# Efficient Stereo with Accurate 3-D Boundaries

Mikhail Sizintsev Richard P. Wildes
Centre for Vision Research
York University
Toronto, ON, Canada

**Abstract**

This paper presents methods for recovering accurate binocular disparity estimates in the vicinity of 3D surface discontinuities. Of particular concern are methods that impact coarse-to-fine, block matching as it forms the basis of the fastest and resource efficient disparity estimation procedures. Two advances are put forth. First, a novel approach to coarse-to-fine processing is presented that adapts match window support across scale to ameliorate corruption of disparity estimates near 3D boundaries. Second, a novel formulation of half-occlusion cues within the coarse-to-fine, block matching framework is described to inhibit false matches that can arise in regions near occlusions. Empirical results show that incorporation of these advances in coarse-to-fine, block matching reduces disparity errors by more than a factor of two, while performing little extra computation.

## 1 Introduction

Recent advances in binocular stereo have led to impressive results. Particularly notable performance in recovered disparity has arisen via application of global optimization methods. In contrast, the fastest methods still mostly rely on local block matching. Further, local matching often uses coarse-to-fine (CTF) refinement. CTF helps remove local minima in search by their reduction at coarser resolutions. As commonly embedded in image pyramids (where image sampling is commensurate with scale) ensuing processing can reduce match ambiguities, as large match windows at fine resolution are covered by smaller windows at coarse resolution. Also, processing speed increases as large disparities at fine resolution can be recovered at coarse resolution with smaller search range (subject to refinement at finer resolution). While recent advances in global methods improve speed [1, 6], block matchers, often with CTF, remain preferred when speed is a concern; such procedures inherently entail lower processing demands, map well to current hardware and software architectures [1, 20, 3] and are suitable for parallel and pipeline computation.

For both local and global methods of disparity estimation, reliable recovery in the vicinity of 3D surface boundaries remains a matter of concern. This problem is of particular note in conjunction with CTF approaches, which tend to resolve poorly such geometries as they are not well represented at coarser resolutions. In the past, much research has considered recovery of binocular disparity near 3D boundaries [14, 5, 1]. For local methods, the use of adaptive spatial support for match windows can ameliorate issues arising in attempts to match near 3D discontinuities by shaping windows to avoid poorly defined matches [8, 9, 19, 21]. Many recent advances in disparity estimation near 3D boundaries

BMVC 2006 doi:10.5244/C.20.25

explicitly consider half-occlusions, where one view sees portions of a background surface that are occluded to the other view by a foreground surface. Some of the most impressive recent results have been demonstrated in conjunction with global methods [10, 13, 17, 4]. In comparison, empirical investigation of half-occlusion detection with local processing underlines shortcomings [5].

This paper presents methods for improved disparity estimates within the CTF, block matching framework with a particular emphasis on resolving information in the vicinity of 3D boundaries. The major contributions of the present research are as follows. (i) A novel approach to adaptively defining match support in CTF refinement is used to ameliorate corruption of 3D boundaries. (ii) A novel method for processing half-occlusions is developed to prevent matches in such areas during CTF estimation. (iii) In empirical evaluation, the overall approach yields disparity estimates with significant improvement over standard CTF, block matching, while preserving efficiency.

## 2 Adaptive coarse-to-fine processing

For present concerns, the basic elements of coarse-to-fine (CTF), block binocular matching are as follows (see [14, 1] and references therein). Initially both images are brought into image pyramid representations via repeated filtering and subsampling. The disparity map is estimated for the coarsest level, then upsampled and scaled (implicitly or explicitly) to the next finer pyramid level where it serves to provide an initial estimate for refined matching. The procedure continues until the finest resolution level is reached. At each level disparity is estimated via basic block-based matching [14, 1].

