Non-Uniform Hierarchical Pyramid Stereo for Large Images

Sébastien Roy and Marc-Antoine Drouin

Département d'informatique et de recherche opérationnelle
C.P. 6128, Succ. Centre-Ville, Montréal, Québec, H3C 3J7
Email: royS@iro.umontreal.ca
Email: drouim@iro.umontreal.ca

Abstract

This paper addresses the stereo correspondence problem where the images are large enough to make stereo matching difficult. In order to reduce the problem size, we propose a new non-uniform hierarchical scheme with the ability to handle different coarseness levels simultaneously. Our framework, based on a maximum flow formulation, allows a much better localization of object boundaries where large depth discontinuities are present. The uniform decomposition fails to localize precisely such borders because it makes the assumption that surfaces are smooth in order to correct the errors from one coarseness level to the next. Our disparity estimation accurately localizes large depth discontinuities and then focus on increasing the resolution of smooth surfaces. Results on synthetic and real images demonstrate the validity of our framework.

1 Introduction

Modern digital cameras can generate images so large that many of the traditional pixel based stereo algorithms cannot process them. In a recent comparative study of such algorithms by Scharstein and Szeliski[1] the images were reduced by a factor of 16 in order to make them usable by all the tested algorithms. Hierarchical approaches have been introduced to deal with those high resolution images [2–10]. In those schemes multiple levels of image reduction are used to reduce the search space. Unfortunately, some matching errors made in an early stage can never be repaired in the following steps. Those errors appear mostly near objects boundaries. In order to minimize the error, we must provide a mechanism that can automatically compensate for errors introduced at lower resolutions.

Many approaches have been proposed to cope with errors induced by pyramids. In the context of terrain model reconstruction, Hung *et al.* [11] suggested using edges detection to help correct errors. Lotti and Giraudon [12, 13] proposed a pyramidal scheme based on cross-correlation where edge detection is used to determine the size of the correlation window. Park and Inoue [14] used an occlusion-overcoming strategy based on the use of 5 cameras coupled with a hierarchical scheme to achieve precise localization of object boundaries. The coarse-to-fine hierarchical schemes presented in [11, 12, 14] all use a uniform grid decomposition making them vulnerable to error propagation. Szeliski and Shum [15], in the context of optical flow, used a quadtree decomposition of the disparity map with a splitting criteria based on normal flow. The motion map is then obtained using the pre-computed pixel grid. Falkenhagen [16] used a standard pyramidal scheme and extends the search interval where large disparity variations are present. Leloglu *et al.* [17] also uses a standard pyramidal scheme where error propagation is limited by using a sphere around each match in the search volume. The search is then limited in the region covered by the different spheres. In [4, 7] a complex discrete wavelet transform is used to improve the matching cost function at each pyramid level. Nevertheless, these methods are very sensitive to errors that occur at the coarsest levels. Alvarez *et al.*[18] used a hierarchical scheme to speed up the convergence of an energy function minimization.

In this paper, we propose a new and more flexible hierarchical stereo algorithm that features a non-uniform spatial resolution, so levels of refinement can be applied selectively where they are needed most in order to preserve a good localization of objects boundaries. Our pyramid approach relies on the maximum-flow formulation of the matching problem [19] which allows an arbitrary non-

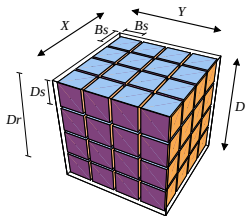

Figure 1: The stereo matching space. The (x, y) plane is the disparity map with blocks of size B_s. The d axis is the disparity to assign over a range D_r and disparity step D_s.

