Surfaces with Occlusions from Layered Stereo

Michael H. Lin
Stanford University
michelin@cs.stanford.edu

Carlo Tomasi
Duke University
tomasi@cs.duke.edu

Abstract

Although steady progress has been made in recent stereo algorithms, producing accurate results in the neighborhood of depth discontinuities remains a challenge. Moreover, among the techniques that best localize depth discontinuities, it is common to work only with a discrete set of disparity values, hindering the modeling of smooth, non-fronto-parallel surfaces.

We propose to estimate scene structure as a set of smooth surface patches. The disparities within each patch are modeled by a spline, while the extent of each patch is represented by a pixelwise labeling of the source images. Disparities and extents are alternately estimated in an iterative, energy minimization framework. Segmentation is via graph cuts, aided by image gradients. Input images are treated symmetrically, and occlusions are addressed explicitly. Promising experimental results are presented.

1. Introduction

The foundations of binocular 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 this correspondence problem. Given geometric calibration of the images, the *epipolar constraint* reduces the search for possible point matches from two dimensions to one. Given photometric calibration of the images, and assuming diffuse surfaces, *color constancy* further narrows the possibilities to points that look alike. Marr and Poggio [21] proposed two additional constraints that mitigate the ill-posedness of the stereo problem: *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."

Many stereo algorithms are based upon these four constraints. Of these, the former two are relatively straightforward, but the manner in which the latter two are applied varies greatly [3, 13, 23]. We propose a three-axis categorization of binocular stereo algorithms according to their interpretation of continuity and uniqueness. In the following subsections, we list last, for all three axes, that category which we consider to be the most preferable.

This material is based upon work supported by the National Science Foundation under Grant No. 0222516.

1.1. Continuity

The first axis describes the modeling of continuity over disparity values within smooth surface patches.

Constant. Every point within any one smooth surface is assigned the same disparity value. Examples include traditional SSD correlation, as well as [7, 17, 18, 21].

Discrete. Disparities are chosen from a finite set of possible values, but with multiple distinct values permitted within each surface. Examples include [4, 16, 22, 27].

Real. Disparities within each smooth surface vary over the real numbers. Examples include [1, 2, 5, 25, 26].

1.2. Discontinuity

The second axis describes the treatment of discontinuities at the boundaries of smooth surface patches. Specifically, the penalty assigned to a discontinuity is examined as a function of the size of the jump of the discontinuity.

Free. Discontinuities are not specifically penalized. Examples include traditional SSD correlation, as well as [17, 21, 27].

Infinite. Discontinuities are penalized infinitely; i.e., they are disallowed. The recovered disparity map is smooth everywhere, although potentially not uniformly so. Examples include [1, 25].

Convex. Discontinuities are allowed but penalized with a finite, positive, convex cost function. 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 include [16, 22, 26].

Non-convex. Discontinuities are allowed but penalized with a non-convex cost function. The resulting discontinuities usually 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 include [4, 10, 11, 14].

1.3. Uniqueness

The third axis describes the application of uniqueness to the occlusions that accompany depth discontinuities.

One-way. Uniqueness is assumed within a chosen reference image, but not considered within the other. That is, each location in the reference image is assigned at most one disparity, but the disparities at multiple locations in the reference image may point to the same location in the other image. Examples include traditional SSD correlation, as well as [5, 10, 17].

Asymmetric two-way. Uniqueness is encouraged for both images, but the two images are treated unequally. Examples include [2, 7, 21, 26, 27].

Symmetric two-way. Uniqueness is enforced symmetrically. Examples include [4, 14, 16, 18].

1.4. Overview

In this paper, we propose an algorithm that lies in the most preferable category of all three axes; to the authors' knowledge, it 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 better localization in depth of surface interiors via subpixel disparity estimation, and better localization in the image plane of surface boundaries via the symmetric treatment of images with proper handling of occluded regions. Our method is loosely based upon [5], with the most significant extension being our simultaneous, consistent estimation of both left and right disparity maps.

In Section 2, we describe our mathematical model of the stereo problem and solutions thereof. In Sections 3 and 4, we describe surface fitting and boundary localization, respectively. In Section 5, we describe the overall optimization algorithm. In Section 6, we present some promising qualitative and quantitative experimental results. Finally, in Section 7, we offer some concluding remarks.

