization toolbox; the specific algorithm chosen is a trust region method with a 2D quadratic subproblem. For each $k > 0$, we call minimizing total energy over $d[k]$, a *surface-fitting step*.

4 Segmentation

In this section, we consider the subproblem of estimating the segmentation f, supposing that the disparity maps $d[k]$ are known. Using this context, we formulate and discuss the minimization of the three energy terms that encourage segmentation nontriviality, smoothness, and consistency.

4.1 Segmentation by Graph Cuts

Boykov et al. [6] showed that certain labeling problems can be formulated as energy minimization problems and solved efficiently by finding minimum-cost cuts of associated network graphs. Formally, let $\mathcal{L}$ be a finite set of labels, $\mathcal{P}$ be a finite set of items, and $\mathcal{N} \subseteq \mathcal{P} \times \mathcal{P}$ be the set of interacting pairs of items. The methods of [6] find a labeling f that assigns exactly one label $f_p \in \mathcal{L}$ to each item $p \in \mathcal{P}$, subject to the constraint that an energy function of the form

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V_{p,q}(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p) \tag{5}$$

be minimized. Individual energies D_p can be arbitrary, while interaction energies $V_{p,q}$ should be either semimetric or metric.

This generic formulation of an energy-minimizing labeling problem maps to our formulation of the stereo problem as follows: The labels are the integers $0 \ldots N$ that are the possible values of the segmentation function f and the items are the pixels of each input image. This is in contrast to [6] in which the items are the pixels of a single reference image and to [13] in which the items are pairs of potentially corresponding pixels. In our formulation, the individual energies stem from testing color constancy at varying disparities and the interaction energies stem from the expectations of smoothness and consistency.

Our algorithm prohibits pixels from being split spatially among several surfaces, instead constraining surface boundaries to lie on pixel boundaries. Thus, in representing the continuous-domain segmentation function f with a finite number of unknowns, we perform nearest-neighbor interpolation on a discrete grid of pixels F, defined on an integer lattice:

$$f(x, y, t) = F(\mathrm{round}(x), \mathrm{round}(y), t).$$

4.2 Segmentation Nontriviality

The primary goal of the segmentation subproblem is to assign each pixel to the surface it fits best. This is accomplished by minimizing E_match_I, defined in (4); here, we consider it as a functional of f with $m[k]$ being constant, instead of vice versa. However, note that, since $g(\cdot)$ is nonnegative, E_match_I is trivially minimized by $f(p) \equiv 0$. To discourage solutions with a large number of unassigned pixels, we add a fixed penalty for each unassigned pixel:

$$E\_unassigned = \sum_p \begin{cases} \lambda\_unassigned & \text{if } f(p) = 0, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, the underlying segmentation problem, for the moment ignoring smoothness and consistency, is to find the labeling f that minimizes the sum $E\_match\_I + E\_unassigned$. Put into the form of (5), this corresponds to the following definition of individual pixel preferences:

$$D_p(f_p) = g\Big(I(m[k](p)) - I(p); A(p)\Big) \quad \text{for } f_p > 0,$$
$$D_p(0) = \lambda\_unassigned.$$

4.3 Segmentation Smoothness

A secondary goal is to encourage a simple segmentation with "smooth" boundaries of surface extents. There are several attributes which can formalize this notion; we choose to minimize boundary length because it is relatively simple to optimize and works fairly well in practice.

In addition to this preference for simple boundaries, there is also an expectation that boundaries will generally be correlated with monocular image features (called "static cues" in [6]). To estimate edge likelihood, we use a function of gradients and local contrast. This measure of edge likelihood at each point is then used to adjust the cost per unit length of boundaries passing through that point.

There is one more issue to consider: *Which* boundaries do we want to minimize? Intuitively, minimizing the length of a boundary will tend to shorten or remove any protrusions or indentations that are long and thin. This makes sense for regions that correspond to surfaces, but not for regions that correspond to occlusions because occlusion regions are in fact usually long and thin.

