

Stereo correspondence with slanted surfaces: critical implications of horizontal slant

Abhijit S. Ogale and Yiannis Aloimonos

Center for Automation Research, University of Maryland, College Park, MD 20742
{ogale, yiannis}@cfar.umd.edu

Abstract

We examine the stereo correspondence problem in the presence of slanted scene surfaces. In particular, we highlight a previously overlooked 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, intensity matching metrics such as the popular Birchfield-Tomasi procedure 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 a 3D uniqueness constraint. This paper discusses these new constraints and provides a simple scanline based matching algorithm for illustration. We experimentally demonstrate test cases where existing algorithms fail, and how the incorporation of these new constraints provides correct results. Experimental comparisons of the scanline based algorithm with standard data sets are also provided.

1. Introduction

The dense stereo correspondence problem consists of finding a mapping between the points in two images of a scene. If the images have been rectified, then a point P in one image may correspond to a point P' in the other image, where P and P' lie on the same horizontal scanline. The difference in the horizontal position of P and P' is termed as horizontal disparity. In this paper, we assume that we are dealing with a rectified pair of images.

1.1. Previous work

There exists a considerable body of work on the dense stereo correspondence problem. Scharstein and Szeliski [19] 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 [16]. Better approaches utilize multiple windows [8, 7], adaptive windows [10] which change their size in order to minimize the error, shiftable windows [4, 21], or predicted windows [14], 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 [15], simulated annealing [9, 1], relaxation labeling [20], non-linear diffusion [18], maximum flow [17] and graph cuts [5, 11]. 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 (eg. [11]). The uniqueness constraint [13] is often used to find regions of occlusion. Egnal and Wildes [6] provide comparisons of various approaches for finding occlusions.

Recently, some algorithms [3] have explicitly incorporated the estimation of slant while performing the estimation of horizontal disparity. Lin and Tomasi [12] 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.

1.2. Our approach

We explicitly examine the stereo correspondence problem in the presence of horizontally slanted scene surfaces. In particular, we lay emphasis on the following geometric effect: a horizontally slanted surface (ie. 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 dealt with. Also, the uniqueness constraint, which is often used to find occlusions by forcing a one-to-one correspondence between pixels, is not true for horizontally slanted surfaces, since a N -to- M correspondence is possible. Hence, the uniqueness constraint must be reformulated in terms of scene visibility in the presence of horizontally slanted surfaces. In Section 2, we examine the above ideas and underscore the need for the treatment of horizontal slant in the first stage of any stereo algorithm during disparity estimation itself, rather than as a post-processing or a feedback step. For the sake of illustration, we present a simple scanline based algorithm in Section 3 which makes use of these constraints, and provide experimental comparisons with existing algorithms using standard data sets in Section 4.

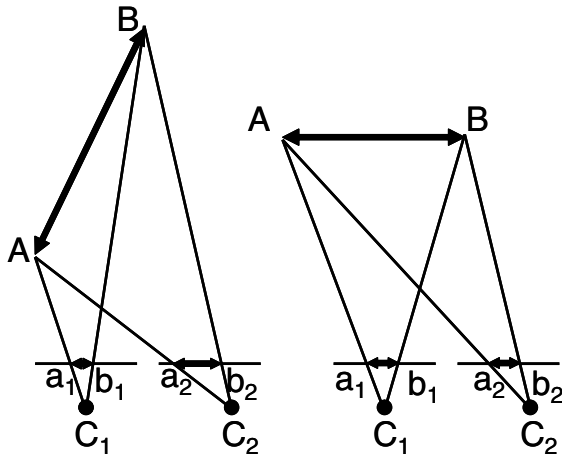


Figure 1. (Left) unequal projection lengths of a horizontally slanted line (Right) equal projection lengths of a fronto-parallel line

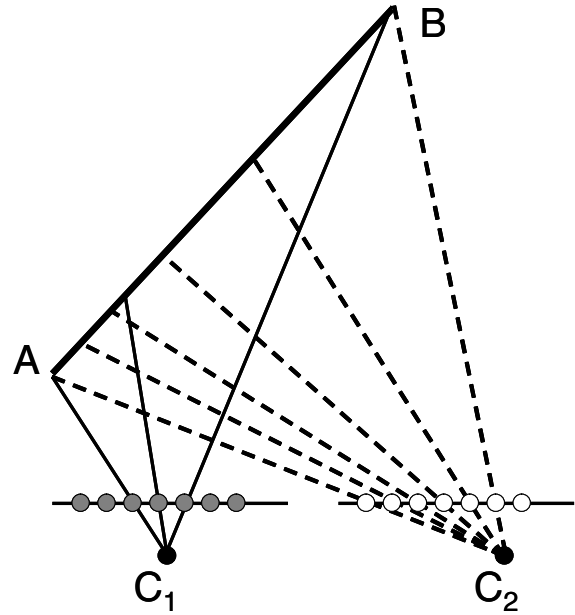


Figure 2. Sampling problem for a horizontally slanted line

2. Effects of Horizontal Slant

2.1. Unequal projection lengths

Using a 1D camera, Figure 1 shows on the left, how 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.