There are two major sources of errors near 3-D discontinuities that arise in a CTF block-matching: i) foreground fattening/shrinking effect due to match windows of fixed shape, ii) refining incorrect disparity upsampled from the disparity map estimated at coarser resolution. While the first flaw of block matcher is well-researched and a number of remedies exist via introduction of shiftable/overlapping/adaptive windows [8, 9, 19, 21], the second one is specific to CTF and has not been paid enough attention. Thus, in the following, more discussion is devoted to the disparity upsampling procedure.

Consider the process of initializing a disparity map at some pyramid level based on estimates from the previous coarser level. Initialization via standard upsampling (e.g., nearest-neighbor, bilinear or Gaussian interpolation) of the disparity map recovered at the coarse level leads to difficulties in the vicinity of underlying disparity discontinuities. Depending on specifics of the situation, upsampled disparities near discontinuities can be incorrectly initialized from a wrong side of the discontinuity or come as an average across the discontinuity. In either case, subsequent refinement often cannot correct for the poor initialization and recovered surface geometry is compromised near 3D boundaries.

The snapshot of CTF estimation in Fig. 1 makes matters more precise. Disparity initialization (offset) for point $x$ can come from coarser level points $a$, $b$, $c$, $d$ or even more distant locations, because information at $x$ was implicitly used to calculate the coarse disparities (i.e. $x$ participated in constructing the low resolution image). If $x$ belongs to a constant disparity region, then disparity values at neighbouring black points (Fig. 1) would be the same and it makes no difference which one is used for initialization. In contrast, if $x$ is near a 3D boundary (i.e., boundary between regions with distinct disparities), then it is appropriate to search for finer disparities at $x$ using each possible initialization

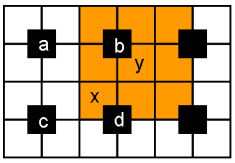


Figure 1: Snapshot of the Coarse-To-Fine (CTF) Estimation Procedure. White cells are pixels at fine level, black pixels are from coarse level. Window size is 3x3. Disparity offset for pixel *x* can be from one of disparities at points *a*, *b*, *c* or *d* (scaled by 2).

separately, i.e., as obtained from *a*, *b*, *c*, *d* (or even broader areas, if larger windows are used). In theory, systematic consideration of all possible CTF initializations should improve estimates near discontinuities as subsequent refinement will be less subject to poor starting disparities, as derived from information across different surfaces.

Brute-force realization of the above observations entails additional search at each finer level (one search for each initialization), with final disparity assignment taken as that yielding the best score under the block-matching metric. A closer look suggests a more efficient approach and one that also selects for the best shifted match window about each point. In Fig. 1, if initialization from *b* gives the best match for finer level refinement at *x*, then *x* and *b* derive from the same surface. Correspondingly, the best (e.g., 3x3 in Fig. 1) shifted match window for *x* would be as shaded. Significantly, the selected window is centered about point *y*, which gets correct initialization from *b* via nearest-neighbor upsampling. Analogous conclusions are drawn when the best initialization for *x* derives from *a*, *c* or *d*. In general, the best initialization, match window and refinement for *x* are achieved via nearest neighbor upsampling of the coarser disparity map and subsequent selection of the best disparity estimate derived across all shifted windows that cover *x* at the finer level. It is not necessary to try all window shifts for all initializations, as the correct disparity offset and window configuration are very tightly coupled.

*The desired computations for each pyramid level can be realized efficiently in two steps:* i) obtain an initial disparity map via central window block matching using Nearest Neighbour upsampled coarse disparity as offset; ii) finalize the disparity map at each pixel by choosing the disparity of the neighboring pixel that has the best match score; here, the neighbourhood is that covered by the match window. The latter step is similar to morphological operation on the match score map (dilation for Normalized Cross Correlation match measure) using the aggregation window as a structural element to simulate shiftable windows in single-scale matching [14]. Note that the proposed approach is *not* identical to estimating disparity estimates at each level via shiftable windows [8], because, for each pixel, each shifted window corresponds to a different disparity offset.