To encourage a simple segmentation, we thus define this energy term for each surface $k > 0$:

$$E\_smooth\_f[k] = \sum_{p \text{ adjacent to } q} \begin{cases} w_s(p,q) & \text{if } f(p) = k \text{ xor } f(q) = k, \\ 0 & \text{otherwise,} \end{cases}$$

where adjacency is according to 4-connectedness within each image and where

$$w_s(p,q) = \lambda\_smooth\_f \cdot \left(1 + e^{-(|\nabla I|^T \cdot A \cdot |\nabla I|)/\tau}\right),$$

where λ_smooth_f and τ are constants and ∇I and A are both evaluated at $(p+q)/2$.

Put into the form of (5), $E\_smooth\_f[k]$ corresponds to this penalty function:

$$\begin{aligned} V_{p,q}(f_p, f_q) &= w_s(p,q) \cdot \sum_{k>0} T\big(f_p = k \text{ xor } f_q = k\big) \\ &= w_s(p,q) \cdot \begin{cases} 0 & \text{if } f_p = f_q \\ 1 & \text{if } f_p \neq f_q \text{ with } f_p = 0 \text{ or } f_q = 0 \\ 2 & \text{if } f_p \neq f_q \text{ with } f_p > 0 \text{ and } f_q > 0 \end{cases} \end{aligned}$$

for p adjacent to q, where $T(\cdot)$ equals 1 if its argument is true and equals 0 otherwise.

4.4 Segmentation Consistency

For exact consistency, the segmentation f should satisfy (2) everywhere. To quantify and discourage any inconsistencies, we formulate an energy term for each surface $k > 0$:

$$E\_match\_f[k] \approx \sum_p \begin{cases} \lambda\_match\_f & \text{if } f(p) = k \text{ xor } f(m[k](p)) = k, \\ 0 & \text{otherwise,} \end{cases}$$

which approximates the area of regions where (2) does not hold. Ideally, as with those in Section 3, this term should be defined with an integral, but, in this case, a naive finite sum is *not* an adequate substitute when subpixel disparities are allowed, as explained in [14]. Instead, we take

$$E\_match\_f[k] = \sum_{p,q} \begin{cases} \lambda\_match\_f \cdot \hat{h}\big(|m[k](p) - q|\big) & \text{if } f(p) = k \text{ xor} f(q) = k, \\ 0 & \text{otherwise,} \end{cases}$$

where p and q are on conjugate epipolar lines, and where

$$\hat{h}(\triangle d) = \begin{cases} \frac{1}{2} & \text{for } |\triangle d| \leq \frac{1}{2}, \\ \frac{3}{4} - \frac{|\triangle d|}{2} & \text{for } \frac{1}{2} < |\triangle d| < \frac{3}{2}, \\ 0 & \text{for } |\triangle d| \geq \frac{3}{2}. \end{cases}$$

Note that our implementation modifies $\hat{h}$ by rounding its "corners" (at $|\triangle d| = \frac{1}{2}$ and $|\triangle d| = \frac{3}{2}$) so that total energy remains differentiable with respect to $d[k]$.

Put into the form of (5), $E\_match\_f[k]$ corresponds to this penalty function:

$$V_{p,q}(f_p, f_q) = \sum_{k>0} w_c[k](p,q) \cdot T\big(f_p = k \text{ xor } f_q = k\big)$$

for p and q in corresponding scanlines, where

$$w_c[k](p,q) = \lambda\_match\_f \cdot \left(\hat{h}\big(m[k](p) - q\big) + \hat{h}\big(m[k](q) - p\big)\right)$$

and λ_match_f is a constant.

4.5 Segmentation Optimization

This section's subproblem is to minimize total energy by varying f while holding all $d[k]$ constant. Total energy is a sum of six terms, two of which are independent of f. In this section, the remaining four terms are written in the form of (5); moreover, our $V_{p,q}$ can be verified to be metric. Hence, the total energy as a function of f can be optimized with graph cut methods [6].

We use a modified version of the expansion algorithm of [6]. This algorithm is built from expansion moves and gets its power from the generality of such moves: An expansion on a label k finds the best configuration reachable by relabeling *any* subset of pixels with k. We precede each expansion of any label k with a contraction of the same label (by first replacing all instances of k with the spatially nearest label that is not k) which strictly enlarges the set of reachable configurations. We call such a contraction-expansion pair on any one label a *segmentation step*.

