Shape and the stereo correspondence problem

Abhijit S. Ogale and Yiannis Aloimonos

Computer Vision Laboratory, Institute for Advanced Computer Studies
Dept. of Computer Science, University of Maryland, College Park, MD 20742, USA

Abstract

We examine the implications of shape on the process of finding dense correspondence and half-occlusions for a stereo pair of images. The desired property of the disparity map is that it should be a piecewise continuous function which is consistent with the images and which has the minimum number of discontinuities. To zeroth order, piecewise continuity becomes piecewise constancy. Using this approximation, we first discuss an approach for dealing with such a fronto-parallel shapeless world, and the problems involved therein. We then introduce horizontal and vertical slant to create a first order approximation to piecewise continuity. In particular, we emphasize the following geometric fact: a horizontally slanted surface (i.e., having depth variation in the direction of the separation of the two cameras) will appear horizontally stretched in one image as compared to the other image. Thus, while corresponding two images, N pixels on a scanline in one image may correspond to a different number of pixels M in the other image. This leads to three important modifications to existing stereo algorithms: (a) due to unequal sampling, existing intensity matching metrics must be modified, (b) unequal numbers of pixels in the two images must be allowed to correspond to each other, and (c) the uniqueness constraint, which is often used for detecting occlusions, must be changed to an interval uniqueness constraint. We also discuss the asymmetry between vertical and horizontal slant, and the central role of non-horizontal edges in the context of vertical slant. Using experiments, we discuss cases where existing algorithms fail, and how the incorporation of these new constraints provides correct results.

1. Introduction

The dense correspondence problem consists of finding a unique mapping between the points belonging to two images of the same scene. If the camera geometry is known, the images can be rectified, and the problem reduces to the stereo correspondence problem, where points in one image can correspond only to points along the same scanline in the other image. If the geometry is unknown, then we have the optical flow estimation problem. In both cases, regions in one image which have no counterparts in the other image, are referred to as occlusions (or more correctly as *half-occlusions*). In this paper, we demonstrate that scene shape has profound implications for any process of establishing point correspondence and occlusion detection. In particular, we show that correspondence, segmentation, occlusion detection, and shape estimation have to be done in concert.

1.1. Previous work

There exists a considerable body of work on the dense stereo correspondence problem. Scharstein and Szeliski [1] have provided an exhaustive comparison of dense stereo correspondence algorithms. Most algorithms generally utilize local measurements such as image intensity (or color) and phase, and aggregate information from multiple pixels using smoothness constraints. The simplest method of aggregation is to minimize the matching error within rectangular windows of fixed size [2]. Better approaches utilize multiple windows [3, 4], adaptive windows [5] which change their size in order to minimize the error, shiftable windows [6, 7, 8], or predicted windows [9], all of which give performance improvements at discontinuities.

Global approaches to solving the stereo correspondence problem rely on the extremization of a global cost function or energy. The energy functions which are used include terms for local property matching ('data term'), additional smoothness terms, and in some cases, penalties for occlusions. Depending on the form of the energy function, the most efficient energy minimization scheme can be chosen. These include dynamic programming [10], simulated annealing [11, 12], relaxation labeling [13], non-linear diffusion [14], maximum flow [15] and graph cuts [16, 17]. Maximum flow and graph cut methods provide better computational efficiency than simulated annealing for energy functions which possess a certain set of properties. Some of these algorithms treat the images symmetrically and explicitly deal with occlusions (e.g., [17]). The uniqueness

constraint [18] is often used to find regions of occlusion. Egnal and Wildes [19] provide comparisons of various approaches for finding occlusions.

The issue of recovering a piecewise planar description of a scene has been previously explored in the context of stereo (e.g., [10]) and motion (e.g., [20]). Recently, some algorithms [21] have explicitly incorporated the estimation of slant while performing the estimation of dense horizontal disparity. Lin and Tomasi [22] explicitly model the scene using smooth surface patches and also find occlusions; they initialize their disparity map with integer disparities obtained using graph cuts, after which surface fitting and segmentation are performed repeatedly. Previously, Devernay and Faugeras [23] have used local image deformations to obtain differential properties of 3D shapes directly.

1.2. Organization of the paper

When we compute the disparity map of a real scene, we would like to model it ideally as a *piecewise continuous* function which explains the observed images and has the minimum possible number of pieces (minimum segmentation). The simplest approximation to piecewise continuity is *piecewise constancy*. In this paper, Section 2 deals with the problem of estimating correspondence and performing disparity segmentation in a Mondrian flatland which consists of flat fronto-parallel surfaces only. In this shape-free world, we argue that correspondence and segmentation, which appear to be chicken-and-egg problems, can only be solved together. We provide a simple algorithm which captures the spirit of this approach and also estimates occlusions using the uniqueness constraint.

Departing from Flatland, we then examine the central role of shape in establishing correspondence, and show why shape must also be estimated concurrently with the correspondence and the segmentation. Section 3 addresses the stereo correspondence problem in the presence of horizontally slanted surfaces (a part of this section appeared in [24]). We lay emphasis on the following geometric effect: a horizontally slanted surface (i.e., having depth variation in the direction of the separation of the two cameras) will appear horizontally stretched in one image as compared to the other image. Thus, when we correspond two images, N pixels on a scanline in one image must be allowed to correspond with a different number of pixels M in the other image. Furthermore, it is evident that the intensity function on the true horizontally slanted scene surface is sampled differently by the two cameras, which is another low-level effect which needs to be addressed. The uniqueness constraint, which is often used to find occlusions by forcing a one-to-one correspondence between pixels within regions which are visible in both views, does not hold true in the presence of horizontally slanted surfaces, since a N-to-M correspondence is possible; we show how the uniqueness constraint must be reformulated in terms of scene visibility in the presence of horizontally slanted surfaces. To illustrate these ideas, we present a simple scanline algorithm by extending the approach of Section 2. In Section 4, we discuss the fundamental differences between vertical and horizontal slant, and show how vertical slant imposes additional restrictions on the manner in which disparity smoothness constraints can be used. We show how non-horizontal intensity edges play an important role in disparity estimation in the presence of vertical slant. In Section 5, we discuss the effects of higher order models of shape. Section 6 concludes by presenting experimental results and comparisons.

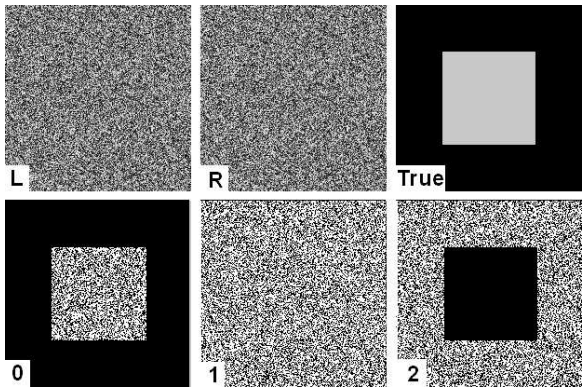

Figure 1. Top row: Left image, Right image, True disparity. Bottom row: Absolute intensity difference of left and right images for horizontal shift $\delta_x = 0, 1, 2$

2. Correspondence in Flatland

2.1. Chicken-and egg problems

Establishing correspondence between two images of a scene involves first selecting a local metric, such as the intensity (gray level) or color of a pixel which forms the basis for local comparisons. However, matching on the basis of such local information alone is almost impossible since many pixels have similar intensity or color. To reduce the correspondence possibilities for a pixel to a single possibility, regions around that pixel must be used along with additional continuity or smoothness assumptions about the scene depth. Thus, information around a pixel must be *aggregated* to obtain a unique match. Enforcing smoothness without a prior knowledge of depth discontinuities (segmentation) will inevitably lead to errors, especially near the discontinuities. Hence, prior knowledge of the segmentation is essential in order to correctly define regions around

a pixel for information aggregation. Conversely, if exact correspondence is known, the segmentation may be easily deduced.

Thus, if we knew the segmentation, then we could better estimate the correspondence. But we need correspondence in order to achieve segmentation. As we shall show in Section 3, scene shape critically affects correspondence, but then again, we need the correspondence in order to find the shape. Correspondence, segmentation and shape are chicken-and-egg problems: we need one in order to solve the other. Any recipe for solving such cyclic problems must involve feedback, either implicitly or explicitly. In the following sections, we examine these loops in an incremental manner. First, we study the relationship between correspondence and segmentation by working in a shapeless world called Flatland - a world which contains only fronto-parallel surfaces. Then, in Section 3, we introduce shape into the picture, and proceed to make explicit the relationship between shape and correspondence.