It is well known that CTF disparity estimation corrupts 3D boundaries. In non-CTF block matching, use of shiftable or otherwise adaptive windows to conform to disparity discontinuities is well established [8, 9, 19, 21]; however, the link to improving CTF disparity refinement seems not to have been stated previously. Recent work that exploits CTF processing for disparity estimation beyond block matching, e.g., with global methods [18, 16, 7, 6], has yielded strong results; however, the importance of considering multiple offsets in projecting CTF has not been addressed clearly. Ideally, these methods should use multiple offsets; whereas, the proposed method is naturally more efficient – window placement and disparity offset are tied to eliminate extra search.

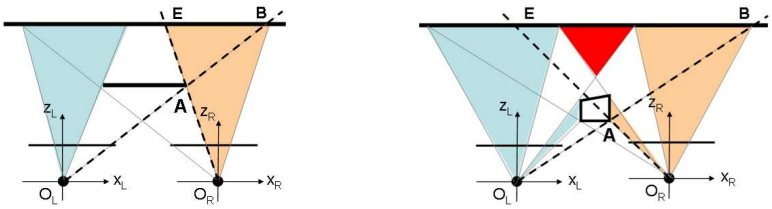


Figure 2: Two Cases of Half-Occlusion Formation. Left: All points on the back surface that are within the forbidden zone of the boundaries of the front surface are half-occluded, e.g., A is the right boundary point of the front surface. Right: Narrower front surfaces allow portions of the back surface within the forbidden zone of the front surface boundaries to be binocularly visible. Further interposed surfaces in the red (dark grey) region allow for half-occlusion relations to occur *recursively*.

# 3 Half-occlusion processing

Given the goal of improved disparity estimates in the vicinity of 3D discontinuities, it is appropriate to account explicitly for binocular half-occlusions, as they bear a close relationship with discontinuous surface geometry. Two illustrative cases are depicted in Fig. 2. Note that angles $EAB$ and $O_LAO_R$ encompass the forbidden zone for point **A**, the region where other points will appear as violations of the ordering constraint [11].

For present purposes, a useful constraint for half-occlusion processing comes by considering the difference in disparity on either side of the occlusion region and region width. Consider the shaded region on the right side of Fig. 2a. Let world point **A** be the left-most point that is binocularly visible, while world point **B** is the right-most half-occluded point (visible only to the right image); let their right image coordinates along a scanline be $a_r$ and $b_r$, resp. (For ease of exposition: Images are taken as rectified; image coordinates are specified as scalars along a given scanline.) The width of the half-occluded region projected to the right image is

$$\Omega_r^w(\mathbf{B},\mathbf{A}) = b_r - a_r. \tag{1}$$

The disparity values for points **A** and **B** are

$$d_r(\mathbf{A}) = \mathbf{a}_l - \mathbf{a}_r; \;\; d_r(\mathbf{B}) = \mathbf{b}_l - \mathbf{b}_r = \mathbf{a}_l - \mathbf{b}_r \tag{2}$$

with $b_l = a_l$ by construction. Correspondingly, the change in disparity across the half-occluded region is given as

$$\Delta d_r(\mathbf{B},\mathbf{A}) = d_r(\mathbf{B}) - d_r(\mathbf{A}) = a_l - b_r - (a_l - a_r) = a_r - b_r \tag{3}$$

Now, taking the ratio of disparity change (3) to occlusion width (1) it is found that

$$\frac{\Delta d_r(\mathbf{B},\mathbf{A})}{\Omega_r^w(\mathbf{B},\mathbf{A})} = \frac{a_r - b_r}{b_r - a_r} = -1. \tag{4}$$

It is seen that this ratio is equal to the disparity gradient limit [2]. Further consideration of the geometry illustrated in Fig. 2 shows that relationship (4) between disparity change and occlusion width also holds for regions visible only to the left view of a binocular pair. In this form, the constraint will be referred to as the *disparity-change/width constraint* in the following. Note that "occlusion width" refers to the region where occlusion *can* appear. Depending on the situation, the whole area can be occluded (Fig. 2a), or it can

have gaps of binocular visibility (Fig. 2b). The loci of points that yield the value of -1 for the disparity gradient limit lie along a boundary of the forbidden zone, e.g. the line through $\mathbf{A}, \mathbf{O}_l$ (and hence $\mathbf{B}$) in Fig. 2a.