5 Overall Optimization

In this section, we consider the problem of simultaneously determining surface shape in the form of disparity maps and surface support in the form of segmentation, when both are unknown.

Our algorithm is built from the surface-fitting and segmentation steps of Sections 3 and 4. Since each of these steps reduces total energy, given a reasonable initial hypothesis, iterating these steps until convergence might give a reasonable final solution. However, during the course of such component-wise optimization, it is quite possible to reach a local minimum. These undesirable configurations are generally of two types: those in which one hypothesized surface spans several actual surfaces and those in which several hypothesized surfaces span one actual surface.

Our algorithm currently cannot reliably extract itself from the former type of local minima and therefore relies upon careful initialization to avoid getting into such situations. The initial hypothesis is formed by placing one fronto-parallel surface at every integer disparity within the specified range of possible disparities; all pixels are initially unassigned (with $f \equiv 0$).

The latter type of situation is more easily handled. Often, when several hypothesized surfaces span one actual surface, one hypothesized surface will eventually come to dominate and the others will naturally be driven to extinction. When this is not the case, a *merge step* will generally remedy the situation. To take a merge step, we first save a snapshot of the current state. We then forcefully remove one surface. The "orphaned" pixels are relabeled with $f = 0$, but are immediately redistributed among the remaining surfaces by a series of segmentation steps. Further surface fitting and segmentation steps are then taken until either the total energy falls below that of the saved snapshot, in which case the merge succeeds and the snapshot is discarded, or the total energy plateaus above that of the snapshot, in which case the merge fails and the snapshot is restored.

The complete algorithm is as follows:

1. Initialize hypothesis with surfaces at integer disparity.
2. Repeat:
   a. Alternately apply segmentation and surface fitting steps until progress is negligible.
   b. For each surface, attempt to merge it until one merge succeeds *or* all merges fail.

   until all merges fail.
3. Optionally "fill in" unmatched regions, using neighboring matched regions (see [14]).

6 Experimental Results

We have implemented our algorithm using a combination of MATLAB and C, and tested it on several nonsynthetic stereo pairs available online [4], [14], [18]. In this section, we present the results of our experiments on a representative subset of these images and compare them to those achieved by other algorithms. Complete results can be found in [14].