2.2. Connected matching regions

Let $I_1(x, y)$ and $I_2(x, y)$ be a given pair of rectified stereo images. The absolute intensity difference image is found using equation 1, where δ_x denotes the relative horizontal shift between the two input images.

$$\Delta I(x, y, \delta_x) = |I_1(x, y) - I_2(x + \delta_x, y)| \qquad (1)$$

The first row of Figure 1 shows a random dot pair of stereo images and the true disparity map. The second row shows the absolute intensity difference images for three horizontal shifts $\delta_x = 0, 1, 2$. If we observe the intensity difference images (second row), we notice that large connected regions of matching pixels (shown in black) appear for certain values of the shift. By the word 'match', we mean that the absolute intensity difference is below a certain threshold t, i.e. $\Delta I(x, y, \delta_x) < t$.

The appearance of large connected sets of matching pixels is the first observation of interest.

In the case of the random-dot pair, the background matches for $\delta_x = 0$ forming a large connected region, and the central square matches for $\delta_x = 2$. We know from the true disparity map that these shifts correspond to the correct disparities of the background and the square. However, we also notice that some pixels in the central square will match and form smaller connected regions even when the shift is wrong (not equal to 2). The same is true for the background pixels. *Thus, a pixel may form a part of a connected matching region even when the shift does not correspond to the true shift.* So how do we choose the correct shift for a pixel?

2.3. Boundaries and connectivity maximization

Recall our definition of Flatland - a world containing only fronto-parallel surfaces. Consider a uniformly colored region R_1 in image I_1 having a disparity δ_x. It corresponds to a region R_2 in the image I_2. Thus, if we shift image I_1 by δ_x and overlay it on I_2, then regions R_1 and R_2 will overlap and match perfectly and yield a connected region having an area equal to the size of R_1(and R_2). However, if the shift is not δ_x but has some other value, parts of R_1 and R_2 may still overlap and yield some connected matching region.

The area of overlap will be maximum only when the boundaries of R_1 and R_2 match perfectly, which occurs only for the true shift δ_x.

For all other shifts, the connected matching area will be less.

Similarly, in case the region R_1 (and hence R_2) is textured, connected matching regions will also be obtained for shifts other than the true shift δ_x. For example, if the regions contain a periodic texture such as a checkerboard, with square size λ, then we will obtain connected matching regions even if the shifts are $\delta_x + 2m\lambda$, where m is an integer. Even if this happens, the largest connected region will occur only when the boundaries of R_1 and R_2 match, which happens only if the shift equals the true shift δ_x. It is clear that only the knowledge of region boundaries allows us to assign correct shifts to the interior pixels. Maximizing the area of connected matching regions around a pixel is intimately related to the matching of region boundaries. As discussed later in Section 5, maximizing the size of matching regions is equivalent to minimizing the disparity segmentation. This *principle of minimum segmentation* is a more general constraint than connectivity maximization.

Using the above arguments, it is straightforward to show that in Flatland: *for any image pixel (x, y), the correct shift δ_x maximizes the area $A(x, y, \delta_x)$ of the connected matching region containing that pixel, and vice-versa.*

2.4. Vertical connectivity and horizontal edges

In Figure 2, we see a stereo pair of images having a gray background which has zero disparity, and a white square in front which has a non-zero disparity. On the right of the figure, we show the absolute intensity difference between the two images for zero relative shift. The black portion of this image indicates regions whose intensities cancel out (i.e., match) perfectly for zero relative shift. Notice that a part of the white foreground object also matches for zero shift, and is connected to the large matching background region. This will cause the entire black portion visible in the right hand side image to be labeled with zero disparity, which is clearly an incorrect result. The problem lies in the propagation of connectivity across horizontal edges. In a stereo

pair, if two single colored regions with different disparity are separated by a horizontal edge, then as we try out different horizontal shifts, points above and below the horizontal edge will always match regardless of the shift being tried. Thus, when we build connected components, regions on the two sides of the horizontal edge will form part of the same connected component, which causes both sides to be eventually assigned the same disparity. Hence, before we build connected components on our thresholded intensity difference images, we must explicitly sever connections across horizontal edges.

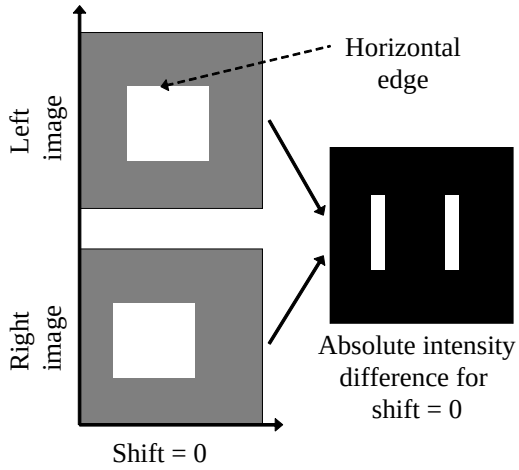

Figure 2. Vertical connectivity must not be established across horizontal edges

2.5. Occlusions and uniqueness

The *uniqueness constraint* states that a pixel in one image may not match more than one pixel in the other image. This basically means that if a region P_L in image I_L corresponds to a region P_R in image I_R, then there exists a one-to-one correspondence between pixels in P_L and pixels in P_R. Hence, except for the occluded pixels, every pixel in one image is paired with exactly one pixel in the other image. Thus, there is a competition between pairs (p_L, p_R) of pixels, where p_L is a pixel in the left image, and p_R is a pixel in the right image. The competition is based on the principle of maximum connectivity, outlined in the earlier section. If a pair (p_L, p_R) wins, it automatically precludes the existence of all pairs of the form $(p_L, p_{R'})$ and $(p_{L'}, p_R)$, such that $p_{L'} \neq p_L$ and $p_{R'} \neq p_R$. Some pixels, which do not form a part of any of the winning pairs, are the occlusions. It is possible to enforce the uniqueness constraint within the correspondence search itself as it progresses: whenever we assign a new partner to a given pixel, we make sure that its previous partner (if it was previously paired) is marked as unpaired.

2.6. An algorithm for Flatland

Our algorithm for Flatland is outlined in Figure 3. Basically, the algorithm consists of the following steps:

1. For every shift $\delta_x \in \{\delta_1, \delta_2, ..., \delta_k\}$, do
   (a) Shift the left image I_L horizontally by δ_x to get I'_L, and match I'_L with I_R
   (b) If a horizontal edge separates a pixel (x, y) and its vertical neighbor $(x, y-1)$, their connection must be severed.
   (c) Build connected components taking into account the vertical connections established in the previous step.
   (d) Find the sizes or weights of the connected components.
   (e) For each pixel, if the connected component containing it is larger than the previous shifts, update left and right disparity maps, while enforcing uniqueness.

Thus, the process consists of matching pixels (using thresholded absolute intensity differences) for various shifts (disparity/flow candidates), finding connected components and maximizing a measure of the connectivity for each pixel. In [25], Boykov et al. present a method which uses the same principle of maximizing connected components, but the vertical connectivity constraints are not imposed.

Measures of connectivity other than the area may also be used; for example, we may use a combination of the area of the connected component and the total intensity difference inside the connected component. Also, pixels which locally match for k shifts out of d possible shifts can be assigned to have an area of $1/k$; this ensures that pixels which match frequently do not dominate the estimation.

For d possible shifts and an image with N pixels, the total running time is $\Theta(Nd)$. Processing times on real images are given in Section 6. In our implementation, we use the technique of Birchfield and Tomasi [26] to calculate the absolute intensity differences. For color images, matching two pixels implies matching all their color channels.

Real world scenes are quite unlike Flatland, since they rarely consist of fronto-parallel surfaces. In the following sections, we shall identify new issues which arise in the presence of slanted surfaces, which require us to alter our definitions of correspondence and introduce novel constraints for detecting occlusions. We shall also see that the problems with vertical connectivity are aggravated in the presence of slanted surfaces.