2. Preliminaries

Belhumeur argued that "depth, surface orientation, occluding contours, and creases should be estimated simultaneously" [4]. To do so, we use a layered model [12] of possible solutions to the stereo problem. Because we assume opacity, each image point can be assigned to at most one surface; this enables the extent of all surfaces to be represented by a single labeling of image points.

2.1. Mathematical Abstraction

Our algorithm follows the common practice of assuming that input images have been normalized with respect to 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, where 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:

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

In other words, these functions are the independent unknowns that are to be estimated.

The segmentation function f specifies to which one of n surfaces, if any, each image location "belongs," where belonging implies that some world point (a) projects to the image location in question, and (b) 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 $(\neg\text{`LEFT'} = \text{`RIGHT'})$ and vice versa. That is, for each surface k, $m[k]$ maps each location in one image to the corresponding location in the other image. Note that, for all k, $d[k]$ and $m[k]$ are both defined for all (x, y, t), regardless of the value of $f(x, y, t)$. Furthermore, for standard camera configurations, $d[k]$ will generally be positive in the right image and negative in the left image.

Thus, the interpretation of this model is, for all p,

$$\begin{aligned} f(p) = k \text{ with } k > 0 &\Rightarrow p \text{ corresponds to } m[k](p); \\ f(p) = 0 &\Rightarrow p \text{ corresponds to nothing.} \end{aligned}$$

That is, a hypothesized solution specfies a set of correspondences between left and right image locations, where each image location is a member of at most one correspondence.

2.2. Desired Properties

Given this abstract representation of a solution, how can we evaluate any particular hypothesized solution? We propose three properties that characterize a "good" solution: non-triviality, smoothness, and consistency.

Non-triviality. Good solutions should explain, rather than ignore, input data. For example, any two input images could be interpreted as views of two painted, planar surfaces, each presented to one camera. Such a trivial interpretation, yielding no correspondence for any image location, would be valid but undesirable. In general, we expect that a correspondence exists for "most" image locations:

$$\text{for most } p: \quad f(p) > 0$$

Moreover, although color constancy is sometimes violated (e.g., due to specularities), and smoothness is needed to fill in the gap, a solution that supposes a perfectly smooth surface, at the expense of violating color constancy everywhere, is also not desirable. In other words, we expect that color constancy holds for "most" image locations:

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

Smoothness. Because the disparity maps $d[k]$ are continuous-valued functions, we take smoothness of $d[k]$ to mean differentiability, with the magnitude of higher derivatives being relatively small.

Because the segmentation function f can only take on the integer values $0 \dots N$, it is piecewise constant, with line-like boundaries separating those pieces. We take smoothness of f to mean simplicity of these boundaries, with the total boundary length being relatively small.

Consistency. Correspondence of image locations should be bidirectional. In other words, if points p and q are images of the same world point, then each corresponds to the other; otherwise, neither corresponds to the other. It would make no sense to say that p corresponds to q but that q does not correspond to p.

Within our mathematical formulation, this constraint, applied to surface shape, gives, for all k, p:

$$m[k](m[k](p)) = p \tag{1}$$

which implies a constraint on each $d[k]$. In particular, for each k, given one of $d[k](\cdot, \cdot, \text{`LEFT'})$ or $d[k](\cdot, \cdot, \text{`RIGHT'})$, the other is uniquely determined.

Regarding segmentation, we also have the constraint on f that, for all p,

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

Ideally, these consistency constraints should be satisfied exactly, but for computational purposes, we merely attempt to maximize consistency.

2.3. Energy Minimization

We formalize the stereo problem in the framework of energy minimization. In general, energy minimization approaches split a problem into two parts: defining the cost of all hypothesized solutions, and finding the best solution by minimizing that cost. This separation facilitates the use of general-purpose minimization techniques, enabling more focus upon the unique aspects of the specific application.

For our application, 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 (see Table 1). These terms are developed in the next two sections; total energy is a positive linear combination of these terms.

	disparity maps	segmentation
non-triviality	E_match_I	$E\_unassigned$
smoothness	E_smooth_d	E_smooth_f
consistency	E_match_d	E_match_f

Table 1. Contributions to energy.