$$\begin{aligned} L_1 &= X_B/Z_B - X_A/Z_A \\ L_2 &= (X_B - t)/Z_B - (X_A - t)/Z_A \end{aligned} \quad (1)$$

Clearly, in general, L_1 and L_2 are not equal. For the fronto-parallel line shown in Figure 1 on the right, $Z_A = Z_B = Z$, hence

$$L_1 = L_2 = (X_B - X_A)/Z \quad (2)$$

Thus, except for the fronto-parallel case, horizontally slanted line segments in space will always project onto segments of different lengths in the two cameras. Hence, 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 therefore make a provision in our stereo algorithms to permit unequal correspondences of this nature.

2.2. 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 2. Birchfield and Tomasi [2] have provided a very useful method for matching pixel intensities, which is insensitive to image sampling. However, due to unequal sampling in the presence of horizontal slant, we must first resample each scanline correctly, and then apply the Birchfield-Tomasi matching procedure, which only uses nearest neighbor pixels for interpolation. In other words, we first stretch (resample) one of the scanlines, by an amount related to the slant we are considering, and then match this stretched scanline with the other unstretched scanline using the Birchfield-Tomasi matching process as usual. For example, if we are considering the linear correspondence function $x_2 = mx_1 + d$ between points of camera 1 and 2, then we must stretch the image of camera 1 by a factor m before performing the intensity based matching.

2.3. Occlusions and the uniqueness constraint

The uniqueness constraint [13] is often used to find occlusions. In its present form, the uniqueness constraint forces a one-to-one correspondence between pixels in the two images. In the end, 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 must modify the uniqueness constraint so that *we enforce 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 3 shows how the modified uniqueness constraint is used. 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 (ie. the ordering constraint is not needed). Now, in part (b), we wish to insert a new pair of corresponding intervals, shown by dashed lines. (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 gray). In order to enforce uniqueness, the gray pair of intervals on the right must be removed, while the gray 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 gray, and the inserted interval is shown as dashed.

3. Scanline stereo algorithm

We now describe a simple algorithm to illustrate how the above ideas may be implemented. For simplicity, the algorithm processes a pair of scanlines $I_L(x)$ and $I_R(x)$ at a time without using any vertical consistency constraints (the results are post-processed by a simple median filter). 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 $[-\Delta_2, -\Delta_1]$. Notice that 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

$$x_R = m_L(x_L) \cdot x_L + d_L(x_L)$$

and reciprocally:

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

Clearly,

$$\begin{aligned} m_R(x_R) &= 1/m_L(x_L) \\ d_R(x_R) &= -d_L(x_L)/m_L(x_L) \end{aligned}$$

The disparities are then computed as:

$$\begin{aligned} \Delta_L(x_L) &= x_R - x_L = (m_L(x_L) - 1) \cdot x_L + d_L(x_L) \\ \Delta_R(x_R) &= x_L - x_R = (m_R(x_R) - 1) \cdot x_R + d_R(x_R) \end{aligned}$$

The function m_L and m_R are 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 . The best value of m_L and d_L for a point is chosen such that it maximizes