Disparity-change/width can be related to the "uniqueness constraint", i.e., that each point in one image should match to only one in the other: Rearrangement of the terms in (4) with substitution from (1) and (3) yields

$$d_r(\mathbf{A}) + a_r = d_r(\mathbf{B}) + b_r, \tag{5}$$

i.e. , uniqueness is violated as both $a_r$ and $b_r$ map to the same location in the left image.

The derived formulae, (4) or (5), can be used to detect half-occlusions. In the following, most emphasis is on (5) as it yields a convenient algorithm. To arbitrate further between visibility and occlusion a second cue to half-occlusion is employed. Since matches in occluded areas have no physically defined match (corresponding points are not imaged to the other view), any attempted match is expected to have a poor match score, at least for areas having distinctive texture. So, given two or more points satisfying (5), the point with the best match score is taken as binocularly visible, and the others as half-occluded.

In practice, straightforward use of uniqueness, (5), is not robust to slanted surfaces [10] and continuous disparity, as integer disparity quantization can cause multiple pixels in one image to map to a single pixel in the other. We deal with this problem efficiently as follows. Integer disparity values are interpolated to subpixel precision (we use parabola fitting around the match peak [9]). Subsequently, disparity relations between adjacent pixels on a scanline are used to group pixels into equivalence classes according to whether or not they are consistent with a single continuous surface. Given this grouping: Pixels consistent with a single surface cannot engage in half-occlusion relationships (violation of uniqueness credited to disparity quantization issues). The criterion for grouping adjacent pixels as arising from a single surface is $\|\Delta d\| < 1$, with $\Delta d$ is the interpixel disparity difference. This criterion is based on the widely used *occlusion* ($\Delta d \geq 1$) and *ordering* constraints ($\Delta d \leq -1$), as they both imply depth discontinuities [5, 1].

*The proposed approach to half-occlusion detection is summarized as follows:* For each scanline, form the surface equivalence classes by considering interpixel disparity difference and obtain the sets of points that violate uniqueness (5). Within each set, find the point which has best match score and declare it as visible; all other points in the set that are not in the same surface class as the visible point are occluded.

In the context of CTF matching, half-occlusion detection is doubly useful. Occlusions are detected on each scale followed by extrapolation of neighboring background surface disparity values into the occluded regions. Such processing in turn yields better ability to initialize disparity estimation at finer levels, esp. near half-occlusions.

The proposed approach to half-occlusion is most similar to others that also explicitly consider disparity of occluded and occluding surfaces. They include "occlusion" (OCC) [5, 17], "ordering" (ORD) [14, 5, 1] and "visibility" (VIS, an extension of "uniqueness") [13, 4] constraints. In turn, unlike OCC, our approach does not require two way matching; unlike ORD, our approach is more specific to occlusion detection, as it only encompasses the forbidden zone boundary, not the entire zone, and works in places where ORD is violated, yet physically correct (Fig. 2b); unlike VIS that enforces one-to-one mapping between continuous *intervals* by affine parameterized matching on segments, our approach is directly applicable to standard block-matching techniques.

The current approach also makes use of match scores in deciding which points are binocularly visible vs. half-occluded. This situation is similar to when global methods

set occlusion cost in their energy formulations to depend on local match scores available to a given pixel [14, 1]. Also similar is the use of inconsistencies between bidirectional matches (i.e., left-right-checking) to detect half-occlusion [5, 9]. While such approaches can detect half-occlusions, they are not specific to this situation; rather, they more generally diagnose problems in matching.