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 explain our model of smooth surfaces; formulate the three energy terms that encourage surface non-triviality, smoothness, and consistency; and discuss the minimization of these energy terms.

3.1. Surface Model

We model the disparity map of each surface as a bicubic spline. This gives us the flexibility to represent a wide range of slanted or curved surfaces with subpixel disparity precision, while ensuring that disparity values and gradients vary smoothly over the surface. The control points of the spline are placed on a regular grid with fixed image coordinates (but variable disparity value). 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 $d[k]$ to take the form of a bicubic spline with control points on a fairly coarse, uniform rectangular grid:

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

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

In general, the spacing of the grid of spline control points should be fine enough so that surface shape details can be recovered. In our experiments, we use a fixed, 5×5 grid for each view (left and right) of each hypothesized surface.

3.2. Surface Non-triviality

This energy term, often called the "data term" in other literature, expresses the assumption of color constancy:

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

where $g(v) = v^T \cdot A \cdot v$, and A is a space-variant 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 we define

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

where ϵ and σ are small constants.

Note that, ideally, E_match_I would be defined as an integral over all $p \in \mathcal{I}$. However, for computational convenience, we approximate the integral with a finite sum over discrete pixel positions, for this and other energy terms. This is reasonable if the summand is spatially smooth.

3.3. Surface Smoothness

Since we consider smooth surfaces to be more likely to occur, we would like to quantify and penalize any deviation from perfect smoothness. We take the class of perfectly smooth surfaces to be the set of planar surfaces (including both fronto-parallel and slanted planes). The usual measure of deviation from planarity is *quadratic variation* [6], but this measure has the disadvantage of using second derivatives, which can be overly susceptible to high-frequency, local deviations. Instead, we add an energy term (in addition to restricting $d[k]$ to take the form of a spline) which, loosely speaking, is proportional to the global "variance" of the surface slope:

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

where the mean is taken over all discrete pixel positions p.

In our experiments, this energy term is given a very small weight, and mainly serves to accelerate the convergence of numerical optimization by shrinking the nullspace of the total energy function. This term does not prevent surfaces from being non-planar.

3.4. Surface Consistency

For perfect consistency, a surface should have left and right views that coincide exactly with one another [Equation (1)]. To quantify and discourage any non-coincidence, we take

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

or equivalently,

$$E\_match\_d[k] = \sum_p \big(d[k](p) + d[k](m[k](p)) \big)^2$$

which, intuitively, measures the distance between the surfaces defined by the left and right views.

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 two terms $E\_unassigned$ and E_smooth_f are shown to depend only on f, and the remaining term E_match_f is shown to depend smoothly on $d[k]$. Therefore, 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. Experimentally, this algorithm exhibits more reliable convergence than the quasi-Newton methods with line searches, and although it requires the calculation of the Hessian, in our implementation, that expense is a small fraction of the total computational requirements.

In this section, we have shown how to minimize total energy by varying each $d[k]$ individually. For each $k > 0$, we call minimizing 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 explain our model of segmentation; formulate the three energy terms that encourage segmentation non-triviality, smoothness, and consistency; and discuss the minimization of these energy terms.

4.1. Segmentation by Graph Cuts

Boykov, Veksler, and Zabih [10] showed that certain labeling problems can be formulated as energy minimization problems and solved efficiently by repeatedly using maximum flow techniques to find 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 a set of interacting pairs of items. The methods of [10] 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

$$E(f) = \sum_{(p,q) \in \mathcal{N}} V_{p,q}(f_p, f_q) + \sum_{p \in \mathcal{P}} D_p(f_p) \qquad (3)$$

be minimized. Individual energies D_p should be nonnegative but can otherwise be arbitrary; interaction energies $V_{p,q}$ should be either *semi-metric* or *metric*. Kolmogorov and Zabih [19] generalize these results, deriving necessary and sufficient conditions on the form of the energy E in order for it to be minimizable with graph cut methods.

Given an energy function in the form of Equation (3) satisfying the relevent conditions, the methods of [10] are extremely effective at finding a minimizing labeling, in terms of both computational complexity and the optimality of the final solution. For this reason, we have chosen to use these methods to solve our segmentation subproblem.