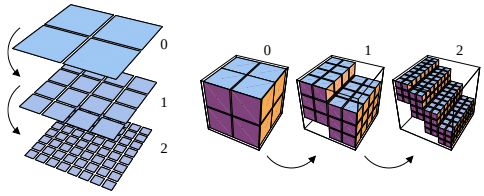

Figure 2: Classical pyramid approach. From one level to the next, notice an increase in spatial and disparity resolution combined to a reduction of the disparity range.

uniform pixels grid decomposition, independently of the epipolar constraint. This concept is similar to the *rectangle* presented by Sun [5] where sub-images with low varying disparity are independently matched using dynamic programming which restricts the non-uniformity to be along epipolar line and does not allow smoothing between successive epipolar lines. Furthermore, the non-uniform grid decomposition in [5] is only used to speed up computation and not to reduce error propagation. Mancini and Konrad [20] introduced a quadtree decomposition similar to our scheme. Their criteria to split a pixel block is based on the value of the matching cost function and does not model depth discontinuities explicitly. Sethuraman *et al.* [21, 22] proposed another quadtree decomposition of the disparity map in the context of stereoscopic image compression. The criteria to split a pixel block is based on the variation of disparity in the 4 inner blocks and does not consider neighborhood blocks as our method suggest. Their scheme is aimed at low bit transmission of stereo pairs and does not provide large and highly detailed disparity maps.

The concept of non-uniform pyramid will be described in Section 2, then the complete pyramid algorithm will be presented in Section 3. Experimental results will follow in Section 4.

2 Non-Uniform Pyramid

When working with large images the size of the solution space is so huge that the problem becomes untracable. The problem space is illustrated in Figure 1. The x and y axis represent the disparity map itself, while the d axis represents the disparity associated to each (x, y) pixel of the disparity map. Since the resolution of the disparity map does not necessarily correspond to the original image resolution, each disparity map pixel is in fact a block of size B_s (along each of its sides, assuming it is square) in image pixels (see Figure 1). The same goes for each disparity value, where going from one disparity value to the next corresponds to a step of D_s pixels in the original image. Also, the disparity values have a range D_r that describes the extent of displacement allowed for each block (x, y) of the disparity map. The goal is to assign for each (x, y) block of the disparity map a disparity d by searching all possible disparities in the range D_r using steps of D_s pixels. In order to solve very large problem instances, many algorithms use a *pyramid* approach where reduced versions of the problem space are successively solved at increasing resolutions while keeping the search space at a reasonable size [2–5, 11, 23].

Those classical pyramid algorithms generally start by applying a large reduction of both spatial and disparity resolutions (see Figure 2, level 0). After a first solution is obtained using a full disparity range (absolute phase), it gradually increases the spatial and disparity resolutions while reducing the disparity range to keep the problem size under control (relative phase;see Figure 2, levels 1 and 2). At various pyramid levels, it is expected that the computed disparity map will differ from the *true* disparity map. We classify the errors in two types.

The first type is *smooth surface* errors, which are induced by the reduced resolution of the search space. The reductions used by classical pyramid approaches are set up to compensate for *smooth surface* errors from the previous level. Those errors are progressively eliminated as the resolution is increased from one pyramid level to the next. For example, it is assumed that for a disparity step of 8 pixels, the disparity solution are within ± 8 pixels of the *true* disparity. The errors can be removed by

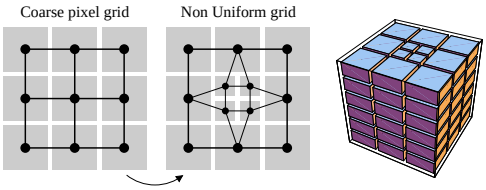

Figure 3: Non-uniform spatial reduction. Each grid is shown with superimposed neighborhood graph (G_{grid}). On the right, the resulting non-uniform problem space ($G_{grid} \times G_{disp}$).

setting the range of the next step to ± 8 around this solution while the disparity step is decreased to 4 pixels. For the purpose of this algorithm, we define a *smooth surface* as one that features disparity discontinuities smaller or equal to the disparity step D_s used at a given pyramid level.

The second type is *large discontinuity* errors which occur when the true disparity solution lies outside a disparity range that does not cover the full extent of allowed disparities, as in Figure 2 levels 1 and 2. Errors of this type cannot be recovered in a classical pyramid approach and are propagated through the next levels unaltered. Large depth discontinuities are typically observed at object boundaries. At such boundary, the disparity error can be as large as the full disparity range. This kind of error can occur at any pyramid level, but is much more prevalent when going from a full disparity range to a reduced one.