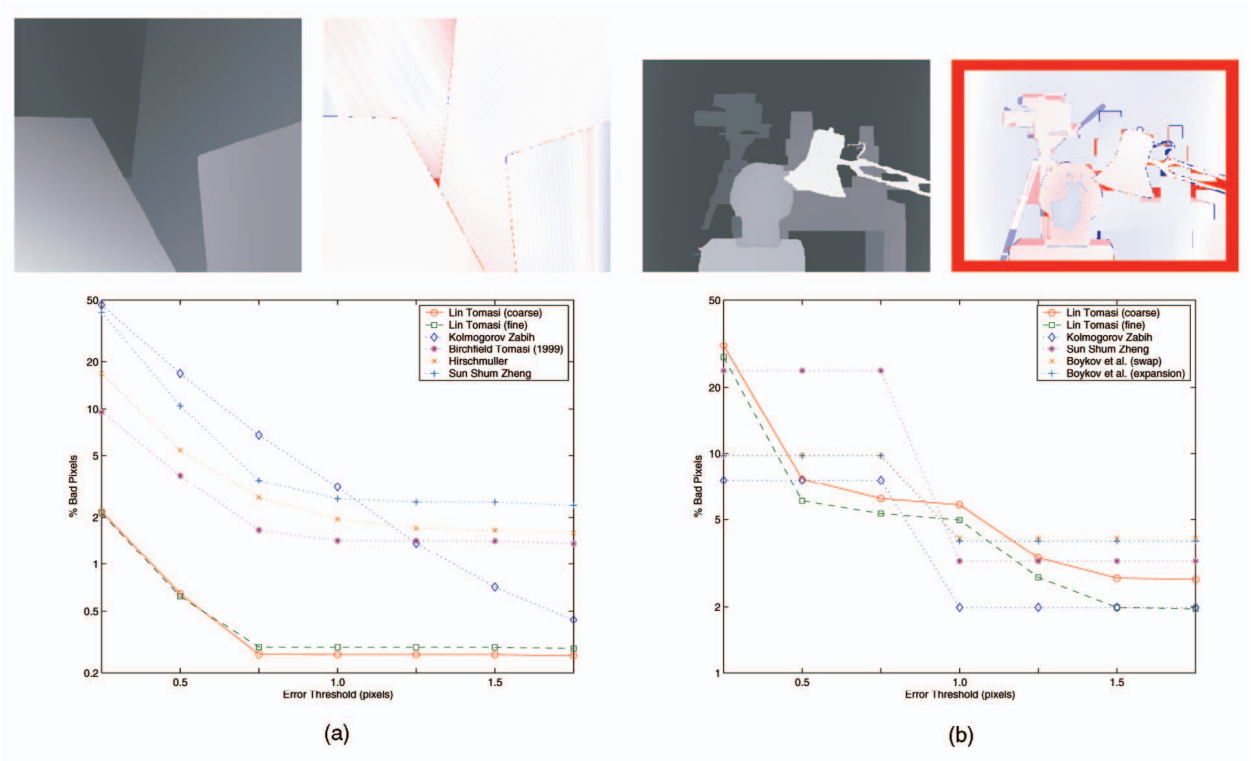

Fig. 1. (a) "Venus" and (b) "Tsukuba" stereo pairs: estimated disparities, disparity errors, and error distributions.

6.1 Quantitative Results

To evaluate the accuracy of our algorithm, we use the general framework proposed by Scharstein and Szeliski [18], who provide four sample stereo pairs with ground truth, describe a metric for comparing results against ground truth, and tabulate results for 20 algorithms. They evaluate overall results by measuring the fraction of "unoccluded" pixels where estimated and ground truth disparities differ by more than one pixel; in particular, they ignore the estimated disparity at occluded pixels. In contrast, we measure the fraction of *all* pixels (both occluded and unoccluded) where disparity error exceeds a threshold. We also consider a range of thresholds, and plot the fraction of "bad" pixels as a function thereof. In these plots, we compare our algorithm with the four that appear to be the most accurate among the others tabulated in [18]. Due to space limitations, we only show results for two of the four stereo pairs used in [18].

"Venus." This color stereo pair (Fig. 1a) shows five slanted planes, including some regions with virtually no texture. Two of the surfaces are joined by a crease edge; the remaining boundaries are all step edges. Our algorithm does extremely well, producing about five times fewer "bad" pixels than the nearest competition in [18] for a significant range of disparity error thresholds. Our largest error occurs at the corner of the V-shaped depth discontinuity, where our penalty for boundary length causes the tip of the "V" to be missed. This type of behavior is a typical result of minimizing boundary length without regard for boundary curvature and junctions.

"Tsukuba." This color stereo pair (Fig. 1b) shows a laboratory scene consisting of various planar, smoothly curved, and non-smooth surfaces. Several long and thin structures are present; our algorithm tends to oversimplify the boundaries thereof. It is notable, however, that, while the given ground truth represents all surfaces as being fronto-parallel at integer disparity, our algorithm produces curved surfaces with subpixel disparities. In particular, our algorithm models the entire head as one curved surface, with a disparity range of approximately one half pixel.

6.2 Qualitative Results

"Clorox." To verify both that our algorithm can recover crease edges and also that it needs neither color nor dense texture, we tested it on two of the grayscale stereo pairs used in Birchfield and Tomasi [4], but, due to space limitations, we only show results for one. The original version of this stereo pair shows minor photometric variations between the left and right images, as well as minor geometric distortion. Here (Fig. 2a), we use a modified version from which the photometric variations have been mostly removed. Note that the geometric distortion was left in place; this manifests itself in the apparent curvature of the floor, as recovered by our algorithm.

This stereo pair is well approximated by five slanted planes. Not all disparity edges are well marked by intensity edges and some distracting intensity edges do not accompany disparity edges. Birchfield and Tomasi's multiway cut algorithm [4] struggles with these images, deceived by the misleading intensity edges into misplacing the crease edges there. Our algorithm does not have this problem, but instead makes a different error: The Clorox box is represented with only one surface.