# 4 Empirical evaluation

The adaptive coarse-to-fine and half-occlusion processing advances have been implemented in software and tested on a standard test set *Tsukuba*, *Venus*, *Teddy* and *Cones* [12] and a dataset with naturalistic imagery (albeit no ground truth), *Rock* and *Stephen*.

Overall, three different algorithm are evaluated: **A1** - single scale with shiftable windows; **A2** - standard CTF; **A3** - proposed adaptive CTF with occlusion detection. We compare to A2 to show our improvements over standard CTF and to A1, as it is a strong single scale block matcher [14]. All three algorithms work on a Gaussian pyramid obtained from grayscale images (A1 works on the finest level only) and use the Normalized Cross Correlation (NCC) match measure [1]; for match windows, A1 uses $17 \times 17$ shiftable windows (which gave the best result in initial tests), A2 and A3 use 5x5 windows; A2 and A3 work over all attainable pyramid levels for a given image size (i.e. coarsest level autoselected when one image dimension becomes unity) and search $\pm 1$ pixel at each level; A1 searched the maximum disparity range for each test case. A more detailed evaluation can be found in our technical report [15].

For data sets with ground truth, three kinds of error statistics have been collected [12]: errors for nonoccluded pixels, all pixels including occluded and pixels near discontinuities. Average statistics for each class of errors are given by taking the weighted average over all 4 stereo pairs, with weights proportional to the number of image pixels. Fig. 3 shows disparity maps while Fig. 4 shows the error statistics.

Comparing A2 and A3, the introduction of the proposed approach to adaptive CTF processing and half-occlusion detection (A3) results in considerable improvement. It is expected that the adaptive approach bests standard CTF, as it was designed for exactly that purpose. Interestingly, adaptive CTF also bests single scale shiftable windows (A1), especially near discontinuities (white bars in Fig. 4); this can be explained by the fact that A3 can use smaller windows (5x5 vs. 17x17) to yield more precise boundary-fitting and search over small ranges (i.e. $\pm 1$ at each resolution) for less ambiguous matches.

Half occlusions are detected reliably, including sharp corners, e.g. tips of the cones in *Cones*, and slanted surfaces, e.g. *Teddy* and *Venus* (Fig. 3). Occlusion results are isolated in Tab. 1. As a comparison for local methods, Tab. 1 also shows results based on the often used left-right checking (LRC) [5, 9]. LRC yields a 7% average hit rate increase and 2.5 times higher false positive rate, supporting the claim that our method is more specific to half-occlusions and therefore better suited when seeking to distinguish 3D boundaries from other sources of match error (e.g., for segmentation).

Moreover, testing on *Tsukuba* and *Venus* exposes the relative weakness of the proposed algorithm (like any local algorithm) when operating in regions with little texture. Another apparent weakness is the lack of resolution for thin structures (typical to CTF processing), e.g. the lamp arms in *Tsukuba* and pencils in *Cones*.

The error statistics can be used for comparison to state-of-the-art solutions (see Tab. 2).

| Alg. | | Tsukuba | Venus | Teddy | Cones | Avg. |
|---|---|---|---|---|---|---|
| A3 | HR (%) | 46.63 | 63.56 | 81.53 | 77.92 | 69.39 |
| | FP (%) | 2.31 | 1.27 | 2.27 | 2.21 | 1.99 |
| LRC | HR (%) | 59.87 | 70.3 | 87.71 | 82.82 | 76.65 |
| | FP (%) | 8.74 | 3.76 | 6.64 | 5.07 | 5.75 |

Table 1: Half-Occlusion Detection for A3 & Left-Right-Checking. Hit rate (HR) & false positive rate (FP) as percent of pixels correctly and incorrectly marked occluded.