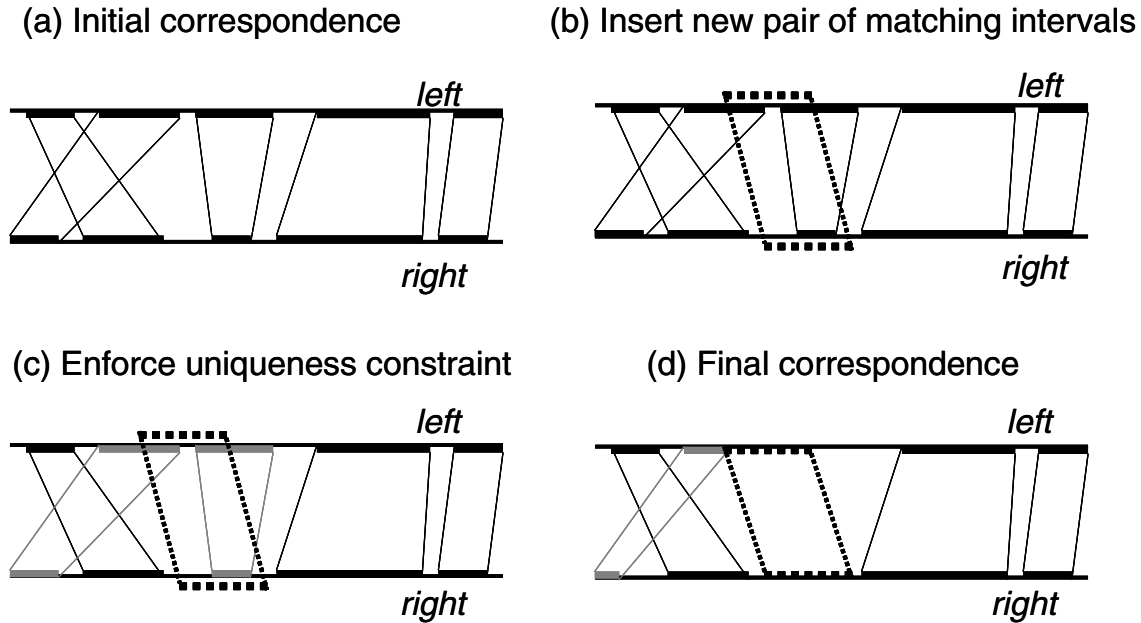


Figure 3. The modified uniqueness constraint operates by preserving a one-to-one correspondence between intervals on the left and right scanlines, instead of pixels

the size of the matching line segment containing that point. This is the simple global optimization which we perform to choose among the possible disparities.

The values of the horizontal slant which are to be examined are provided as inputs, ie. $m_L, m_R \in M$, where $M = \{m_1, m_2, \dots, m_k\}$. Thus, given the possible slants M and the disparity search range $[\Delta_1, \Delta_2]$, the possible values of d_L and d_R for each position can be restricted.

In order to find the occlusions, we enforce the uniqueness constraint in its modified form as shown in Figure 3. 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 previous correspondences of segments in S_L which overlap with i_L are removed, and the same is done for i_R and S_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.

4. Experiments

Scharstein and Szeliski [19] have set up a test suite (at www.middlebury.edu/stereo) of stereo image pairs along with ground truth disparities for comparing the results of dense stereo algorithms. The disparity map d_{out} generated by an algorithm is compared to the true disparity d_{true} , and the pixels which deviate by more than 1 unit from their true disparity are termed as ‘bad’ pixels. The percentages 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 4) using the scanline algorithm to the Middlebury website created by Scharstein et al (mentioned earlier). The simple scanline algorithm presented earlier (denoted ‘slanted scanline’ in the table) ranks ninth overall, while the ranks in each column are showed in brackets, below the error percentages. This performance evaluation is presented only for the sake of completeness, since the primary purpose of this paper is not to provide an algorithm, but rather to understand the effects of horizontal slant, and propose methods for correctly dealing with them. We expect that the constraints presented above will improve the results of many existing stereo algorithms.

The correctness of our approach immediately becomes evident when dealing with the stereo pair shown in Figure 5. This pair of test images shows a black object which is hor-

Table 1. Performance comparison from the Middlebury Stereo Vision Page (overall rank is 9'th among 29 algorithms). The table shows only the top ten algorithms. Error percentages and rank in each column (in brackets) is shown.