The non-uniform framework

In order to better localize object boundaries, we propose to increase *non-uniformly* the spatial resolution only where it is needed, that is where large discontinuities are occurring. This reduction allows to run a problem again with the same disparity range with a negligible increase in problem size, making it possible to reduce or remove the large discontinuity localization errors.

A non-uniform reduction consists in applying a reduction to only a selected subset of the problem space. The case where only spatial resolution is affected is of particular interest for solving large discontinuity errors. As illustrated in Figure 3, some blocks (here the one in the center) are selected and then *exploded* into 4 new smaller blocks.

The fact that blocks of many different sizes can be present in a single level of the pyramid creates new requirements on the problem space representation and the algorithm used for solving it. The following sections describe these issues in detail.

A Graph formulation

When solving for a disparity map over some problem space, most algorithms try to apply some form of smoothing constraint between neighboring pixels of the disparity map [19, 24]. This is usually applied as direct search with a large correlation window [25], dynamic programming [26], or maximum-flow [19, 27]. The maximum-flow method with linear penalty costs [19] can be easily adapted to a non-uniform disparity grid by simply changing the topology of the flow graph. It allows to solve efficiently the whole disparity map in a single global minimization and featured a very flexible neighborhood representation.

The graph G is defined, as in [19], as the product $G = G_{grid} \times G_{disp}$ of two basic graph, G_{grid} expressing the disparity map, and G_{disp} expressing the disparity range. In the graph G_{grid}, each disparity map block is a node and is connected by a *smoothness edge* to neighboring blocks. We consider two blocks to be neighbors when they have an adjacent face, regardless of their size. Figure 3 illustrates the graph G_{grid} as it undergoes a non-uniform transformation. The black points are nodes and the lines joining them are *smoothness edges*. Usually, a global *smoothness* flow capacity is given to all smoothness edges of the graph. Due to the irregular nature of neighborhoods, each smoothness edge is now defined as a common *smoothness flow capacity* multiplied by the size of the shared side of the neighboring blocks. In Figure 3, this is illustrated by the thicker lines linking large blocks and thinner lines linking a small block to its neighbor since they share a smaller common edge than two large blocks. This use of *weighted* smoothness allows the enforcement of a uniform degree of smoothness across a mixture of blocks of arbitrary sizes. The disparity graph G_{disp} is the same as in the regular grid case. It spans the full disparity range as a chain of nodes, one for every disparity step D_s. However, the final graph G will be sparse since only a fraction of the full disparity range will remain after the solution is computed.

This section has presented how a non-uniform pyramid level can be represented as a graph and

solved using a maximum-flow algorithm. The sequence of reductions used to generate all the pyramid levels is described in the following section.

3 The algorithm

All pyramid algorithms must start with an *absolute* phase where they solve over the full disparity range, and then proceed with a *relative* phase where they gradually reduce the disparity range until it reaches the desired resolution, usually one pixel.

Table 1 illustrates our non-uniform pyramid algorithm. As introduced in Figure 1, we define B_s as the **smallest** block size present in the problem space, D_s as the disparity step size used, and D_r the disparity range. For the purpose of expressing the range, $[d_{min} \ldots d_{max}]$ represents the full absolute range of allowed disparity while $S \pm d$ represents a range of plus or minus d pixels relative to a previous disparity solution S. A particular reduced problem instance P^i is describes by the vector $P^i = (B_s^i, D_s^i, D_r^i)$.

The classical algorithm solves the smallest problem instance over the full range of disparity. It then proceeds to gradually refine the solution while reducing the disparity range. The non-uniform algorithm proceeds similarly but contains extra steps which are shown in Table 1 and described next.