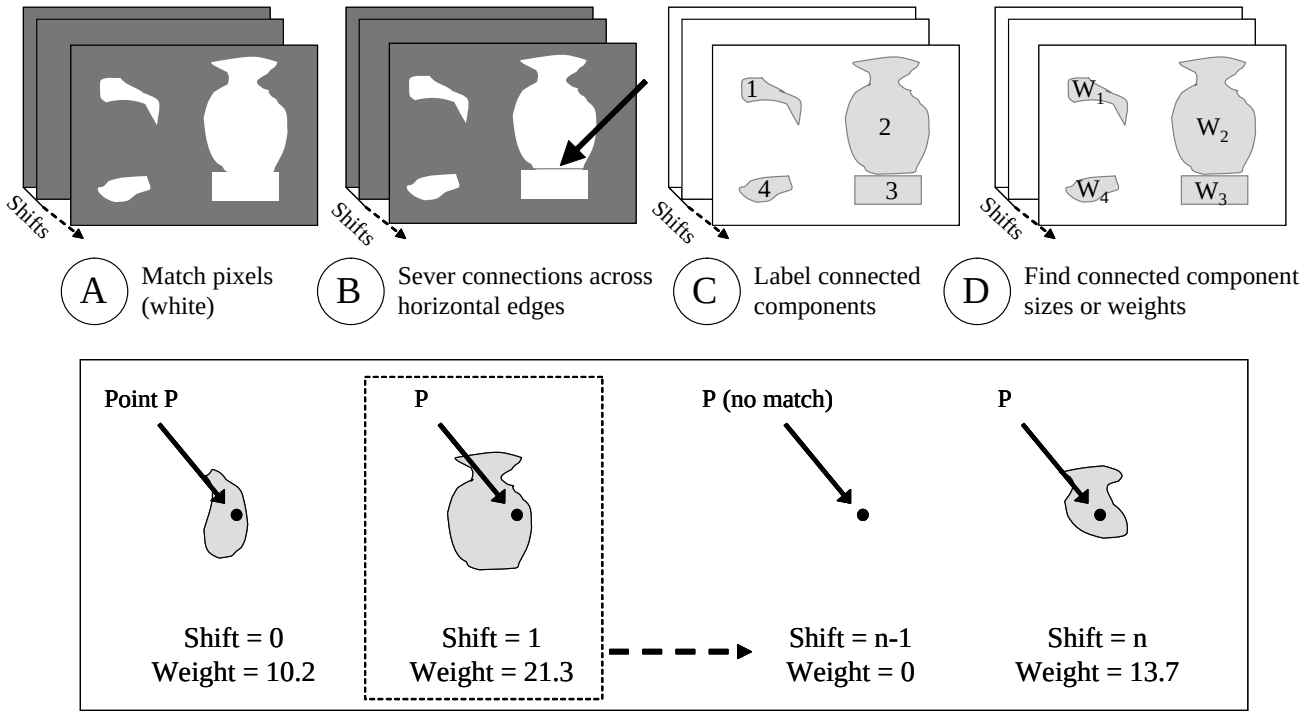

Figure 3. An algorithm for Flatland

3. Horizontal slant

Let us leave Flatland by introducing horizontally slanted surfaces into the scene, i.e., the disparity on such surfaces changes as we move along the X-axis (horizontally), and does not change if we move along the Y-axis. Let us also assume for simplicity that we are using a stereo system with a parallel viewing geometry and in which the cameras are separated only by a translation along the X-axis. Our system therefore provides us with rectified images.

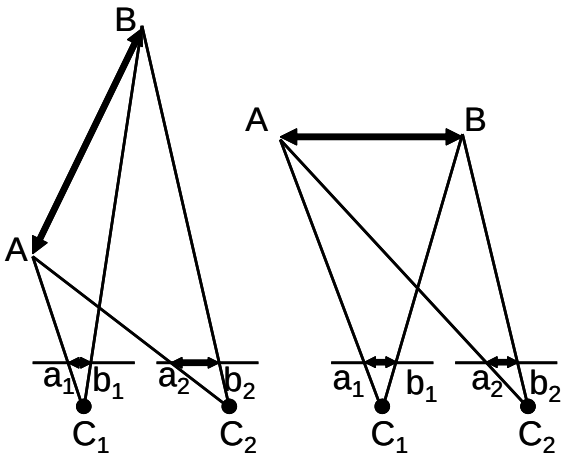

Figure 4. (a) unequal projection lengths of a horizontally slanted line (b) equal projection lengths of a fronto-parallel line

3.1. Unequal projection lengths and interval matching

Figure 4(a) shows that a horizontally slanted line AB in the scene projects onto the line segment a_1b_1 in camera C_1, and a_2b_2 in camera C_2. Clearly, the lengths of a_1b_1 and a_2b_2 are not equal. Assume that the cameras have focal length equal to 1. Let the point A have coordinates (X_A, Z_A) in space with respect to camera 1, and point B have coordinates (X_B, Z_B), where the X-axis is along the scanline, and the Z-axis is normal to the scanline. Then, if the cameras are separated by a translation t, we can immediately find the lengths L_1 and L_2 of the projected line segments in the two cameras.

$$\begin{aligned} L_1 &= X_B/Z_B - X_A/Z_A \\ L_2 &= (X_B - t)/Z_B - (X_A - t)/Z_A \end{aligned} \tag{2}$$

Clearly, in general, $L_1$and L_2 are not equal. For the fronto-parallel line shown in Figure 4(b), $Z_A = Z_B = Z$, hence

$$L_1 = L_2 = (X_B - X_A)/Z \tag{3}$$

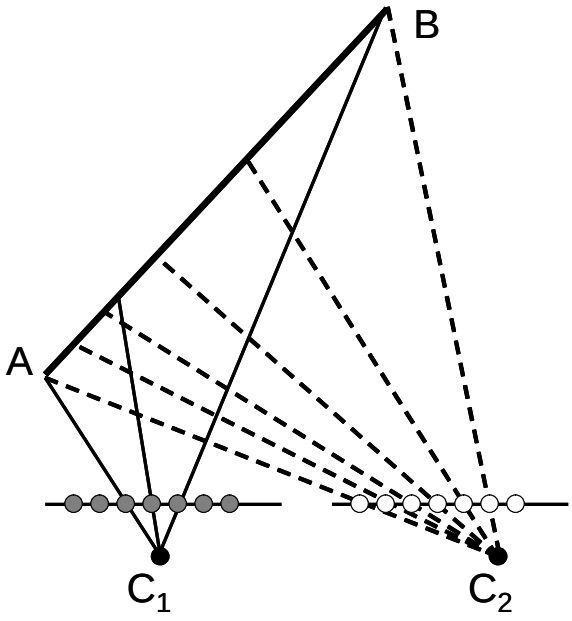

Figure 5. Sampling problem for a horizontally slanted line

Thus, we have the following:

- *Except for the fronto-parallel case, horizontally slanted line segments in space will always project onto segments of different lengths in the two cameras.*
- *Consequently, N pixels on a scanline in one image can correspond to a different number of pixels M on a scanline in the other image.*

We must ensure that our stereo algorithms permit unequal correspondences of this nature; hence, an interval on a scanline in one image must be matched to an interval on a scanline in the other image, where the two intervals being matched may have different lengths. Note that the scanline is treated as a continuous entity rather than a discrete pixelized entity.

Conclusion: We must perform interval matching instead of pixel matching.

3.2. Slant affects Sampling

Since a horizontally slanted line segment in space has different projection lengths in the two cameras, its intensity function is also sampled differently by the two cameras as shown in Figure 5. Birchfield and Tomasi [26] have provided a very useful method for matching pixel intensities, which is used by many of the best performing stereo algorithms. Let us briefly describe what this procedure does:

Given two scanlines $I_L(x)$ and $I_R(x)$, we have to find the absolute intensity difference between pixel x_L in the left scanline and pixel x_R in the right scanline. We first find $I_L(x_L - 1/2)$, $I_L(x_L + 1/2)$, $I_R(x_R - 1/2)$ and $I_R(x_R + 1/2)$ by a simple linear interpolation. These values are used to find $I_L^{min} = min\{I_L(x_L - 1/2), I_L(x_L), I_L(x_L + 1/2)\}$, $I_L^{max} = max\{I_L(x_L - 1/2), I_L(x_L), I_L(x_L + 1/2)\}$, and similarly I_R^{min} and I_R^{max}. The left difference is $d_L = max\{0, I_L(x_L) - I_R^{max}, I_R^{min} - I_L(x_L)\}$ and the right difference is $d_R = max\{0, I_R(x_R) - I_L^{max}, I_L^{min} - I_R(x_R)\}$. Finally, the absolute intensity difference between the pixels is $d = min\{d_L, d_R\}$. The procedure is therefore symmetric and linearly interpolates the intensity function between neighboring pixels. Such a matching procedure cannot be applied directly in the presence of horizontal slant, due to the unequal sampling.