This generic formulation maps to our stereo problem as follows: the labels are the integers $0 \dots N$ (the possible values of the segmentation function f), and the items are the pixels of each input image. The individual energies stem from testing color constancy at varying disparities, and the interaction energies stem from smoothness and consistency.

Because of computational considerations, in our algorithm, all visible world points are assumed to be completely opaque. Furthermore, pixels are prohibited from being split spatially among several surfaces, but instead are constrained to be indivisible, forcing surface boundaries to lie on pixel boundaries. Thus, in representing the continuous-domain segmentation function f on a discrete grid of pixels, we essentially perform nearest-neighbor interpolation:

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

where F is defined on an integer lattice.

4.2. Segmentation Non-triviality

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 as a functional of f, with all $d[k]$ held constant. However, note that since $g(\cdot)$ is non-negative, E_match_I is trivially minimized by $f(p) \equiv 0$. To discourage such solutions with large unassigned regions, we add a fixed penalty for each unassigned pixel:

$$E\_unassigned = \sum_p \begin{cases} 1 & \text{if } f(p) = 0, \\ 0 & \text{otherwise.} \end{cases}$$

While it is not uncommon among stereo algorithms to have an occlusion penalty such as this one, it should be noted that this term is not solely for handling occlusions; for example, it also limits the influence of gross outliers in the input image data.

Thus, the underlying segmentation problem, for the moment ignoring smoothness and consistency, is to find the labeling f that minimizes a linear combination of E_match_I and $E\_unassigned$. Put into the form of Equation (3), this corresponds to the following definition of individual pixel energies:

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

where $\lambda\_unassigned$ is a constant.

4.3. Segmentation Smoothness

In addition to minimizing pointwise costs, we would also like to encourage a simple segmentation with "smooth" boundaries of surface extents. There are several attributes that can be used to formalize this notion, including boundary length and curvature [6]. We choose to minimize boundary length without separate regard for boundary curvature, because it is simpler to optimize, and works fairly well in practice.

Furthermore, there is an expectation that boundaries will be correlated with monocular image features (called "static cues" in [10]). Thus, we would like to reward the placement of boundaries at edge-like image locations. There are many ways 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.

We 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

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

for p adjacent to q, where τ is a constant, and ∇I and A are both evaluated at the subpixel position $(p + q)/2$.

Put into the form of Equation (3), $E\_smooth\_f[k]$ corresponds to this penalty function:

$$V_{p,q}(f_p, f_q) = \lambda\_smooth\_f \cdot w_s(p, q) \cdot \sum_{k>0} T\big(f_p = k \text{ xor } f_q = k\big)$$

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

4.4. Segmentation Consistency

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

$$E\_match\_f[k] \approx \sum_p \begin{cases} 1 & \text{if } f(p) = k \text{ xor } f(m[k](p)) = k, \\ 0 & \text{otherwise,} \end{cases}$$

which approximates the area of inconsistent regions, where (2) does not hold. As before, this term should ideally be defined with an integral, but in this case, a naive finite sum is *not* an adequate substitute, as we explain in [20]. Instead, we take

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

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

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

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

Put into the form of Equation (3), $E\_match\_f[k]$ corresponds to this penalty function:

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

for p and q in corresponding scanlines, where

$$w_c[k](p,q) = \hat{h}\big(m[k](p) - q\big) + \hat{h}\big(m[k](q) - p\big),$$

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 (E_smooth_d and E_match_d) are independent of f. In this section, the remaining four terms are written in the form of Equation (3); moreover, the penalty functions $V_{p,q}$ can be verified to be metric. Hence, the total energy as a function of f can be optimized with graph cut methods [10, 19].

We use a modified version of the expansion algorithm of [10]. This greedy algorithm is built from expansion moves, and gets its power from the generality of such moves: an expansion move on a label k finds the best configuration reachable by relabeling *any* subset of pixels with k. Our modification is to precede each expansion with a contraction of the same label, 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 complete problem of simultaneously determining surface shape in the form of disparity maps, and surface support in the form of segmentation, when both are initially unknown.

Our overall algorithm is built from the surface-fitting and segmentation steps that were defined in Sections 3 and 4. Because each of these steps decreases total energy, given a reasonable initial hypothesis, iterating these steps until convergence might give a reasonable final solution. However, there are a few complications.