| **Alg.** | Avg. | **Tsukuba** | **Venus** | **Teddy** | **Cones** |
|---|---|---|---|---|---|
| | rank | nonocc all disc | nonocc all disc | nonocc all disc | nonocc all disc |
| **A3** | 14.8 | $10.2_{22}$ $11.5_{22}$ $20.3_{17}$ | $4.58_{19}$ $5.22_{18}$ $14.2_{15}$ | $8.39_{10}$ $13.7_{8}$ $20.0_{12}$ | $5.03_{11}$ $10.8_{9}$ $13.9_{12}$ |
| ... | ... | ... | ... | ... | ... |
| SSD+MF | 18.5 | $5.23_{19}$ $7.07_{18}$ $24.1_{20}$ | $3.74_{17}$ $5.16_{17}$ $11.9_{14}$ | $16.5_{20}$ $24.8_{19}$ $32.9_{20}$ | $10.6_{19}$ $19.8_{19}$ $26.3_{20}$ |

Table 2: Snapshot of the evaluation results on Middlebury stereo test-bed.

A3 ranks in the mid-range according to average rank; however, higher for the most complex *Teddy* and *Cones*, 8th and 9th, resp.

In any case, the *critical comparison* is that of A3 to standard CTF (e.g., A2), as a major goal of the present work is improved disparity estimates for this style of efficient processing; such improvement is clear in Figs. 3 and 4, e.g. average errors reduced by a factor of two or more for all error classes plotted in Fig. 4.

Results of running A1-A3 on images of more naturalistic scenes are shown in Fig. 5. These images reflect situations where the proposed algorithm would be more useful, e.g. scenes for automatic navigation. All estimations were done using the same parameters as in tests with the Middlebury dataset, Fig. 3. It is evident that the proposed CTF disparity estimation (A3) significantly outperforms standard CTF (A2) and also bests single-scale matching (A1), both in textureless regions and near 3-D boundaries.

Significantly, A3 has few parameters to tune. Window size can be small, as CTF allows greater aggregation ($5 \times 5$ used here). The tradeoff between search range on each pyramid level and number of levels (the more levels, the greater the possibility of artifacts, esp. near borders, but the smaller the search range) is not as critical as in standard CTF: The proposed algorithm ameliorates boundary errors during CTF refinement and has allowed the current results while starting at the coarsest attainable pyramid levels (i.e., when one of the image dimensions has been reduced to 1 pixel) with search of $\pm 1$ pixel at all levels. Moreover, half-occlusion detection is essentially parameter free.

Finally, speed is an important advantage of any CTF algorithm including the proposed algorithm. For image and match window sizes $m \times n$ and $w^2$, resp., the theoretical complexity is $O(mndw^2) = O(mnw^2)$ (search range at each pyramid level is $d = 1$), and can be decreased to $O(mn)$ via a running box filter implementation for window cost aggregation [16]. The advances over standard CTF that are embodied in A3 do not degrade this complexity. Moreover, the algorithm is suitable for parallel/pipeline implementation.