1	**Create** coarsest pyramid level $P^0 = (B_s^0, D_s^0, [d_{min} \ldots d_{max}])$ where B_s^0 and D_s^0 are based on memory and time constraints Select a threshold T for discontinuities
2	**Set** $i = 0$, then **Solve** problem P_0 to obtain solution S_0
2a	**Setup** next pyramid level $P_{i+1} = (\frac{1}{2}B_s^i, D_s^i, D_r^i)$ (augment non-uniformly spatial resolution at discontinuities $> T$ in S_i) **Set** $i = i + 1$, then **Solve** problem P_i to obtain solution S_i
2b	**If** $B_s^i > 1$ then **Repeat** step 2a

3	**Setup** next pyramid level
	if D_r^i is the Full Absolute Range then
	$P_{i+1} = (1, \frac{1}{2}D_s^i, S^i \pm T)$
	(augment non-uniformly spatial resolution
	at discontinuities $\leq T$ in sol S_i,
	augment disparity resolution uniformly,
	reduce relative disparity range)
	else
	$P_{i+1} = (1, \frac{1}{2}D_s^i, S^i \pm D_s^i)$
	(augment non-uniformly spatial resolution
	at discontinuities $\leq D_s^i$ in sol S_i,
	augment disparity resolution uniformly,
	reduce relative disparity range)
	Set $i = i + 1$, then **Solve**
	problem P_i to obtain solution S_i

3a	**Setup** next pyramid level $P_{i+1} = (1, D_s^i, D_r^i)$ (augment non-uniformly spatial resolution at discontinuities $> D_s^i$ in sol S_i, no change to disparity resolution and range) **Set** $i = i + 1$, then **Solve** problem P_i to obtain solution S_i
4	**If** $D_s^i > 1$ then **Repeat** steps 3 and 3a

Table 1: Non-uniform pyramid algorithm. The double line separates the absolute phase (steps 1,2,2a,2b) and the relative phase (steps 3,3a,4).

Absolute phase

The first phase (steps 1, 2, 2a, 2b) is labeled *absolute* because it searches for a match over the full disparity range at a fixed disparity step size. Designed to reduce the localization error of large depth discontinuities, it proceeds by gradually increasing the spatial resolution (step 2a), but only where large variations of disparity are observed. This is accomplished by selectively splitting cells that are significantly different in disparity from their neighbors.

After splitting, a new disparity map is recomputed using the same disparity range (D_r) and disparity step size (D_s). This process is repeated (step 2b) until the cells size can reach a single pixel in size ($B_s = 1$), thereby matching the original image resolution and allowing object contours to be accurate down to a single pixel.

A disparity difference is considered significant when it is larger than an user defined *threshold* T. If T is too small then it could have a significant impact on the memory requirement of the *absolute phase*, but it would not influence the quality of the final solution. If T is to big then serious artifacts could compromise the quality of the final solution. Nevertheless, the choice of an acceptable value is quite intuitive.

Relative phase

The second phase is designed to reduce the smooth surface errors and occasional new depth discontinuity errors that may appear. This phase does not search over the full disparity range. It rather uses a small range *relative* to the previous disparity so-

lution (step 3). This allows a dramatic reduction of the number of disparity steps, making possible the simultaneous increase of both spatial and disparity resolutions (thereby decreasing B_s and D_s) while keeping the memory requirement at an acceptable level. This *simultaneous* increase is required in order to remove smooth surface errors and explains why these surfaces are not considered during the *absolute* phase.

The use of a *relative* disparity interval introduces an important drawback. It makes it impossible for a disparity to change by a larger amount than the relative interval used. This is why it is so important to remove any large depth discontinuity error beforehand in the *absolute* phase.

It is possible that new large depth discontinuities will appear during the relative phase. These new discontinuities represent new object contours that may be badly localized and must be improved. The pixel blocks involved in the new contours will be split and the disparity map will be re-computed while keeping the same disparity range D_r and disparity step size D_s (step 3a).

At a given pyramid level during the *relative* phase a disparity interval twice as big as the previous disparity step is used. When the algorithm go from *absolute* to *relative* phase we exceptionally use a disparity interval D_r that is twice as big as the threshold value. Using a smaller interval could introduce serious artifacts by wrongly classifying pixels blocks.