During the course of component-wise optimization using the surface-fitting and segmentation steps, it is quite possible to reach an undesirable local minimum. These problematic 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. It thus requires careful initialization to avoid getting into such situations. Our algorithm requires that, alongside the input image pair, the range of possible disparities also be specified. The initial hypothesis is then formed by placing one fronto-parallel surface at every integer disparity within that range. 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, and a true local minimum is reached, a *merge step* will generally remedy the situation.

To take a merge step, the algorithm first saves a checkpoint of the current state. It then forcefully removes 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 checkpoint, in which case the merge succeeds and the checkpoint is discarded, or the total energy plateaus above that of the checkpoint, in which case the merge fails and the checkpoint 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 hypothesized surface:
       - Attempt to merge it.

       until one merge succeeds *or* all merges fail.

   until all merges fail.
3. Optionally "fill in" unmatched regions (see [20]).

6. Experimental Results

We have implemented our algorithm using a combination of Matlab and C, and tested it on several stereo pairs available online [5, 23]. Due to space limitations, we only present a representative subset of our results; complete results can be found in [20].

6.1. Quantitative Results

To evaluate the accuracy of our algorithm, we use the general framework proposed by Scharstein and Szeliski [23], who provide four sample stereo pairs with ground truth, describe a metric for comparing results against ground truth, and tabulate results for 20 algorithms.

Scharstein and Szeliski [23] evaluate overall results by measuring the fraction of "unoccluded" pixels at which estimated and ground truth disparities differ by more than one pixel. In contrast, we retain the occluded pixels, and measure the fraction of *all* pixels at which disparity error exceeds a threshold. We also consider a range of thresholds, and plot the fraction of "bad" pixels as a function thereof.

For our comparison, we used two sets of parameters, differing only in the relative weight of E_smooth_f within total energy, to produce slightly coarser or finer segmentations. In the aforementioned plots, we summarize results for both sets of parameters, and compare them to results obtained by the four algorithms that appear to be the most accurate among the remaining algorithms tabulated in [23].

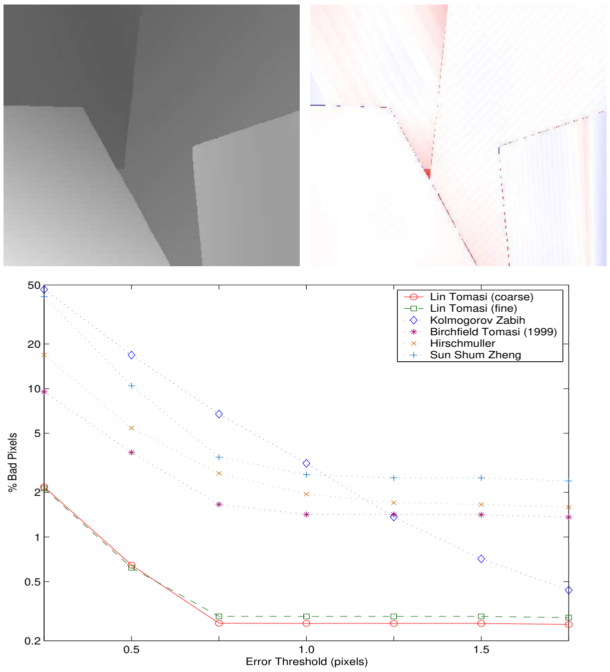

Figure 1. "Venus." Top: our result; disparity error. Bottom: errors for our algorithm and [5, 15, 18, 24].

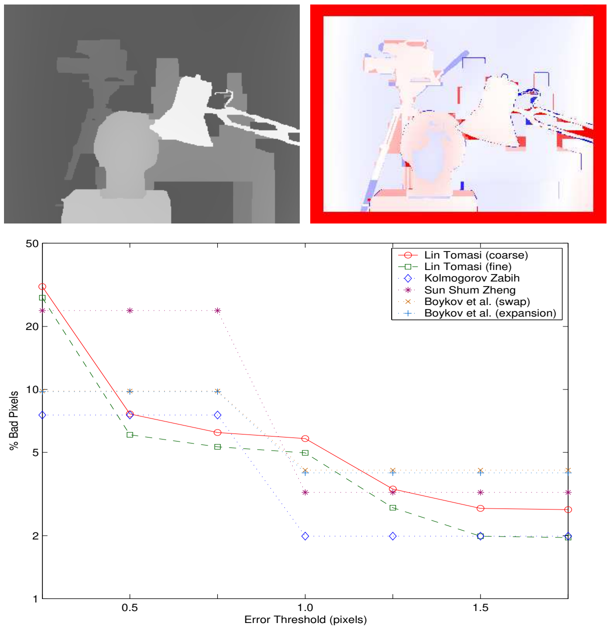

Figure 2. "Tsukuba." Top: our result; disparity error. Bottom: errors for our algorithm and [8, 9, 18, 24].