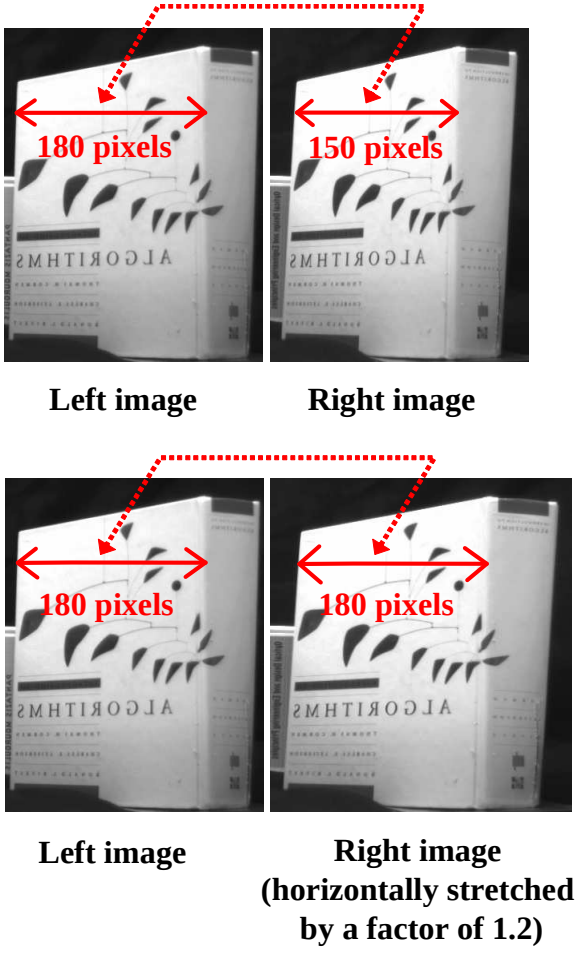

Figure 6. Stretch and match

We must first resample each scanline correctly, and then apply the Birchfield-Tomasi matching method. In other words, we first stretch one of the scanlines, by an amount related to the horizontal slant we are considering, and then match this stretched scanline with the other unstretched scanline using the Birchfield-Tomasi matching method as usual. For example, if we are considering the linear correspondence function $x_R = mx_L + d$ between points of camera L and R, then we must stretch the image of camera L by a factor m before performing the intensity based matching. Thus, we first compute I_L^{min} and I_L^{max} for the unstretched scanline I_L, then stretch all three by a factor m, and then apply the remainder of the Birchfield-Tomasi method. As we try various values of the slant, we appropriately resample the scanlines before matching. Figure 6 shows how corresponding line segments of unequal length attain the same length after stretching one of the images.

Conclusion: Stretch one of the images first and then match .

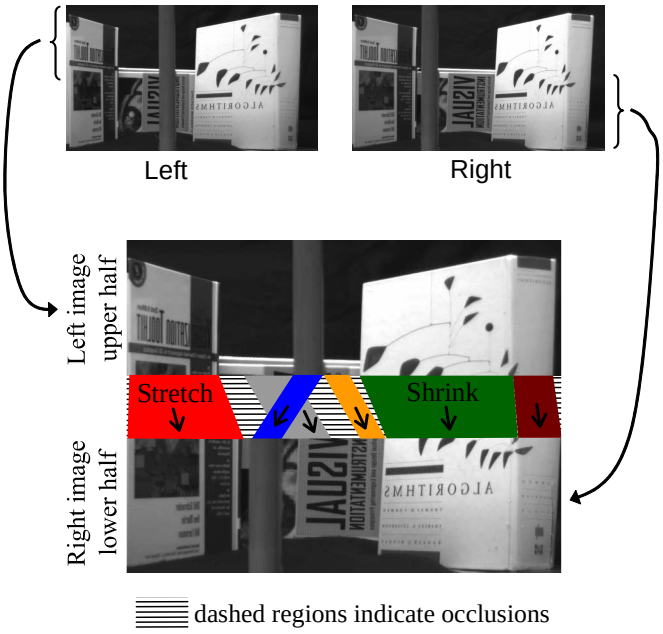

Figure 7. Top: stereo pair of images. Bottom: Corresponding intervals on the left and right scanlines can have different length. The order (left-right) of matching intervals can also change (see the blue and gray intervals).

3.3. Occlusions and the new interval uniqueness constraint

The uniqueness constraint [18] is often used to find occlusions. In its present form, it states that a pixel in one image may not match more than one pixel in the other image. This basically means that if a region P_L in image I_L corresponds to a region P_R in image I_R, then there exists a one-to-one correspondence between pixels in P_L and pixels

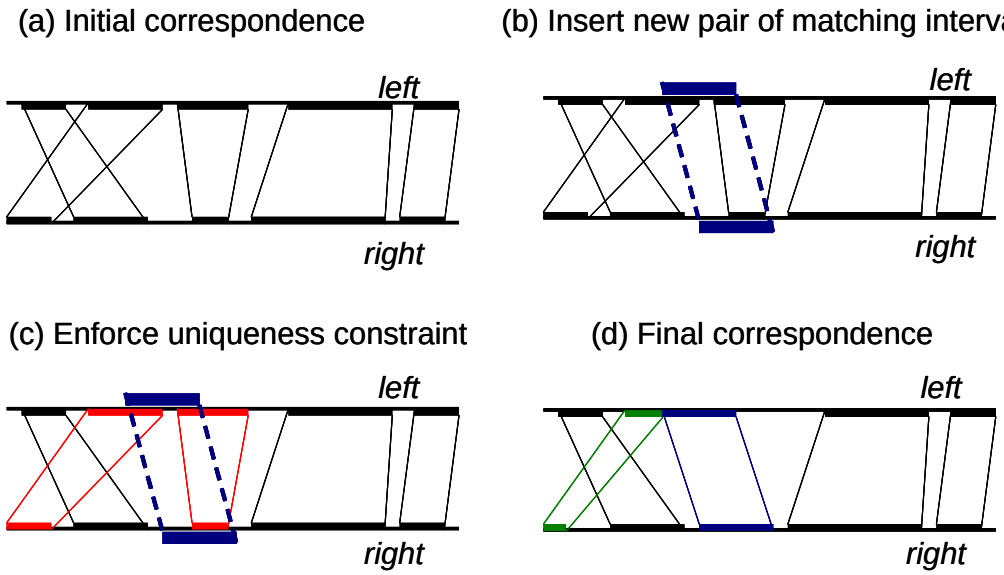

Figure 8. The modified uniqueness constraint operates by preserving a one-to-one correspondence between intervals on the left and right scanlines, instead of pixels.

in P_R. Hence, except for the occluded pixels, every pixel in one image is paired with exactly one pixel in the other image; the unpaired pixels are the occlusions. However, since horizontal slant allows N pixels in one image to match with a different number of pixels M in the other image, we can no longer impose a one-to-one correspondence for finding occlusions. We propose a new uniqueness constraint which enforces *a one-to-one mapping between continuous intervals* (line segments) in the two scanlines, instead of pixels. An interval in one scanline may correspond to an interval of a different length in the other scanline, as long as the correspondence is unique. This is equivalent to enforcing uniqueness in the scene space instead of the image space, hence we may also refer to this constraint as the 3D uniqueness constraint. Figure 7 illustrates the idea of interval mapping and occlusions detection using a real example. In this figure, we see how intervals of different length can correspond to each other, leaving behind the occlusions.

Figure 8 shows how the new uniqueness constraint can be implemented. Part (a) shows an existing one-to-one correspondence between intervals on the left and right scanlines. This denotes an intermediate state in the progress of a stereo matching and segmentation algorithm. Notice that the intervals may correspond in any order (i.e., the ordering constraint is not needed). Now, in part (b), we wish to insert a new pair of corresponding intervals, shown in blue. (This new pair of matching intervals improves upon the existing matches according to some energy metric which depends on the stereo algorithm being used). In part (c), we see that the insertion of this pair of intervals conflicts with existing intervals (shown in red). In order to enforce uniqueness, the red pair of intervals on the right must be removed, while the red pair of intervals on the left must be resized. In part (d), we see the new correspondences. The interval pair which was resized is shown in green, and the newly inserted pair is shown in blue.

Conclusion: There exists a one-to-one mapping between intervals (possibly having unequal lengths), and not between pixels.

3.4. An algorithm to deal with horizontal slant

We now describe a simple scanline algorithm which implements the ideas presented above; this algorithm also uses the concept of connectivity maximization presented in Section 2 along scanlines instead of the whole image, and simultaneously searches the space of possible disparities and horizontal slants. It processes a pair of scanlines $I_L(x)$ and $I_R(x)$ at a time without using any vertical consistency constraints. Horizontal disparities $\Delta_L(x)$ are assigned to the left scanline within a given range $[\Delta_1, \Delta_2]$, and $\Delta_R(x)$ to the right scanline in the range $[-\Delta_2, -\Delta_1]$. The disparities are not assigned to pixels, but continuously over the whole scanline. The disparities are not directly estimated, but instead, we search for functions $m_L(x)$ and $d_L(x)$ for the left scanline, and $m_R(x)$ and $d_R(x)$ for the right scanline, such that given a point x_L on the left scanline, its corresponding

point x_R in the right scanline would be

$$x_R = m_L(x_L) \cdot x_L + d_L(x_L) \tag{4}$$

and reciprocally:

$$x_L = m_R(x_R) \cdot x_R + d_R(x_R)$$

Clearly,

$$m_R(x_R) = 1/m_L(x_L)$$
$$d_R(x_R) = -d_L(x_L)/m_L(x_L)$$

The disparities are then computed as:

$$\Delta_L(x_L) = x_R - x_L = (m_L(x_L) - 1) \cdot x_L + d_L(x_L)$$
$$\Delta_R(x_R) = x_L - x_R = (m_R(x_R) - 1) \cdot x_R + d_R(x_R) \tag{5}$$

The functions m_L and m_R are the horizontal slants, which allow line segments of different length in the two scanlines to correspond. The scanlines are represented continuously by linearly interpolating intensities between pixel locations. Thus, if $m_L = 2$, then the left scanline is stretched (resampled) by a factor of 2, and then matched with the unstretched right scanline using the Birchfield-Tomasi method. Due to the stretching of one scanline before performing the intensity based matching, we are automatically modifying the traditional Birchfield-Tomasi method to properly deal with horizontal slant. For each possible m_L and d_L, absolute intensity differences between corresponding points are computed, and thresholded by a threshold t; in Section 6, we briefly discuss extensions which eliminate the need for a threshold. Then, the best value of m_L and d_L for a point is chosen such that it maximizes the size of the matching line segment containing that point (i.e., the maximum connectivity approach of Section 2).