Rank	Algorithm	Tsukuba			Sawtooth			Venus			Map	
		all	untex	disc.	all	untex	disc.	all	untex	disc.	all	disc.
1	Segm.-based GC	1.23 (3)	0.29 (2)	6.94 (4)	0.30 (1)	0.00 (1)	3.24 (1)	0.08 (1)	0.01 (1)	1.39 (1)	1.49 (21)	15.46 (26)
2	Layered	1.58 (5)	1.06 (7)	8.82 (6)	0.34 (2)	0.00 (1)	3.35 (2)	1.52 (9)	2.96 (18)	2.62 (3)	0.37 (11)	5.24 (11)
3	Belief prop	1.15 (1)	0.42 (3)	6.31 (1)	0.98 (9)	0.30 (14)	4.83 (6)	1.00 (5)	0.76 (5)	9.13 (13)	0.84 (18)	5.27 (12)
4	MultCam GC	1.85 (9)	1.94 (14)	6.99 (5)	0.62 (6)	0.00 (1)	6.86 (11)	1.21 (7)	1.96 (9)	5.71 (7)	0.31 (8)	4.34 (10)
5	GC+occl 2b	1.19 (2)	0.23 (1)	6.71 (2)	0.73 (8)	0.11 (8)	5.71 (8)	1.64 (12)	2.75 (16)	5.41 (6)	0.61 (14)	6.05 (13)
6	Impr. Coop.	1.67 (6)	0.77 (5)	9.67 (10)	1.21 (13)	0.17 (11)	6.90 (12)	1.04 (6)	1.07 (6)	13.68 (18)	0.29 (6)	3.65 (7)
7	GC+occl. 2a	1.27 (4)	0.43 (4)	6.90 (3)	0.36 (3)	0.00 (1)	3.65 (3)	2.79 (20)	5.39 (21)	2.54 (2)	1.79 (22)	10.08 (20)
8	Disc. pres.	1.78 (7)	1.22 (10)	9.71 (11)	1.17 (11)	0.08 (7)	5.55 (7)	1.61 (11)	2.25 (12)	9.06 (12)	0.32 (9)	3.33 (6)
→ 9	<i>Slanted Scanline</i>	1.82 (8)	1.09 (8)	9.47 (8)	0.72 (7)	0.24 (13)	6.00 (9)	3.25 (21)	5.73 (22)	8.51 (11)	0.22 (2)	3.10 (4)
10	Graph cuts	1.94 (11)	1.09 (9)	9.49 (9)	1.30 (15)	0.06 (6)	6.34 (10)	1.79 (15)	2.61 (15)	6.91 (8)	0.31 (7)	3.88 (8)
		↓			↓			↓				
29	Max. surf.	11.10 (29)	10.70 (27)	41.99 (29)	5.51 (29)	5.56 (29)	27.39 (28)	4.36 (24)	4.78 (20)	41.13 (28)	4.17 (28)	27.88 (28)

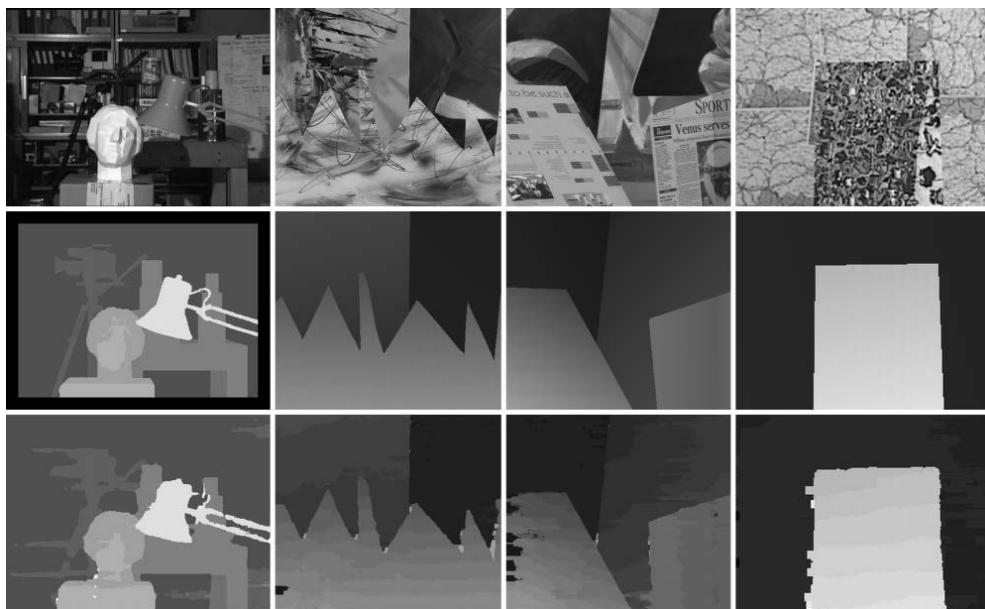


Figure 4. Top row (Left frames), Middle row (ground truth), Bottom row (our results). Occlusions were filled in before performing the evaluation.

izontally slanted (depth decreases from left to right). The second row of the figure shows on the left the output of the graph cuts algorithm of Kolmogorov et al [11]. The graph cuts result was obtained using software kindly provided by the authors (www.cs.cornell.edu/People/vnk/software.html). Our results are shown in the second row on the right hand side. The graph cuts algorithm finds a constant disparity value in the interior of the slanted object, which is clearly incorrect. Our algorithm correctly shows the disparity of the slanted object linearly decreasing from left to right (from white to dark gray). The detected occlusions are shown in black.