"Umbrella." To verify that our algorithm can reconstruct curved surfaces without dense texture, we also tested it on a stereo pair of our own creation. This stereo pair (Fig. 2b) shows five surfaces. Three are essentially planar but strongly slanted; among these, the floor has low-contrast, fine-grained texture, while both checkerboard patterns have high-contrast, coarse-grained texture. Additionally, the rear surface is a large, somewhat warped, essentially textureless sheet of cardboard and the strongly curved umbrella is composed of large, essentially textureless panels joined together by high-contrast color edges that are *not* disparity edges. The simultaneous combination of all these disparate features makes this stereo pair particularly challenging.

Although we do not have results by other algorithms for this stereo pair, we note that few of the algorithms in [18] are capable of representing smoothly curved surfaces with subpixel disparity

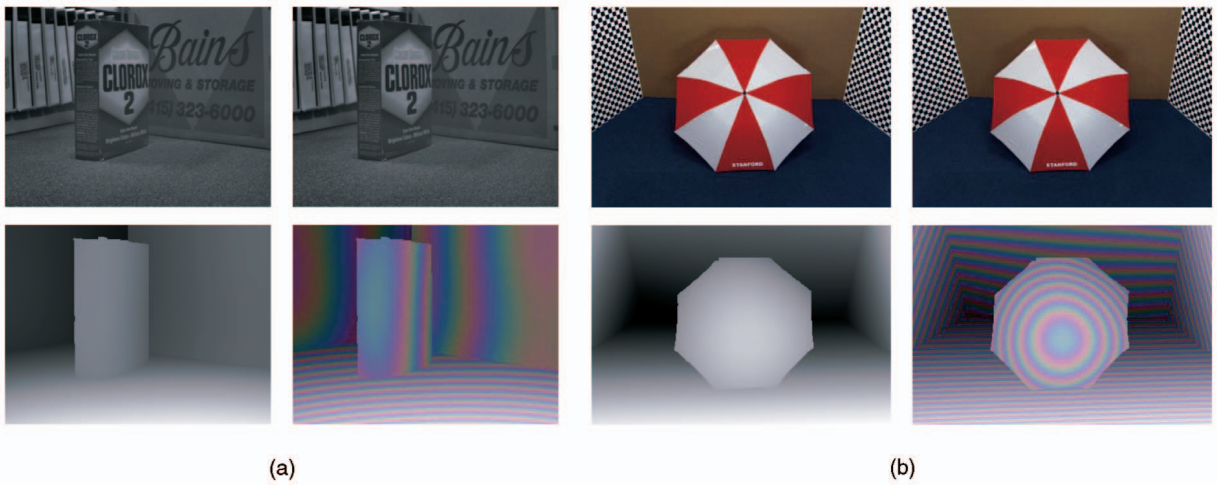

Fig. 2. (a) "Clorox" and (b) "Umbrella" stereo pairs: input images, grayscale estimated disparities, and color-coded estimated disparities. *(All trademarks remain the property of their respective owners. All trademarks and registered trademarks are used strictly for educational and scholarly purposes and without intent to infringe on the mark owners.)*

values and, among those, fewer still readily reproduce sharp discontinuities in the disparity map.

Our algorithm correctly segments the scene into five smooth surfaces, each of which is represented by a real-valued disparity map that contains no kinks or creases, even in the presence of strong color edges that might suggest otherwise. Our algorithm places boundaries accurately at crease edges as well as at edges accompanied by significant occlusion regions and qualitatively recovers the curvature of both the background and the umbrella with very little help from texture.

7 Discussion and Future Work

The quantitative and qualitative results presented suggest that, for scenes consisting of smooth surfaces, our algorithm produces very accurate reconstructions, with subpixel disparity values and explicit and precise localization of boundaries, whether the surfaces are planar or curved, textured or untextured, high-contrast or low-contrast, color or grayscale. Despite this achievement, however, there is nonetheless much room for improvement.

