Non-Uniform Hierarchical Geo-consistency for Multi-baseline Stereo

Marc-Antoine Drouin Martin Trudeau Sébastien Roy
DIRO, Université de Montréal
{drouim,trudeaum,roys}@iro.umontreal.ca

Abstract

We propose a new and flexible hierarchical multi-baseline stereo algorithm that features a non-uniform spatial decomposition of the disparity map. The visibility computation and refinement of the disparity map are integrated into a single iterative framework that does not add extra constraints to the cost function. This makes it possible to use a standard efficient stereo matcher during each iteration. The level of refinement is increased automatically where it is needed in order to preserve a good localization of boundaries. While two graph-theoretic stereo matchers are used in our experiments, our framework is general enough to be applied to many others. The validity of our framework is demonstrated using real imagery with ground truth.

1. Introduction

The goal of binocular stereo is to reconstruct the 3D structure of a scene from two views. Occlusion occurs when part of a scene is visible in one camera but not the other (see figure 1). In this paper, we use a cross-shaped configuration with 4 cameras equidistant to the center one which is the reference. The difficulty of detecting occlusion comes from the fact that it is induced by the 3D structure of the scene, which is unknown until the correspondence is established, as it is the final goal of the algorithm.

In this paper, we propose a new and flexible hierarchical stereo algorithm that features a non-uniform spatial resolution. The visibility computation and refinement of the disparity map are integrated into a single iterative framework that does not add extra constraints to the cost function. The level of refinement is increased automatically where it is needed in order to preserve a good localization of boundaries. Our algorithm uses the occlusion model proposed by [7]. This model relies on geometric inconsistencies in the disparity map to detect occlusions. As will be shown, our hierarchical algorithm provides major speedups over the non-hierarchical one of [7] while preserving the quality of the final solution. In this paper, we use graph-cut[4] and maximum flow formulation with a linear smoothing term[37]. As in [7], our framework is general enough to be used with other stereo algorithms. A survey paper by Scharstein and Szeliski [41, 40] compares various standard algorithms.

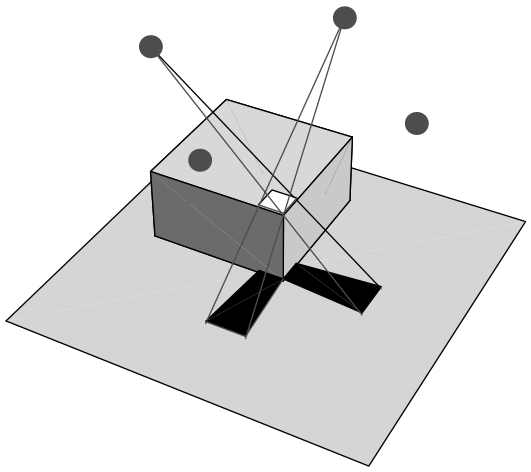

Figure 1. Example of occlusion. Occluded pixels appear in black, occluders in white.

The rest of this paper is divided as follows: in Section 2, previous work will be presented. Section 3 describes the visibility modeling framework. Our proposed framework is described in Section 4. Experimental results are presented in Section 5.

2. Previous Works

In Egnal [9], five basic strategies to overcome occlusion for two cameras are presented: left-right checking, bimodality test, goodness Jumps constraint, duality of disparity discontinuity and occlusion, and uniqueness constraint. Some algorithms rely on one or more of these strategies, and are often based on varying a correlation window posi-

tion or size [20, 14, 49, 21]. Other algorithms use dynamic programming [32, 18, 5] or graph-cut [19, 23]. In [46], visibility and disparity are iteratively minimized using belief propagation.

Many algorithms are specially designed to cope with occlusion in multi-baseline stereo. They can be coarsely divided into three categories [8]. Some approaches are based on visibility heuristics [21, 31, 39, 35, 15, 44, 8]. Others guarantee a solution that is geo-consistent [7, 26, 43, 11, 24]. These approaches preserve the consistency between the recovered visibility and the geometry [7]. Our algorithm belongs to this category. Finally, some algorithms are mixes between heuristic and geo-consistent algorithms [6, 53, 16].

Many hierarchical approaches have been introduced to speed up computation of stereo matching [30, 33, 36, 54, 30, 45, 2, 33, 52, 25, 42, 17, 28, 34]: 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. These errors appear mostly near object boundaries. Many approaches have been proposed to cope with errors induced by pyramids [17, 10, 13, 13, 38, 28, 29, 17]. Some methods extend the search interval where large disparity variations are present. Some of them use feature extraction to improve border localization, whilst others used a non-uniform decomposition of the disparity map. Some approaches use multi-resolution techniques to speed up the convergence of an energy function minimization [1, 12].