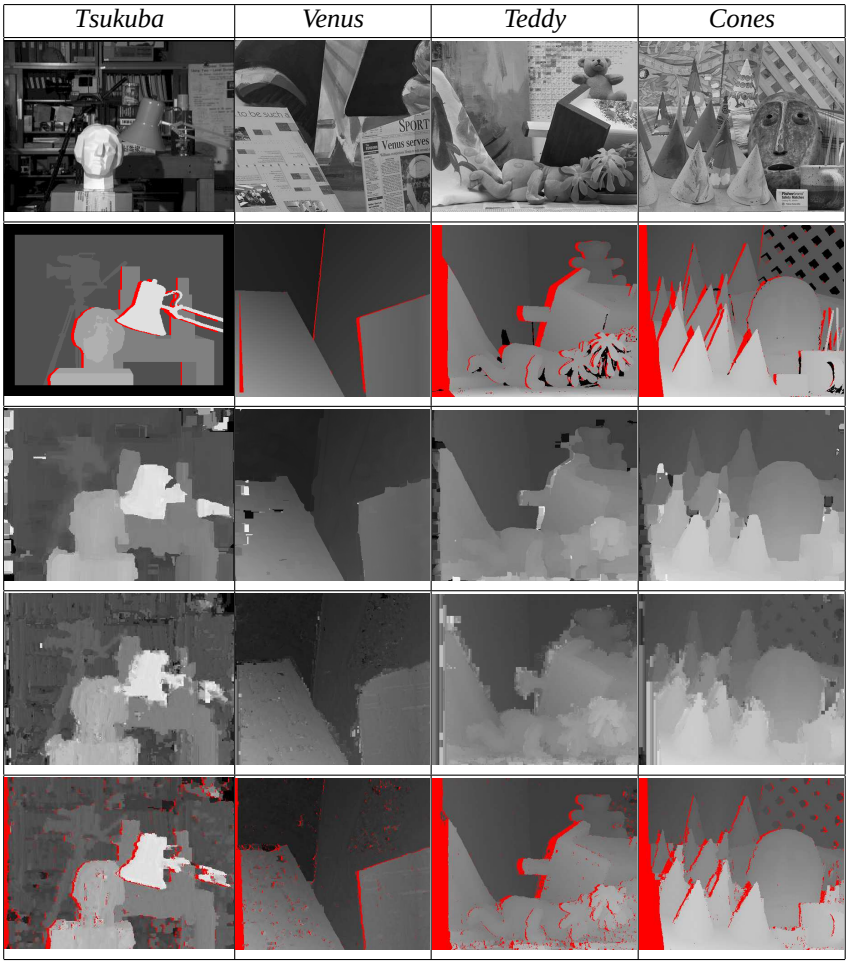


Figure 3: Disparity Maps Recovered for Middlebury Dataset [12]. Top-bottom: Left image, ground truth and recovered disparity maps using algorithms A1, A2 and A3. Red (dark grey) for A3 denotes half-occlusions.

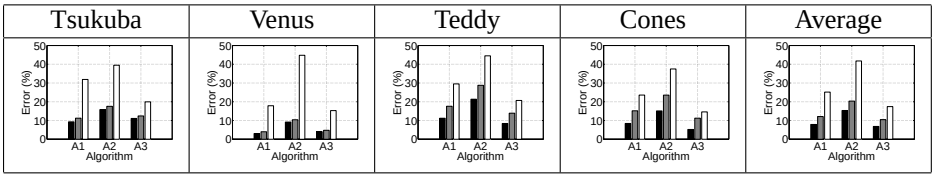


Figure 4: Error Statistics Across Algorithms. Triplet bars represents error statistics of A1-A3 for non-occluded (black), all (gray), and discontinuity (white) pixels.