An example

A synthetic stereo image pair ($512 \times 512 \times 60$ disparities), shown in Figure 4, illustrates how the algorithm is able to recover very good disparity maps with excellent object boundary localization. In this example, 96% of the computed disparities are within ± 1 pixel of the ground truth, with most of the errors in the occluded area on the left side of the sphere (errors are expected since we do not model occlusions). The bottom rightmost result of Figure 4 is from the MRRS method of Sun [5]. The result of a uniform pyramid scheme using the maximum-flow formulation is also illustrated to demonstrate the usefulness of non-uniform decomposition. With this scheme only 67% of the computed disparities are within ± 2 pixel of the ground truth. Like most uniform pyramid methods, both uniform approach feature large disparity errors at

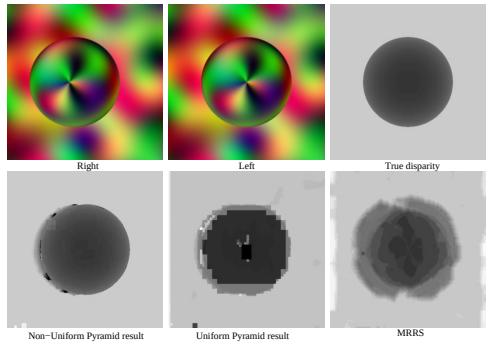

Figure 4: Sample reconstruction. At the top, a synthetic stereo pair with large discontinuity at object boundary and the true disparity map. At the bottom, the result of the non-uniform pyramid (left),the result of uniform pyramid (middle) ,and the result of MRRS classical pyramid (right).

the object boundaries since it propagates discontinuity errors from each pyramid level to the next.

The steps of the algorithm are illustrated in Figure 5 for a horizontal slice of the disparity map. The first six slices (Fig. 5 a-f) represent the absolute phase, where the spatial resolution is non-uniformly increased while the disparity resolution and range remain unchanged. The large discontinuities at the object boundary are gradually getting accurately localized, especially on the right side of the sphere. The left side is not as accurate as the right one since it is occluded and thereby impossible to match.

The last four steps (Fig. 5 g-j) illustrate the relative phase where disparity and spatial resolutions are increased while the disparity range is reduced. The gradual improvement of the smooth surface of the sphere is obvious. Notice how the absolute phase is only concerned with accurate localization of discontinuities and does not improve the solution along smooth surfaces. Inversely, the relative phase can only improve smooth surfaces since large discontinuities are outside of its *operational* range.

Limitations

This paper is aimed at showing the usefulness of non-uniform grid decomposition in the context of large and highly detailed disparity map computation. In this context, we isolate the effect of our framework by always using a single simple cost function that does not modeled occlusion. Never-

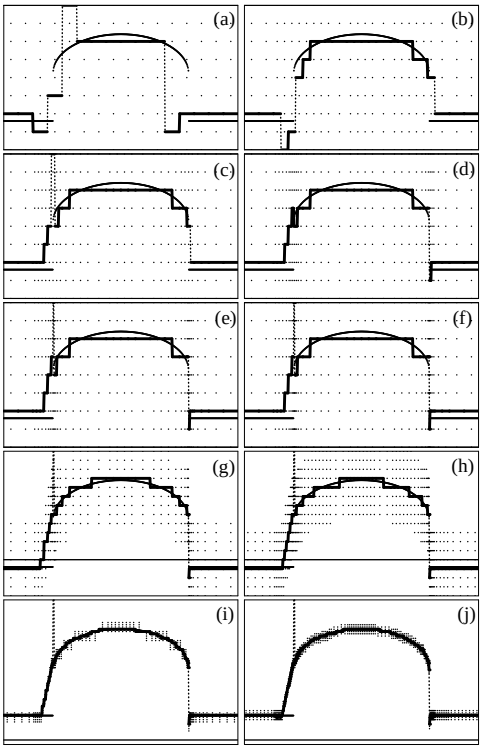

Figure 5: Algorithm steps for a horizontal slice of image, displayed with ground truth. Vertical axis is disparity, horizontal axis is x dimension of disparity map. The dots represent the disparity solutions under consideration.

theless, by post-processing it would be possible to extract an occlusion map directly from the disparity map.