The values of the horizontal slant which are to be examined are provided as inputs, i.e., $m_L, m_R \in M$, where $M = \{m_1, m_2..., m_k\}$, such that $m_1, m_2, ..., m_k \geq 1$. The disparity search range $[\Delta_1, \Delta_2]$ is also provided as an input. In order to find the occlusions, we enforce the uniqueness constraint in its modified form as shown in Figure 8. We maintain a one-to-one correspondence between intervals in the two scanlines. Hence, at any stage of the process, we have a set S_L of non-overlapping intervals in the left scanline and a set S_R of non-overlapping intervals in the right scanline. An interval i is of the form $[x_1, x_2)$. The uniqueness constraint enforces a one-to-one mapping U between the elements of S_L and the elements of S_R. When a new corresponding pair of intervals i_L and i_R is found, the segment previously corresponding to i_L is removed if present, and the same is done for i_R. Then, i_L is added to S_L, and i_R to S_R, and the one-to-one mapping in U is updated. Thus, we always ensure that a line segment in the left scanline uniquely maps to a line segment in the right scanline. In the end, line segments which remain unmapped are the occlusions. In our implementation, we have used hash-tables to maintain the interval information and detect overlaps. The skeleton of the algorithm is shown below.

1. For all $m_L \in M$, $\Delta_L \in [\Delta_1, \Delta_2]$, do
   (a) stretch I_L by m_L to get I'_L
   (b) find range for d_L using given range for Δ_L and eqn. 5
   (c) for every d_L, match I'_L and I_R and find connected matching segments and their sizes; update correspondence map while enforcing the uniqueness constraint.
2. For all $m_R \in M$, $\Delta_R \in [-\Delta_2, -\Delta_1]$, do
   (a) stretch I_R by m_R to get I'_R
   (b) find range for d_R using given range for Δ_R and eqn. 5
   (c) for every d_R, match I'_R and I_L and find connected matching segments and their sizes; update correspondence map while enforcing the uniqueness constraint
3. $m_L = m_R = 1$
   (a) for every $d_L \in [\Delta_1, \Delta_2]$, match I_R and I_L and find connected matching segments and their sizes; update correspondence map while enforcing the uniqueness constraint

4. Vertical slant

4.1. Fundamental differences between vertical and horizontal slant

Assume that we are given a rectified stereo pair of images. Due to the discrete (pixelized) nature of the images, changes in disparity as we move from left to right along a scanline may be caused by one of two factors:

- There exists a depth discontinuity (as seen in Figure 9(a)), or
- The pixels form a part of a horizontally slanted surface (Figure 9(b)).

In this case, we can distinguish between these two possibilities because only a true depth discontinuity will cause an occlusion to appear (as seen in Figure 9(a)). However, if we

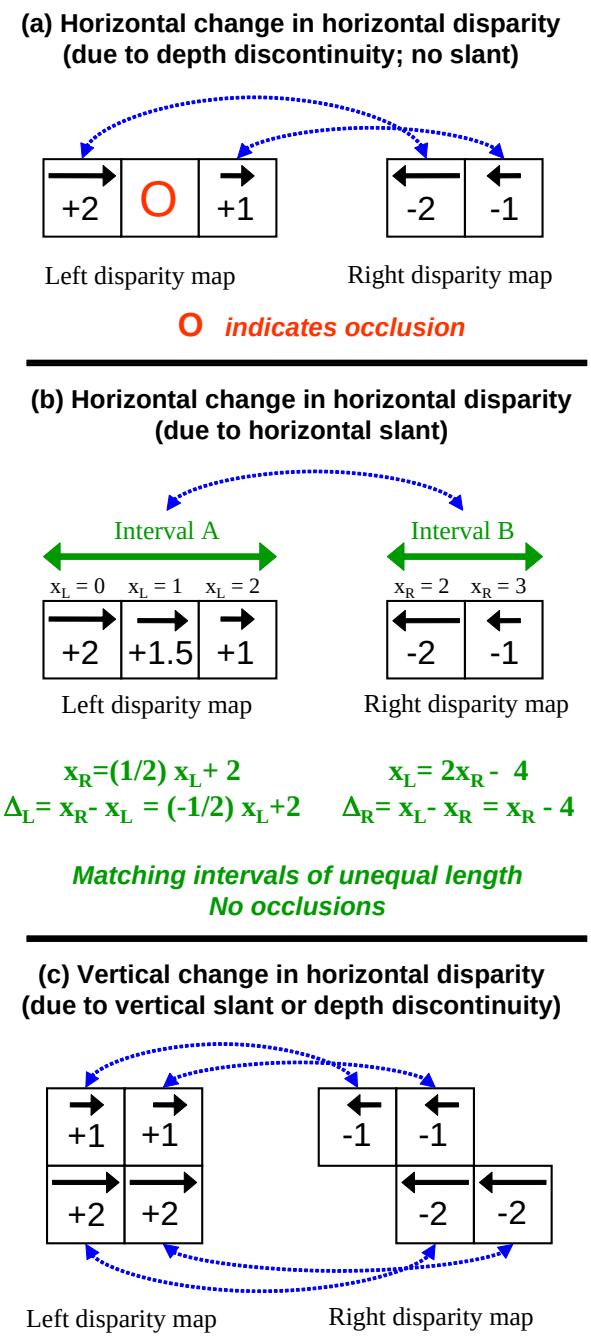

Figure 9. Top: Horizontal changes in horizontal disparity due to a discontinuity create an occlusion. Middle: Horizontal changes in horizontal disparity due to horizontal slant lead to stretching/shrinking but no occlusions. Bottom: Vertical changes in horizontal disparity due to discontinuity and vertical slant cannot be distinguished (since no occlusions occur in either case).

move vertically and find a disparity change (as seen in Figure 9(c)), we have no way of distinguishing whether the vertical change is caused by a discontinuity or by a vertically slanted surface, since neither causes occlusions to appear. Thus, there is a fundamental difference between horizontal and vertical slant.

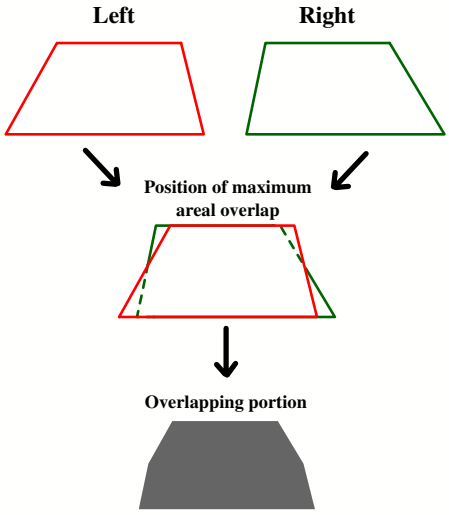

Figure 10. Top: images of a vertically slanted plane. Middle: images overlaid to maximize overlap. Bottom: area of largest overlap

4.2. Vertical connectivity and non-horizontal edges

Since we cannot distinguish whether purely vertical changes in disparity are due to a true discontinuity or due to a vertically slanted surface, additional assumptions must be employed if we are to enforce any vertical consistency constraints on the disparity map. Consider this example: in the image, if we have a horizontal intensity edge (a horizontal line), we have two possibilities (a) this edge corresponds to a depth discontinuity, and pixels separated by it do not lie on the same surface, or (b) the edge is just an intensity edge, and pixels separated by it lie on the same surface. If we commit the error of assuming that the pixels separated by the horizontal edge are connected and it happens to be a discontinuity, our solution will yield a disparity map without the depth discontinuity, which is clearly incorrect. It is safer to assume that such pixels are *not* connected, to allow the possibility that there may be a depth discontinuity.

Therefore, vertical neighbors separated by a horizontal edge or no edge at all should not be connected.

Also, as shown in Figure 10, if we have two images of a single-colored object, and we assume vertical connections in the interior, then we will get a single disparity in the en-

tire interior (when maximum overlap of the images takes place) instead of a vertical gradient. Thus, we cannot assume that disparity is vertically constant even if two vertical neighbors have the same color/intensity. *Disparity can change even when there is no change in color or intensity.* (Note that we *can* assume that disparity is *continuous*, but not necessarily *constant*, if intensity or color do not vary in a region.)