3. Visibility framework

In this section, we review the visibility framework that will be used and that comes from [7]. When doing stereo matching, there is a set $\mathcal{P}$ of reference pixels, for which we want to compute disparity, and a set $\mathcal{D}$ of disparity labels. A $\mathcal{D}$-configuration $f : \mathcal{P} \mapsto \mathcal{D}$ associates a disparity label to every pixel. In order to simplify the discussion, we will always consider the disparity as a positive value independently of the supporting camera used. We also assume that the supporting images are at an equal distance from the reference. To model occlusion, we must compute the volumetric visibility $V_i(\mathbf{p}, d, f)$ of a reference pixel $\mathbf{p}$ located at disparity d from the point of view of a camera i, given a disparity configuration f defined for all other pixels. It is set to 1 if the point is visible and 0 otherwise. The visibility information is collected into a vector, the *visibility mask*

$$V(\mathbf{p}, d, f) = (V_1(\mathbf{p}, d, f), \dots, V_N(\mathbf{p}, d, f))$$

where N is the number of cameras outside the reference. We call $\mathcal{M}$ the set of all possible visibility masks; an $\mathcal{M}$-configuration $g : \mathcal{P} \mapsto \mathcal{M}$ associates a mask to every pixel. Let us define the special configuration g^0 with $g^0(\mathbf{p}) = (1, \dots, 1)$ for all $\mathbf{p} \in \mathcal{P}$; this corresponds to the case where all cameras are visible by all points. The problem is the minimization in f and g of

$$E(f,g) = \underbrace{\sum_{\mathbf{p}\in\mathcal{P}} e(\mathbf{p}, f(\mathbf{p}), g(\mathbf{p}))}_{\substack{\textit{pixel} \\ \textit{likelihood}}} + \underbrace{\sum_{\mathbf{p}\in\mathcal{P}} \sum_{\mathbf{r}\in\mathcal{N}_\mathbf{p}} s(\mathbf{p}, \mathbf{r}, f(\mathbf{p}), f(\mathbf{r}))}_{\textit{pixel smoothing}} \tag{1}$$

with the constraint

$$g(\mathbf{p}) \leq V(\mathbf{p}, f(\mathbf{p}), f) \tag{2}$$

for each component of these vectors and all $\mathbf{p} \in \mathcal{P}$. The constraint of Eq. 2 is named *geo-consistency* and the inequality allows the mask to contain a subset of the visible cameras[7]. The removal of some extra cameras has been observed to have little impact on the quality of the solution and is used in many multi-baseline stereo algorithms[31, 7, 6, 8, 21, 31, 39, 35, 15]. The pixel likelihood term is defined as

$$e(\mathbf{p}, d, \mathbf{m}) = \frac{\mathbf{m} \cdot C(\mathbf{p}, d)}{|\mathbf{m}|} \quad \text{for } \mathbf{p} \in \mathcal{P}, d \in \mathcal{D}, \mathbf{m} \in \mathcal{M}$$

where $C(\mathbf{p}, d) = (C_1(\mathbf{p}, d), \dots, C_N(\mathbf{p}, d))$ is the vector of matching costs of the pixel $\mathbf{p}$ at disparity d for each camera. We use $|\mathbf{m}|$ to represent the l_1-norm which is just the number of cameras used for pixel $\mathbf{p}$ at disparity d.

In [7], it was proposed to reduce the dependency between f and g by making it *temporal*: we let f^0 be the $\mathcal{D}$-configuration minimizing $E(f^0, g^0)$ in f and for $t > 0$, let iteratively f^t be the function minimizing $E(f^t, g^t)$ with g^t defined as

$$g^t(\mathbf{p}) = H(\mathbf{p}, f^t(\mathbf{p}), t-1) \tag{3}$$

and

$$H_i(\mathbf{p}, d, t) = \prod_{0\leq k\leq t} V_i'(\mathbf{p}, d, f^k) \tag{4}$$

where H is a *visibility history mask* and V' is the pseudo-visibility described below. Because of the way g^t is defined, cameras that are removed at one iteration cannot be kept at the next. This greedy approach guarantee convergence (or stabilization) in a polynomial number of steps. The case where $|g^t(\mathbf{p}, d)| = 0$ is discussed in [7].

In [7] a significant bias was measured in the localization of depth discontinuities. Front objects are enlarged and this discourages the direct use of visibility to update the *visibility history mask*. Instead, a pseudo-visibility

$$V'(\mathbf{p}, d, f) = (V_1'(\mathbf{p}, d, f), \dots, V_N'(\mathbf{p}, d, f))$$

was introduced, which compensates for the bias by labeling both occluders and occludees as invisible. Discussion about

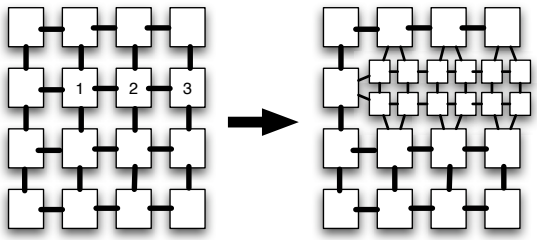

Figure 2. The division of blocks. Blocks 1 and 3 are labeled as occluder and occludee respectively. Assuming horizontal epipolar lines, the node 2 must also be split.