Our framework has two potential weaknesses: First, if the initial pixel block size is too large, a block containing a large disparity in its middle may never be split during the absolute phase and thereby introduce an artifact. Second, if the *threshold* is too large, smooth surfaces will be difficult to recover. A very effective way[1] of reducing the impact of both limitations is to repeat step 3a before proceeding to step 4 in the algorithm of Table 1. At a given pyramid level with a maximum block size of $n \times n$ pixels, step 3a would be repeated at most $\log n$ times.

[1]We did not use such a strategy in the results presented in this paper.

4 Experimental Results

We tested our non-uniform pyramid stereo algorithm on a variety of stereo images. On synthetic images, such as the results presented in Figure 4 of Section 2, the algorithm achieved its goal of very good discontinuity localization and overall quality of the disparity map. The typical running time, on a 1.4 Ghz AMD Athlon with non optimized code, is about 10 seconds for a problem space of 17 million ($512 \times 512 \times 64$) possible matches. It requires 200 thousand elementary operations [2] to complete while the full size problem is estimated to require about 100 million elementary operations (the quarter size problem takes about 20 million operations). The saving of the pyramid approach is quite substantial. Uniform and non-uniform pyramids have similar running time, but the quality is much better in the non-uniform case. The experimental results on real imagery are presented in the next two sections. The cost function used in the following section is based on simple block matching and SSD.

Teapot

This data-set, courtesy of Jean-Yves Bouguet at Intel, features high resolution (2048x1536) images (see Figure 6) that contains some interesting features such as a slanted surfaces with few texture details, and a large disparity range of 400 pixels. The ground truth was provided in the form of a 3D model obtained with a structured light scanner. The *true* disparity map is computed from the calibration data and this 3D model. It itself contains some errors and provides depth only for the teapot itself and not the table, but still provides a good reference for comparison.

The full disparity map of the teapot is shown in Figure 6. The result of dynamic programming [28] is added for comparison. In general, dynamic programming is more sensitive to lack of texture and suffers from its inability of propagating smoothness across epipolar lines. The slanted table was well recovered by our method given its lack of texture. Most object boundaries are sharp and well localized. Right boundaries are more accurate than left boundaries because of occlusions, which are not currently detected.

[2]We count the number of *Discharge* operations in the preflow-push-relabel algorithm.

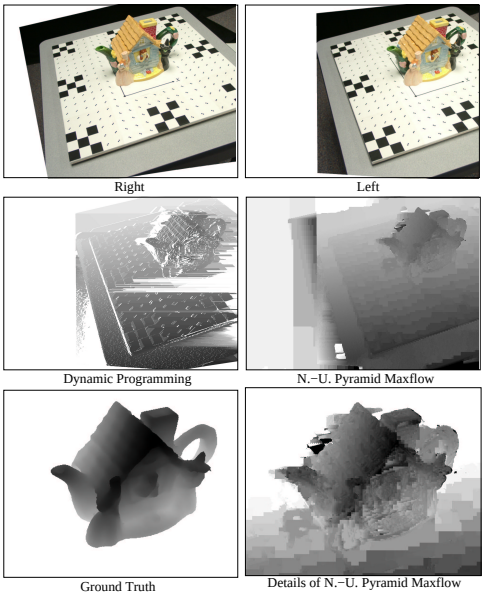

Figure 6: Teapot disparity map. Middle left, result obtained with dynamic programming [28] . Middle right, the non-uniform pyramid disparity map. The full disparity range is $[-600 \ldots -200]$ pixels. Bottom left, details of the true disparity map obtained from a scanned 3D model of the teapot. Bottom right, details of the non-uniform pyramid disparity map. The displayed disparity range is $[-460 \ldots -350]$ pixels.