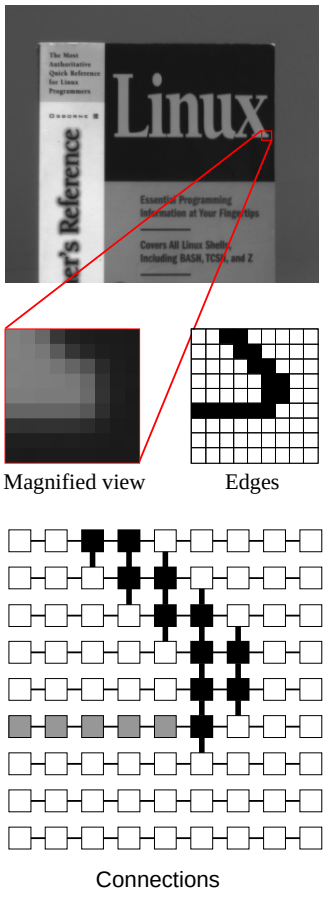

Figure 11. Vertical connections between pixels are established only along non-horizontal edges

However, if we have a non-horizontal edge running across the image, it will cause occlusions to appear if it is a discontinuity, and no occlusions will appear if both sides of it lie on the same surface. This distinguishing ability allows us to make the assumption that:

Vertical neighbors lying on non-horizontal edges should be connected (Figure 11).

This can be implemented by introducing a simple modification to the scanline algorithm presented earlier. If we assume that each pixel is connected by links to the pixel directly above it and the pixel directly below it, then the only links which are left intact are the ones lying on non-horizontal edges. Then, the connected component labeling can be performed as before, and if two vertical neighbors are linked, then they belong to the same connected component. Edge detection is done using a standard Canny edge detector, and edge directions are found by computing the gradient direction.

4.3. Cue integration along the vertical direction

We have examined the differences in the character of vertical and horizontal slant in the previous sections. It is clear that if there are vertical changes in the horizontal disparity, we cannot distinguish whether we have a discontinuity or merely a vertically slanted surface. In this situation, it is conceivable that inputs from 'Shape from X' modules (e.g., shape from texture, shape from shading) are critical to establish vertical consistency and construct vertically smooth models of the scene shape and structure.

We believe that such external cues may strongly influence the estimation of vertical slant in the human visual system, although not so much the horizontal slant. There exists some support for this idea in studies dealing with the perception of slanted surfaces by humans, which conclude that there is an anisotropy in the perception of stereoscopic slant [27, 28, 29, 30, 31], i.e., a horizontally slanted surface and a vertically slanted surface having the same slant are perceived differently. For example, [28] states that vertical gradients of horizontal disparity are more easily perceived compared to the horizontal gradients, which means that horizontally slanted surfaces tend to appear more frontoparallel than vertically slanted surfaces. The authors explored the role of orientation disparity cues which may influence the case of vertically slanted surfaces, but concluded that even if the orientation disparity cues are equally strong for the horizontally and vertically slanted case, the anisotropy in slant perception persists. They conclude that there must exist other anisotropic processes which are involved in the computation of slant. Studies by Gilliam and Ryan [29, 31] also discuss the role of configural properties (such as surface contours) in determining the slant for a given disparity gradient. They also conclude that there exist configural effects other than orientation disparity and perspective which also contribute to the anisotropy in slant perception. If shape from disparity is being integrated with other cues such as shape from texture differently in the vertical direction than the horizontal, such an anisotropy in slant perception could arise; this has recently been explored in [32], and we feel that our arguments regarding vertical and horizontal slant provide a reason for such anisotropic cue combination to occur.

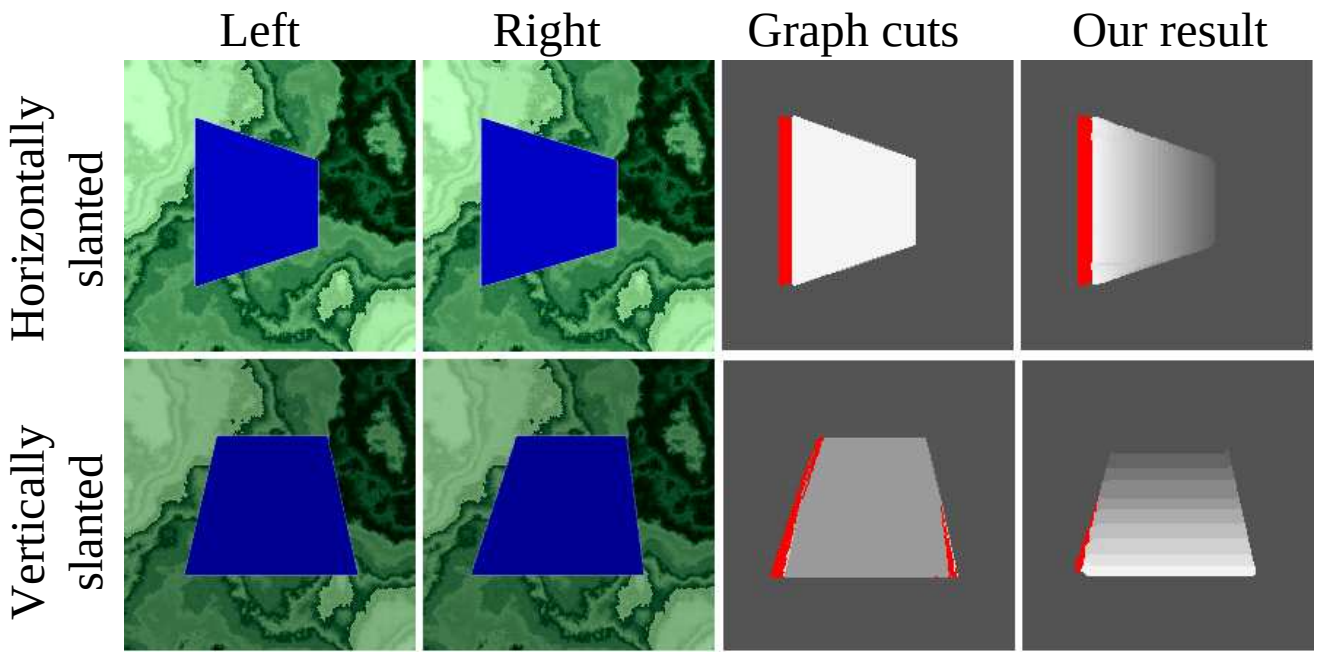

Figure 12. Columns 1 to 4: Left image, right image, graph cuts result for the left disparity map, our result for left disparity map. Row 1: horizontally slanted object, Row 2: vertically slanted object. Occlusions are shown in red.

5. Higher order models of shape

When we compute the disparity map of a real scene, the desirable property of such a disparity map is that it should explain the observed images while minimizing the number of discontinuities. In other words, we would like to model the disparity map as a *piecewise continuous* function which is consistent with the observed images and has the minimum possible number of pieces. This *principle of minimum segmentation* implies that we desire each segment to have the maximum possible size, which is consistent with the ideas on connectivity maximization which we have introduced in Section 2.

To simplify the problem computationally, we often choose more restrictive versions of the general model of a piecewise continuous disparity map. The simplest but most restrictive version models the disparity map as a *piecewise constant* function, which we discussed in Section 2. An obvious improvement is to model the disparity map as a *piecewise linear* function (i.e., the depth is piecewise hyperbolic), which we discuss in Sections 3 and 4. This method works well for a world consisting of planar surfaces, but we need even better models to properly deal with curved surfaces. The method we have discussed can be easily extended to include more complex models of the disparity and shape, such as quadratic and cubic models, in an attempt to get closer to the true property of piecewise continuity. However, these models increase the dimensionality of the search space. We are also developing methods which allow us to progressively increase the complexity of the models by using the results of a simpler model (such as piecewise planar) to restrict the search space for a more complex model (such as piecewise quadratic). This approach will allow us to use quadratic or cubic models for the disparity while preserving the low dimensionality of the search space.

6. Experiments

We have shown in the previous sections that horizontal and vertical slant play a critical role in the estimation of correspondence and occlusions. The first row of Figure 12 shows a stereo pair of images in which the blue object is horizontally slanted (i.e., depth varies from left to right), and the second row shows a stereo pair in which the blue object is vertically slanted. The third column of this figure shows the results of the graph cuts [17] algorithm, while the fourth column shows our results for each of these stereo pairs. In these results, occlusions are shown in red. The graph cuts result was obtained using software kindly provided by the authors (*www.cs.cornell.edu/People/vnk/software.html*). It is clear that for both these stereo pairs, the graph cuts result gives a

constant disparity for the blue object, while our result correctly shows the slant and still finds the occlusions.