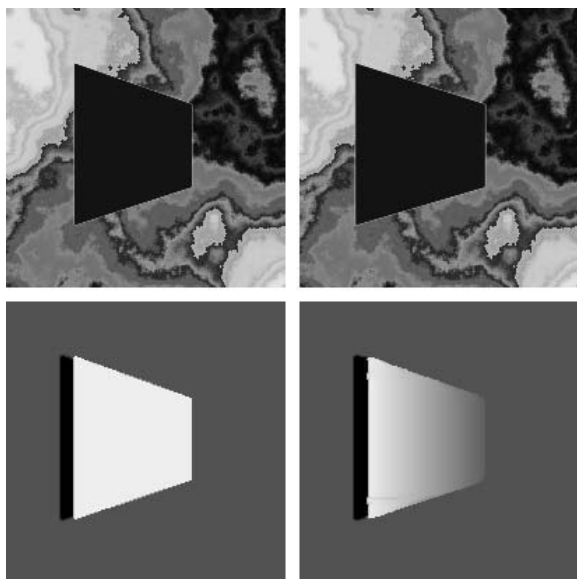


Figure 5. Horizontally slanted object. Top row: left image, right image. Bottom row: (left) results using graph cuts [11], (right) our results. Occlusions are shown in black

5. Conclusions

We have discussed the effects of horizontal slant on the stereo correspondence problem. We have shown that horizontal slant leads to unequal projections in the two cameras, which requires us to modify stereo algorithms for allowing M-to-N pixel correspondences. Furthermore, we have shown that horizontal slant leads to uneven sampling of a surface by the two cameras, and hence local intensity matching metrics must be suitably modified. Finally, the uniqueness constraint for finding occlusions, which imposes a one-to-one correspondence between image pixels, must be modified to enforce a one-to-one correspondence between scanline intervals instead of pixels. We have also

presented a simple scanline based algorithm which implements these constraints, and provided experimental comparisons with existing methods.

References

- [1] S. T. Barnard. Stochastic stereo matching over scale. *IJCV*, 3(1):17–32, 1989.
- [2] S. Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Trans. PAMI*, 20(4):401–406, 1998.
- [3] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. *ICCV*, 1:489–495, 1999.
- [4] A. F. Bobick and S. S. Intille. Large occlusion stereo. *IJCV*, 33(3):181–200, Sept 1999.
- [5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. PAMI*, 23(11):1222–1239, Nov 2001.
- [6] G. Egnal and R. Wildes. Detecting binocular half-occlusions: empirical comparisons of five approaches. *IEEE Trans. PAMI*, 24(8):1127–1133, Aug 2002.
- [7] A. Fusiello, V. Roberto, and E. Trucco. Efficient stereo with multiple windowing. *CVPR*, pages 858–863, June 1997.
- [8] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. *ECCV*, pages 425–433, 1992.
- [9] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. PAMI*, 6(6):721–741, Nov 1984.
- [10] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: theory and experiment. *IEEE Trans. PAMI*, 16(9):920–932, 1994.
- [11] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. *ICCV*, pages 508–515, July 2001.
- [12] M. Lin and C. Tomasi. Surfaces with occlusions from layered stereo. *CVPR*, 1:1–710–717, June 2003.
- [13] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. Royal Soc. London B*, 204:301–328, 1979.
- [14] J. Mulligan and K. Daniilidis. Predicting disparity windows for real-time stereo. *Lecture Notes in Computer Science*, 1842:220–235, 2000.
- [15] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Trans. PAMI*, 7(2):139–154, March 1985.
- [16] M. Okutomi and T. Kanade. A multiple baseline stereo. *IEEE Trans. PAMI*, 15(4):353–363, April 1993.
- [17] S. Roy and I. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. *ICCV*, pages 492–499, 1998.
- [18] D. Scharstein and R. Szeliski. Stereo matching with nonlinear diffusion. *IJCV*, 28(2):155–174, 1998.
- [19] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1):7–42, April 2002.
- [20] R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *IJCV*, 5(3):271–302, Dec 1990.
- [21] H. Tao, H. Sawhney, and R. Kumar. A global matching framework for stereo computation. *ICCV*, 1:532–539, July 2001.