Venus. This stereo pair (Figure 1) shows five slanted planes with varying amounts of texture. Regarding disparity estimation, our algorithm does extremely well, with the only gross error occurring at the corner of a V-shaped depth discontinuity, where our penalty for boundary length causes the tip of the "V" to be missed. Regarding segmentation, however, our algorithm recovers only four distinct surfaces, missing the vertical crease in the sports page.

Tsukuba. This stereo pair (Figure 2) shows a laboratory scene consisting of various planar, smoothly curved, and non-smooth surfaces; boundaries are fairly complex, with several long and thin structures. Our algorithm tends to over-simplify these boundaries, even with the parameter set that prefers a finer segmentation. However, it is notable 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 the nose and chin being closest to the camera, and the left and right sides of the head being farther by approximately one half pixel of disparity.

6.2. Qualitative Results

Among the four stereo pairs used in the benchmark by Scharstein and Szeliski, three consist solely of planar surfaces, with all but one boundary being discontinuous in depth. However, our algorithm was designed to be able to handle curved surfaces and crease edges as well, so to test its ability to do so, we photographed our own scene with those features. Although we were unable to obtain ground truth data, we present the results for qualitative evaluation.

This stereo pair (Figure 3) shows five surfaces. The floor has some fine-grained texture, and is planar. The right checkerboard pattern is also planar, but the left checkerboard pattern is very slightly warped. The rear surface is a more severely warped, unmarked sheet of cardboard. The two-tone umbrella is obviously curved, and rests on the floor, but does not contact the rear sheet of cardboard.

Our algorithm correctly segments the scene into five surfaces, and places boundaries accurately at crease edges as well as at edges accompanied by occlusion regions. Our algorithm qualitatively recovers the warped shape of the background and the curvature of the umbrella, both with very little help from texture.

7. Conclusion

The quantitative and qualitative results presented suggest that, for scenes consisting of smooth surfaces, our algorithm obtains accurate results, with subpixel disparity values, and explicit and precise localization of boundaries. For other scenes, however, further work is needed.

The most limiting aspect of the current implementation is its model of surfaces. To be able to reconstruct surfaces with finer detail, it should use a finer grid of control points for the splines that define surface shape. Implementing this efficently would likely require a more scalable technique for optimization that does not require an exact Hessian.

As mentioned in Section 5, one failure mode of the current algorithm occurs in some cases when a single hypoth-