We expect that the constraints presented above will improve the results of many existing dense stereo algorithms in both qualitative and quantitative ways. However, for the sake of completeness, we compare our results with other algorithms using the test procedure created by Scharstein and Szeliski [1] available at *www.middlebury.edu/stereo*. They compare the disparity map d_{out} generated by an algorithm to the true disparity d_{true}, and the pixels which deviate by more than 1 unit from their true disparity are labeled as 'bad' pixels. The percentage of bad pixels in the entire image, in the untextured regions and near depth discontinuities are used to compare the results of various algorithms. The percentages of bad pixels are reported in Table 1, which was generated by submitting our disparity maps (Figure 14) to the web-based evaluation program created by Scharstein and Szeliski. Our algorithm ('connectivity-slant') ranks sixth overall, while the ranks in each column are showed in brackets, below the error percentages. For the bottom left of the Venus sequence, it is not possible to assign correct disparities, since the corresponding points in the second image lie outside the image. Scharstein and Szeliski exclude a ten pixel boundary before evaluation, but it is not adequate to remedy this situation (a twenty pixel left boundary will suffice). The execution time of the algorithm on these image pairs is in the range of 1-5 seconds on a Pentium 2.4 GHz machine (some of our matching code is available at *www.cs.umd.edu/users/ogale*).

The results shown here use the vertical connectivity constraints discussed in Section 4. Compared to the scanline algorithm, the addition of these constraints improves the results marginally and reduces some of the streaking. To see the effect of these vertical consistency constraints, the reader can compare the results given here to the results of the scanline algorithm which appear in [24].

Figure 15 shows the results for two more stereo pairs: the tree branch pair and the corridor pair. The tree branch illustrates the ability of the algorithm to correctly handle thin overlapping objects. The corridor scene contains many untextured surfaces which are strongly slanted. Note the correctness of the results for the walls, and especially for the left wall, which has a very large slant.

It is important to mention at this point that our current algorithm presents us with the problem of choosing a hard threshold t to decide if two pixels match or not, and the choice of this threshold can affect the results. To address this problem, we have proposed a new method in [33] which generalizes the connected component process discussed in this paper to a single-pass diffusion process. This modified representation does not require a threshold to be set, and can even correspond images with very different contrast and additive noise; an example result is shown in Figure 16.

7. Conclusion

We have analyzed the effects of shape in establishing dense point correspondence between a stereo pair of images. Ideally, it is desirable to model the disparity map as a piecewise-continuous function having the minimum number of pieces, which upto zeroth order, can be approximated by a piecewise-constant function. The idea of connectivity maximization was proposed for this flat fronto-parallel world, which is equivalent to finding the minimum segmentation. Proceeding to a first order approximation, we then examined the effects of horizontal slant: unequal projection lengths, sampling issues, and invalidation of the uniqueness constraint for finding occlusions. It was shown that interval matching and a new uniqueness constraint are required to handle horizontal slant. We then highlighted qualitative differences between horizontal and vertical slant, and discussed the importance of non-horizontal edges for the latter. Experimental results with quantitative and qualitative comparisons were provided. The ideas presented in this paper for the case of stereo correspondence are directly applicable to the case of dense optical flow and, in [34], these extensions have been discussed. In conclusion, as indicated in Figure 13, we have shown that correspondence, segmentation, occlusion detection, and shape estimation influence each other and have to be solved in concert, with other modalities such as 'Shape from X' possibly influencing the computation of shape in an anisotropic manner.

8. Acknowledgements

The support of the US National Science Foundation and ARDA is gratefully acknowledged.

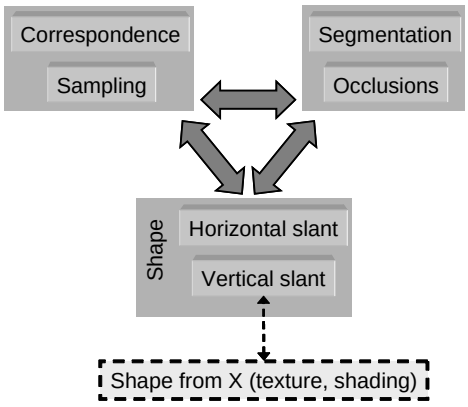

Figure 13. Visual problems such as correspondence, depth segmentation, and shape estimation must be solved simultaneously

Table 1. Performance comparison from the Middlebury Stereo Vision Page (overall rank is 6'th among 28 algorithms). The table shows only the top ten algorithms. Error percentages and rank (in brackets) in each column is shown.

Rank	Algorithm	Tsukuba			Sawtooth			Venus			Map	
		all	untex	disc.	**all**	untex	disc.	**all**	untex	disc.	**all**	disc.
1	Layered	1.58 (4)	1.06 (7)	8.82 (5)	0.34 (1)	0.00 (1)	3.35 (1)	1.52 (8)	2.96 (17)	2.62 (2)	0.37 (11)	5.24 (11)
2	Belief prop	1.15 (1)	0.42 (2)	6.31 (1)	0.98 (8)	0.30 (13)	4.83 (5)	1.00 (4)	0.76 (4)	9.13 (12)	0.84 (18)	5.27 (12)
3	Multcam GC	1.85 (8)	1.94 (13)	6.99 (4)	0.62 (6)	0.00 (1)	6.86 (10)	1.21 (6)	1.96 (8)	5.71 (6)	0.31 (8)	4.34 (10)
4	GC+occl.	1.19 (2)	0.23 (1)	6.71 (2)	0.73 (7)	0.11 (7)	5.71 (8)	1.64 (11)	2.75 (15)	5.41 (5)	0.61 (14)	6.05 (13)
5	Impr.Coop.	1.67 (5)	0.77 (4)	9.67 (9)	1.21 (12)	0.17 (10)	6.90 (11)	1.04 (5)	1.07 (5)	13.68 (17)	0.29 (6)	3.65 (7)
→	***6***	**1.77**	**0.95**	**9.48**	**0.61**	**0.17**	**5.05**	**3.00**	**5.22**	**7.63**	**0.21**	**3.01**
	connectivity -slant	**(6)**	**(5)**	**(7)**	**(4)**	**(11)**	**(6)**	**(20)**	**(20)**	**(8)**	**(2)**	**(4)**
7	GC+occl.	1.27 (3)	0.43 (3)	6.90 (3)	0.36 (2)	0.00 (1)	3.65 (2)	2.79 (19)	5.39 (21)	2.54 (1)	1.79 (21)	10.08 (20)
8	Disc. pres.	1.78 (7)	1.22 (9)	9.71 (10)	1.17 (10)	0.08 (6)	5.55 (7)	1.61 (10)	2.25 (11)	9.06 (11)	0.32 (9)	3.33 (6)
9	Graph cuts	1.94 (10)	1.09 (8)	9.49 (8)	1.30 (14)	0.06 (5)	6.34 (9)	1.79 (14)	2.61 (14)	6.91 (7)	0.31 (7)	3.88 (8)
10	Symbiotic	2.87 (13)	1.71 (11)	11.90 (11)	1.04 (9)	0.13 (8)	7.32 (13)	0.51 (2)	0.23 (2)	7.88 (9)	0.50 (13)	6.54 (14)
	⇩				⇩						⇩	
28	Max. surf.	11.10 (28)	10.70 (26)	41.99 (28)	5.51 (28)	5.56 (28)	27.39 (27)	4.36 (23)	4.78 (19)	41.13 (27)	4.17 (27)	27.88 (27)

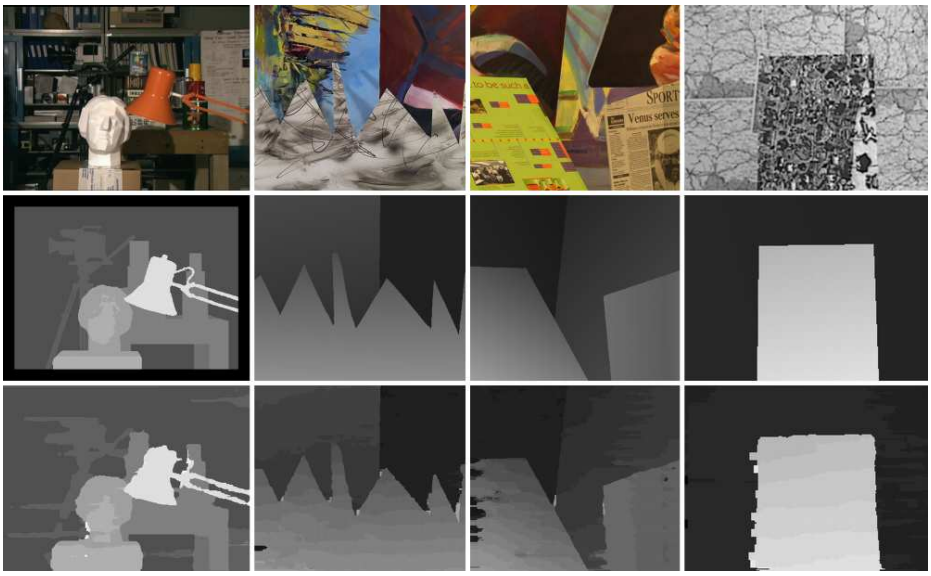

Figure 14. Top row (Left frames), Middle row (ground truth), Bottom row (our results). Occlusions were filled in as required by the evaluation procedure.