Baseball

The baseball stereo pair, courtesy of Bill Hoff at the University of Illinois, is shown in Figure 7. It features very highly textured surfaces, exposure variations between the two images, and very sharp object contours. The top left disparity map is the result of a fast pyramid stereo algorithm by C. Sun [5]. While it runs very fast, this algorithm does not recover very sharp contour since it does not explicitly model them. The bottom results are obtained respectively with the full size maximum-flow algorithm from Roy [19], the non-uniform maximum-flow pyramid and the non-uniform maximum-flow pyramid working in the sub-pixel domain. Both results from pyramidal maximum-flow are comparable, if not better, than the full size maximum flow result. Also, as expected, the running time is about 30 times faster with the pyramid.

5 Conclusion

This paper presented a new non-uniform approach to hierarchical stereo matching, aimed toward efficiently matching large images. One objective of the method is that it detects and accurately localizes large depth discontinuities typical of object boundaries, which is usually hard to accomplish using a classical pyramid approach.

The algorithm reduces the spatial resolution of the disparity map non-uniformly so different levels of coarseness can be present at the same time, thereby drastically improving the results while limiting the memory requirements. It uses a graph formulation to represent the problem space, thus enabling the use of the maximum-flow algorithm. Compared with the non-pyramid Maximum-Flow approach, our results show good speed improvements and the ability of our method to tackle much larger problems. It solve efficiently and globally the matching problem at various hierarchical levels with arbitrarily complex neighborhood structures independent of the epipolar constraint. Moreover, it provides good stability when intensity variation are present in the stereo pair.

As for future research, are goal is to obtain a pyramid scheme that can be proved to never propagate errors between successive levels, which corresponds to obtaining the solution of the original non-pyramid problem.

Acknowledgments

The authors express their gratitude to Jean-Yves Bouguet (Intel) for providing images and reconstruction results and to C. Sun (CSIRO, Mathematical and information science, Australia) for providing an online version of his pyramid stereo code. Also, our gratitude goes to Max Mignote and Jean Meunier (Université de Montréal) for their comments and suggestions. This work was made possible by NSERC (Canada) and NATEQ (Québec) governmental grants.

References

[1] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV IJCV 47(1/2/3):7-42, April-June 2002.*, 47, 2002.

[2] L. H. Quam. Hierarchical warp stereo. In *In Image Understanding Workshop*, pages 149–155, New Orleans, Lousiana, December 1984.

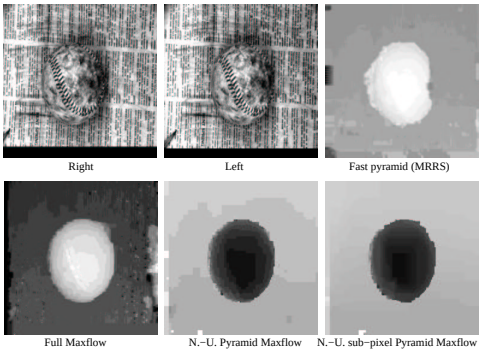

Figure 7: Baseball results. Top right, MRRS Stereo result. Bottom left, Maximum-flow full size solution ($256 \times 256 \times 18$ nodes resulting in 9.4 million discharge operation) Bottom middle, Non-uniform Maximum-flow pyramid (total of 0.57 million discharge operation) Bottom right, Non-uniform Maximum-flow with disparity of $\frac{1}{4}$ pixel (total of 2.37 million discharge operation).

[3] A. Witkin, D. Terzopoulos, and M. Kass. Signal matching through scale space. *International Journal of Computer Vision*, 1:133–144, 1987.

[4] J. Magarey and A. Dick. Multiresolution stereo image matching using complex wavelets. In *Proc. 14th Int. Conf. on Pattern Recognition*, volume I, pages 4–7, August 1998.

[5] C. Sun. Multi-resolution stereo matching using maximum-surface techniques. In *Digital Image Computing: Techniques and Applications*, pages 195–200, Pert, Australia, 1999.