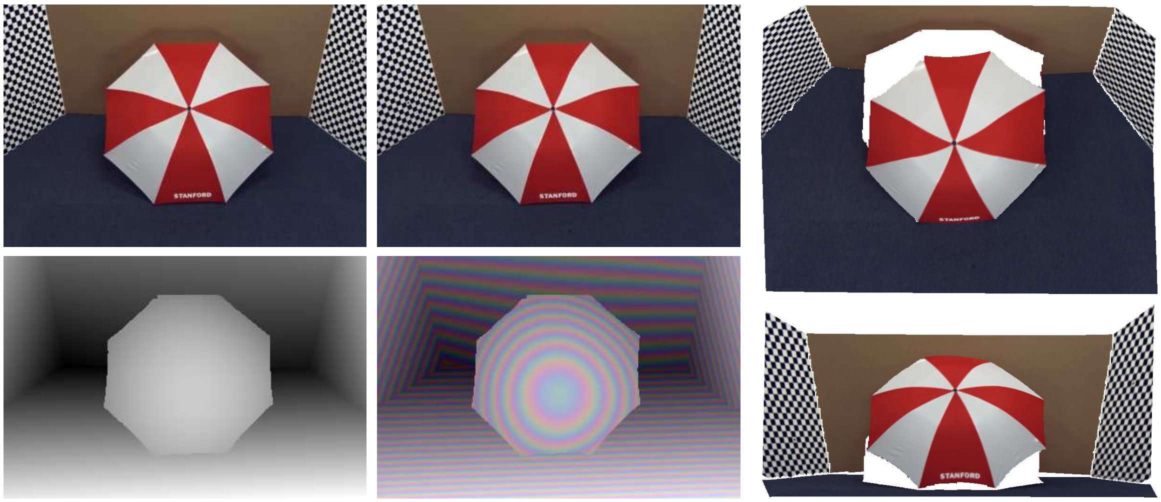

Figure 3. "Umbrella." Top: input. Bottom: our result; with isocontours emphasized. Right: Reprojected views.

esized surface spans what is actually multiple surfaces. That is, although our algorithm attempts to ensure that no merging of surfaces could result in a decrease of total energy, it does not do the same for the splitting of surfaces. It would be useful if some method were devised for automatic splitting as well as automatic merging of surfaces.

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 could vary from surface to surface, and even within the same surface. If, in addition to the disparity maps and segmentation, these parameters themselves could also be estimated adaptively for each surface, we believe that our algorithmic framework would be capable of producing accurate results for a wide variety of scenes.

References

[1] Y. S. Akgul and C. Kambhamettu. Recovery and tracking of continuous 3D surfaces from stereo data using a deformable dual-mesh. In *ICCV*, pp 765–772, 1999.

[2] S. Baker, R. Szeliski, and P. Anandan. A layered approach to stereo reconstruction. In *CVPR*, pp 434–441, 1998.

[3] S. T. Barnard and M. A. Fischler. Computational stereo. *Computing Surveys*, 14:553–572, 1982.

[4] P. N. Belhumeur. A Bayesian approach to binocular stereopsis. *IJCV*, 19:237–260, 1996.

[5] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *ICCV*, pp 489–495, 1999.

[6] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, Cambridge, MA, 1987.

[7] A. F. Bobick and S. S. Intille. Large occlusion stereo. *IJCV*, 33(3):181–200, 1999.

[8] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *CVPR*, pp 648–655, 1998.

[9] Y. Boykov, O. Veksler, and R. Zabih. Approximate energy minimization with discontinuities. In *EMMCVPR*, pp 205–220, 1999.

[10] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11), 2001.

[11] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. Maggs. A maximum likelihood stereo algorithm. *CVIU*, 63(3):542–567, 1996.

[12] T. Darrell and A. Pentland. Cooperative robust estimation using layers of support. *PAMI*, 17(5):474–487, 1995.

[13] U. R. Dhond and J. K. Aggarwal. Structure from stereo – a review. *IEEE SMC*, 19(6):1489–1510, 1989.

[14] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. *IJCV*, 14(3):211–226, 1995.

[15] H. Hirschmüller, P. R. Innocent, and J. Garibaldi. Real-time correlation-based stereo vision with reduced border errors. *IJCV*, 47:229–246, 2002.

[16] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *ECCV*, vol 1, pp 232–249, 1998.

[17] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *PAMI*, 16(9):920–932, 1994.

[18] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *ICCV*, pp 508–515, 2001.

[19] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *ECCV*, vol 3, pp 65–81, 2002.

[20] M. Lin. *Surfaces with Occlusions from Layered Stereo*. PhD thesis, Stanford University, 2002. See also http://robotics.stanford.edu/˜michelin/layered_stereo/ .

[21] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, 1976.

[22] S. Roy and I. Cox. A maximum-flow formulation of the *N*-camera stereo correspondence problem. In *ICCV*, pp 492–499, 1998.

[23] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002. See also http://www.middlebury.edu/stereo/ .

[24] J. Sun, H.-Y. Shum, and N.-N. Zheng. Stereo matching using belief propagation. In *ECCV*, vol 2, pp 510–524, 2002.

[25] R. Szeliski and J. Coughlan. Spline-based image registration. *IJCV*, 22(3):199–218, 1997.

[26] H. Tao, H. S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *ICCV*, pp 532–539, 2001.

[27] C. L. Zitnick and T. Kanade. A cooperative algorithm for stereo matching and occlusion detection. *PAMI*, 22(7):675–684, 2000.