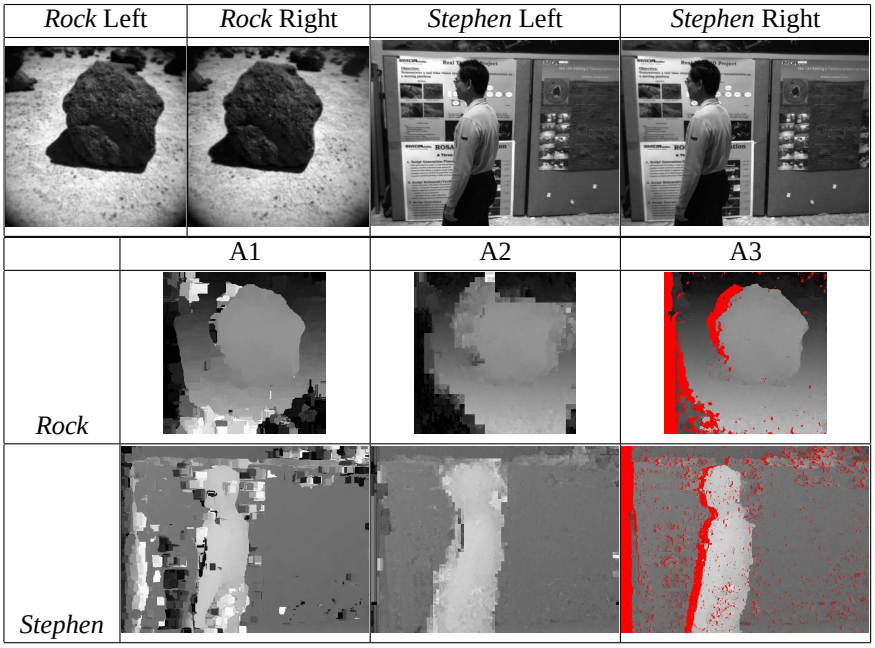


Figure 5: Results for Naturalistic Scenes. Red (dark grey) for A3 denotes half-occlusions.

# 5 Conclusion

This paper has presented simple, effective procedures for improving disparity estimates near 3D boundaries within coarse-to-fine (CTF), block-matching. The procedures entail adaptive CTF refinement that avoids corruption of disparities across 3D discontinuities and accurate half-occlusion recovery. Empirical evaluation of an embodiment of these advances in a CTF, block-matcher shows its superior performance in comparison to the same matcher without the proposed advances. Significantly, the enhanced disparity estimator enjoys the same efficient style of computation as does standard CTF, block-matching. In practice, the proposed advances should have considerable utility owing to their efficient, effective nature, small number of parameters and applicability to any CTF, block matcher.

**Acknowledgements**

This work was supported by MDA Space Missions (Brampton, Ontario, Canada), OCE and NSERC.

# References

[1] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *PAMI*, 25:993–1008, 2003.

[2] P. Burt and B. Julesz. A disparity gradient limit for binocular fusion. *Nature*, 208:615–617, 1980.

[3] N. Cornelis and L. Van Gool. Real-time depth computation using graphics hardware. In *CVPR*, pages 1099–1104, 2005.

[4] Y. Deng, Q. Yang, X. Lin, and X. Tang. A symmetric patch-based correspondence model for occlusion handling. In *ICCV*, pages 1316–1322, 2005.

[5] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *PAMI*, 24:1127–1133, 2002.

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. In *CVPR*, pages 261–268, 2004.

[7] S. Forstmann, Y. Kanou, J. Ohya, S. Thuering, and A. Schmitt. Real-time stereo by using dynamic programming. In *CVPRW*, volume 3, pages 29–35, 2004.

[8] A. Fusiello and V. Roberto. Efficient stereo with multiple windowing. In *CVPR*, pages 885–863, 1997.

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

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

[11] J. Krol and W. van der Grind. Rehabilitation of a classical notion of Panum's fusion area. *Perception*, 11:615–619, 1982.

[12] *http://www.middlebury.edu/stereo/*.

[13] A. S. Ogale and Y. Aloimonos. Stereo correspondence with slanted surfaces: critical implications of horizontal slant. In *CVPR*, pages 568–573, 2004.

[14] D. Scharstein and R. Szeliski. Taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002.

[15] M. Sizintsev and R. Wildes. Coarse-to-fine stereo vision with accurate 3D boundaries. Technical Report CS-2006-07, York University, Toronto, Canada, 2006.

[16] C. Sun. Fast stereo matching using rectangular subregioning and 3D maximum-surface techniques. *IJCV*, 47:99–117, 2002.

[17] J. Sun, Y. Li, S. B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. In *CVPR*, pages 399–406, 2005.

[18] G. Van Meerbergen, M. Vergauwen, M. Pollefeys, and L. Van Gool. A hierarchical symmetric stereo algorithm using dynamic programming. *IJCV*, 47:275–282, 2002.

[19] O. Veksler. Fast variable window for stereo correspondence using integral images. In *CVPR*, pages 556–561, 2003.

[20] R. Yang and M. Pollefeys. Multi-resolution real-time stereo on commodity graphics hardware. In *CVPR*, pages 211–217, 2003.

[21] K.-J. Yoon and I.-S. Kweon. Locally adaptive support-weight correspondence. In *CVPR*, volume 2, pages 924–931, 2005.