[6] M. Berthod, L. Gabet, G. Giraudon, and J.L. Lotti. High-resolution stereo for the detection of buildings. In *Ascona95*, pages 135–144, 1995.

[7] H. Pan and J. Magarey. Multiresolution phase-based bidirectional stereo matching with provision for discontinuity and occlusion. *Digital Signal Processing—A Review Journal*, 1999.

[8] G.-Q. Wei, W. Brauer, and G. Hirzinger. Intensity- and gradient-based stereo matching using hierarchical gaussian basis functions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(11):1143–1160, 1998.

[9] A. Koschan and V. Rodehorst. Dense depth maps by active color illumination and image pyramids. In F. Solina, W.G. Kropatsch, R. Klette, and R. Bajcsy, editors, *Advances in Computer Vision*, pages 137–148, Springer, Vienna, Austria, 1997.

[10] H. Schultz. Terrain reconstruction from oblique views. In *ARPA Image Understanding Workshop*, pages II:1001–1008, Monterey, CA, november 1994.

[11] Y.P. Hung, C.S. Chen, K.C. Hung, Y.S. Chen, and C.S. Fuh. Multipass hierarchical stereo matching for generation of digital terrain models from aerial images. *MVA*, 10(5-6):280–291, April 1998.

[12] J. Lotti and Gerard Giraudon. Adaptive Window Algorithm for Aerial Image Stereo. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 701–703, Jerusalem, October 1994.

[13] J. Lotti and G. Giraudon. Correlation algorithm with adaptive window for aerial image in stereo vision. In *Image and Signal Processing for Remote Sensing*, Rome, Italy, 1994.

[14] Jong-Il Park and Seiki Inoue. Hierarchical depth mapping from multiple cameras. In *Int. Conf. on Image Analysis and Processing*, volume 1, pages 685–692, Florence, Italy, 1997.

[15] Richard Szeliski and Heung-Yeung Shum. Motion estimation with quadtree splines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12):1199–1210, 1996.

[16] L. Falkenhagen. Hierarchical block-based disparity estimation considering neighbourhood constraints. In *International workshop on SNHC and 3D Imaging*, Rhodes, Greece, September 1997.

[17] M. Fradkin, M. Roux, H. Maitre, and U. M. Leloglu. Surface reconstruction from multiple aerial images in dense urban areas. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 262–267, Fort Collins, Colorado, USA, June 1999.

[18] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. Dense disparity map estimation respecting image discontinuities: a pde and scalespace based approach. Technical Report RR-3874, INRIA, 2000.

[19] S. Roy and I. J. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *Proc. Int. Conference on Computer Vision*, pages 492–499, Bombay, India, 1998.

[20] A. Mancini and J. Konrad. Robust quadtree-based disparity estimation for the reconstruction of intermediate stereoscopic images. In *SPIE Stereoscopic Displays and Virtual Reality Systems*, volume 3295, pages 53–64, January 1998.

[21] S. Sethuraman, M. Siegel, and A. Jordan. A multiresolutional region based segmentation scheme for stereoscopic image compression. In *IS+T/SPIE's Symposium*, volume 2419, pages 265–75, San Jose,US, 1995.

[22] S. Sethuraman, M. Siegel, and A. Jordan. Segmentation based coding of stereoscopic image sequences. In *SPIE/IS+T*, Bellingham WA, 1996.

[23] A. Koschan, V. Rodehorst, and K. Spiller. Color stereo vision using hierarchical block matching and active color illumination. In *Proc. of Int. Conf. on Pattern Recognition*, volume 1, pages 835–839, Vienna, Austria, Vol. I, pp. 835-839, 1996., 1996.

[24] R. Szeliski. Stereo algorithms and representations for image-based rendering. In *British Machine Vision Conference*, pages 314–328. Springer-Verlag, 1999.

[25] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(9):920–932, 1994.

[26] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline using dynamic programming. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 7(2):139–154, 1985.

[27] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, june 1998.

[28] I. J. Cox, S. Hingorani, B. M. Maggs, and S. B. Rao. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996.