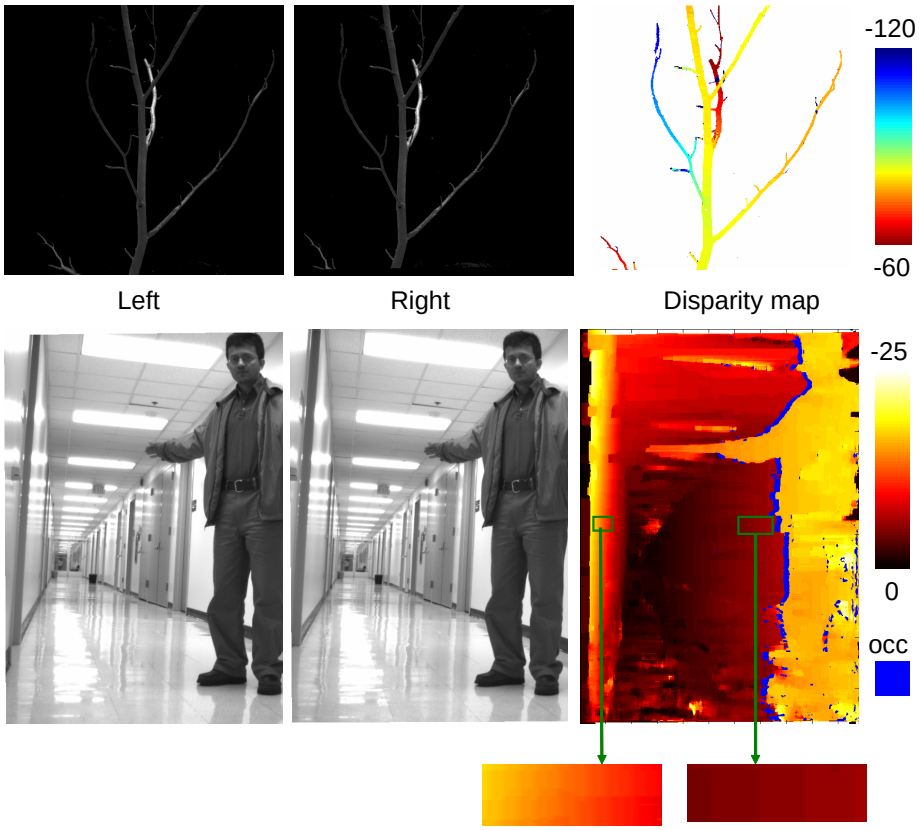

Figure 15. Top row: tree stereo pair and disparity map. Bottom row: Corridor stereo pair with the disparity map and occlusions (blue regions). Note the disparity variation for the left and right walls of the corridor.

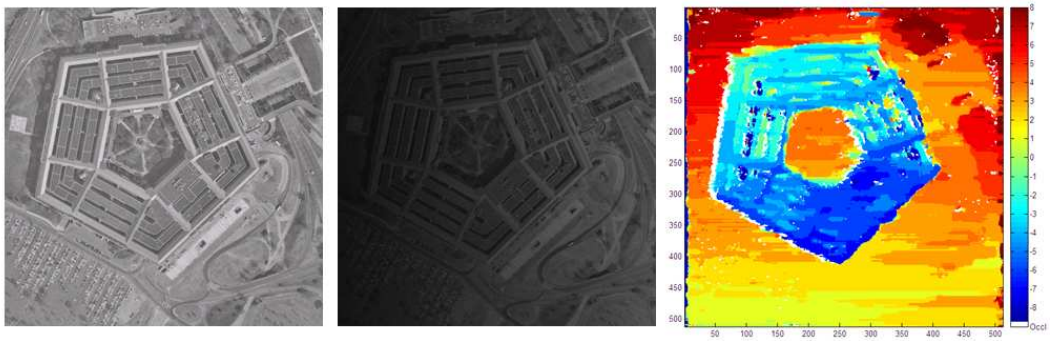

Figure 16. Contrast invariant stereo correspondence (see [33]): left image, right image, disparity map with occlusions (white). Note that the right image has a different contrast than the left, and that the contrast is spatially varying.

References

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int'l Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, April 2002.

[2] M. Okutomi and T. Kanade, "A multiple baseline stereo," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, April 1993.

[3] D. Geiger, B. Ladendorf, and A. Yuille, "Occlusions and binocular stereo," *Proc. European Conf. Computer Vision*, pp. 425–433, 1992.

[4] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 858–863, June 1997.

[5] 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, 1994.

[6] A. F. Bobick and S. S. Intille., "Large occlusion stereo," *Int'l Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, Sept 1999.

[7] H. Tao, H. Sawhney, and R. Kumar, "A global matching framework for stereo computation," *Proc. Int'l Conf. Computer Vision*, vol. 1, pp. 532–539, July 2001.

[8] M. Okutomi, Y. Katayama, and S. Oka, "A simple stereo algorithm to recover precise object boundaries and smooth surfaces," *Int'l Journal Computer Vision*, vol. 47, no. 1-3, pp. 261–273, 2002.

[9] J. Mulligan and K. Daniilidis, "Predicting disparity windows for real-time stereo," *Lecture Notes in Computer Science*, vol. 1842, pp. 220–235, 2000.

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

[11] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, Nov 1984.

[12] S. T. Barnard, "Stochastic stereo matching over scale," *Int'l Journal of Computer Vision*, vol. 3, no. 1, pp. 17–32, 1989.

[13] R. Szeliski, "Bayesian modeling of uncertainty in low-level vision," *Int'l Journal of Computer Vision*, vol. 5, no. 3, pp. 271–302, Dec 1990.

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

[15] 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.

[16] 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, pp. 1222–1239, Nov 2001.

[17] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," *Proc. Int'l Conf. Computer Vision*, pp. 508–515, July 2001.

[18] D. Marr and T. Poggio, "A computational theory of human stereo vision," *Proc. Royal Soc. London B*, vol. 204, pp. 301–328, 1979.

[19] G. Egnal and R. Wildes, "Detecting binocular half-occlusions: empirical comparisons of five approaches," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1127–1133, Aug 2002.

[20] O. Faugeras and F. Lustman, "Motion and structure-from-motion in a piecewise planar environment," *Int'l Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.

[21] S. Birchfield and C. Tomasi, "Multiway cut for stereo and motion with slanted surfaces," *Proc. Int'l Conf. Computer Vision*, vol. 1, pp. 489–495, 1999.

[22] M. Lin and C. Tomasi, "Surfaces with occlusions from layered stereo," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. I–710–I–717, June 2003.

[23] F. Devernay and O. Faugeras, "Computing differential properties of 3-D shapes from stereoscopic images without 3-D models," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 208–213, 1994.

[24] A. Ogale and Y. Aloimonos, "Stereo correspondence with slanted surfaces: critical implications of horizontal slant," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 568–573, June 2004.

[25] Y. Boykov, O. Veksler, and R. Zabih, "A variable window approach to early vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1283–1294, Dec 1998.

[26] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 401–406, 1998.

[27] B. Rogers and M. Graham, "Anisotropies in the perception of three-dimensional surfaces," *Science*, vol. 221, pp. 1409–1411, 1983.

[28] G. Mitchison and S. McKee, "Mechanisms underlying the anisotropy of sterescopic tilt perception," *Vision Research*, vol. 30, no. 11, pp. 1781–1791, 1990.

[29] B. Gilliam and C. Ryan, "Perspective, orientation disparity, and anisotropy in sterescopic slant perception," *Perception*, vol. 21, pp. 427–439, 1992.

[30] R. Caganello and B. Rogers, "Anisotropies in the perception of sterescopic surfaces: the role of orientation disparity," *Vision Research*, vol. 33, no. 16, pp. 2189–2201, 1993.

[31] C. Ryan and B. Gilliam, "Cue conflict and sterescopic surface slant about horizontal and vertical axes," *Perception*, vol. 23, pp. 645–658, 1994.

[32] J. Hillis, S. Watt, M. Landy, and M. Banks, "Slant from texture and disparity cues: optimal cue combination," *Journal of Vision*, vol. 4, pp. 967–992, 2004.

[33] A. Ogale and Y. Aloimonos, "Robust contrast invariant stereo correspondence," in *Proc. IEEE Conf. on Robotics and Automation*, April 2005.

[34] A. S. Ogale, "The compositional character of visual correspondence," Ph.D. dissertation, University of Maryland, College Park, August 2004.