The most limiting aspect of our current implementation is its model of surfaces. Although our model works quite well for most of the results that we have presented, it can be overly restrictive for scenes whose surfaces are less smooth. To be able to handle surfaces with more shape detail, our implementation should use a much finer grid for the control points of the splines that define surface shape. This would likely require a more refined model of surface smoothness.

Finally, we note that many of the parameters of our algorithm, controlling such things as coarseness of segmentation and amount of surface shape detail, do not have to be constant, but can in fact vary from surface to surface and even within the same surface. If, in addition to the disparity maps and segmentation, these parameters themselves could be estimated separately and adaptively for each surface, we believe that our algorithmic framework would be capable of producing accurate results for a much wider variety of scenes.

References

[1] S. Baker, R. Szeliski, and P. Anandan, "A Layered Approach to Stereo Reconstruction," *Proc. Computer Vision and Pattern Recognition*, pp. 434-441, 1998.

[2] S. Barnard and M. Fischler, "Computational Stereo," *Computing Surveys*, vol. 14, pp. 553-572, 1982.

[3] P.N. Belhumeur, "A Bayesian Approach to Binocular Stereopsis," *Int'l J. Computer Vision*, vol. 19, pp. 237-260, 1996.

[4] S. Birchfield and C. Tomasi, "Multiway Cut for Stereo and Motion with Slanted Surfaces," *Proc. Int'l Conf. Computer Vision*, pp. 489-495, 1999, http://vision.stanford.edu/~birch/multiwaycut/.

[5] A.F. Bobick and S.S. Intille, "Large Occlusion Stereo," *Int'l J. Computer Vision*, vol. 33, no. 3, pp. 181-200, 1999.

[6] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization Via Graph Cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, Dec. 2001.

[7] I.J. Cox, S.L. Hingorani, S.B. Rao, and B. Maggs, "A Maximum Likelihood Stereo Algorithm," *Proc. Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 542-567, 1996.

[8] T. Darrell and A. Pentland, "Cooperative Robust Estimation Using Layers of Support," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 474-487, May 1995.

[9] U.R. Dhond and J.K. Aggarwal, "Structure from Stereo—A Review," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1489-1510, 1989.

[10] D. Geiger, B. Ladendorf, and A. Yuille, "Occlusions and Binocular Stereo," *Int'l J. Computer Vision*, vol. 14, no. 3, pp. 211-226, 1995.

[11] H. Ishikawa and D. Geiger, "Occlusions, Discontinuities, and Epipolar Lines in Stereo," *Proc. European Conf. Computer Vision*, vol. 1, pp. 232-249, 1998.

[12] T. Kanade and M. Okutomi, "A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920-932, Sept. 1994.

[13] V. Kolmogorov and R. Zabih, "Computing Visual Correspondence with Occlusions Using Graph Cuts," *Proc. Int'l Conf. Computer Vision*, pp. 508-515, 2001.

[14] M. Lin, "Surfaces with Occlusions from Layered Stereo," PhD thesis, Stanford Univ., 2002, http://robotics.stanford.edu/~michelin/layered _stereo/.

[15] D. Marr and T. Poggio, "Cooperative Computation of Stereo Disparity," *Science*, vol. 194, pp. 283-287, 1976.

[16] T. Poggio, V. Torre, and C. Koch, "Computational Vision and Regularization Theory," *Nature*, vol. 317, pp. 314-319, 1985.

[17] S. Roy and I. Cox, "A Maximum-Flow Formulation of the *N*-Camera Stereo Correspondence Problem," *Proc. Int'l Conf. Computer Vision*, pp. 492-499, 1998.

[18] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *Int'l J. Computer Vision*, vol. 47, pp. 7-42, 2002, http://www.middlebury.edu/stereo/.

[19] R. Szeliski and J. Coughlan, "Spline-Based Image Registration," *Int'l J. Computer Vision*, vol. 22, no. 3, pp. 199-218, 1997.

[20] H. Tao, H.S. Sawhney, and R. Kumar, "A Global Matching Framework for Stereo Computation," *Proc. Int'l Conf. Computer Vision*, pp. 532-539, 2001.

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