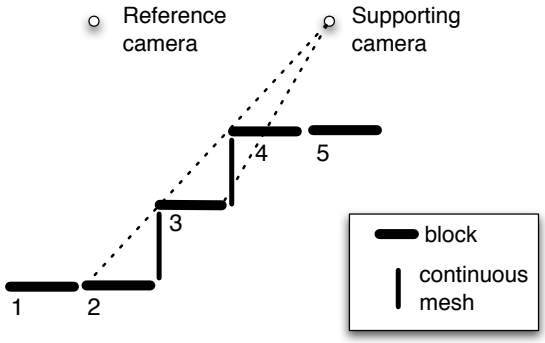

Figure 3. Nodes 2 and 3 are occludees and node 4 is an occluder.

the computation of the pseudo-visibility is postponed until section 4.1. In this framework, the disparity map is represented as a continuous mesh, this guarantees the preservation of the ordering constraint between the reference and any supporting camera.

In the next section, we will present the integration of our non-uniform hierarchical decomposition of the disparity map with the iterative visibility framework presented in this section.

4. Our framework

We propose the use of a non-uniform iterative decomposition of the disparity map. The energy minimization does not associate a disparity to every pixel, but it associates a single disparity to all the pixels included in a block. Each pixel has a visibility mask and it is thus possible to have two pixels in the same block having different masks. At each iteration t, the disparity map is represented as a graph $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$ where $\mathcal{V}^t$ is the set of nodes representing the blocks of pixels and $\mathcal{E}^t$ is the set of edges. An edge between two nodes indicates that smoothing is applied between them (see Fig. 2). The $\mathcal{D}$-configuration $f^t : \mathcal{V}^t \mapsto \mathcal{D}$ associates a disparity label to every block in $\mathcal{V}^t$. The problem is the minimization in f^t of

$$E_{\mathcal{G}^t}(f^t, g^t) = \sum_{\mathbf{p} \in \mathcal{V}^t} e_{\mathcal{G}^t}(\mathbf{p}, f^t(\mathbf{p}), g^t(\mathbf{p})) + \begin{array}{l} \textit{inter-block} \\ \textit{smoothing} \end{array} \tag{5}$$

where g^t is the $\mathcal{M}$-configuration as defined in Eq. 3. The block likelihood term $e_{\mathcal{G}^t}$ is simply the sum of the likelihoods of pixels included in the block. The inter-block smoothing is simply defined as the sum of the pixel smoothing costs.

In this paper, we used as the initial graph $\mathcal{G}^0$ a regular grid with a user-defined block size. The initial graph could also be obtained from a segmentation of the reference image [27]. Discussion about the block size of the initial graph is postponed until section 5.3. The initial masks $g^0(\mathbf{p})$ have all cameras visible for all $\mathbf{p}$. Once f^t is found, g^{t+1} and $\mathcal{G}^{t+1}$ can be computed using rendering techniques that will be described in section 4.1. The pseudo-code of our algorithm is shown in Figure 4. Since the blocks that are split at one iteration cannot be merged at a later one, the convergence (or stabilization) is guaranteed. This is achieved in a polynomial number of steps. Indeed, $|\mathcal{V}^t|$ is monotonically increasing with t and is at most $|\mathcal{P}|$ and $H(\mathbf{p}, d, t)$ is monotonically decreasing in t for all $\mathbf{p}$ and d. Moreover, if $\mathcal{G}^t = \mathcal{G}^{t+1}$ and $H(\mathbf{p}, d, t-1) = H(\mathbf{p}, d, t)$ for all $\mathbf{p}$ and d, then $f^t = f^{t+1}$ since both are solutions to the same minimization problem, and the process has stabilized. The algorithm converges (or stabilizes) to a geo-consistent solution, but can go through intermediate ones that are not.

NON-UNIFORM HIERARCHICAL GEO()
1 Build $\mathcal{G}^0$
2 $g^0(\mathbf{p}) \leftarrow (1, ..., 1) \quad \forall \mathbf{p} \in \mathcal{P}$
3 $t \leftarrow 0$
4 $f_{\mathcal{G}^t} \leftarrow \arg\min_f E_{\mathcal{G}^t}(f, g^t)$
5 Render $f_{\mathcal{G}^t}$ in all supporting cameras
6 Compute g^{t+1} and $\mathcal{G}^{t+1}$ using render buffers of previous line
7 **if** $g^t = g^{t+1}$and $\mathcal{G}^t = \mathcal{G}^{t+1}$
8 **then return** $f_{\mathcal{G}^t}$
9 $t \leftarrow t + 1$
10 **goto** 4

Figure 4. Algorithm overview

4.1. Pseudo-visibility and block division

A block is called occluder when one of its pixels occludes a pixel from anther blocks (called occludee). When the blocks contain many pixels, the discontinuity between

an occluder and an occludee is probably poorly located. Both blocks must be split in order to improve the localization of the depth discontinuity at the next iteration. The labeling of blocks as occludee and occluder is done concurrently with the pseudo-visibility computation. The pseudo-visibility masks V_i' are computed by using rendering techniques. Two renderings of the current disparity map f are done from the point of view of each supporting camera i: one with an regular Z-buffer and one with a reverse Z-buffer test. Two disparity maps S_i^f and L_i^f are thus obtained and they contain the minimal and maximal pixelwise disparity observed by the supporting camera i. In [7], the pseudo-visibility function $V_i'(\mathbf{p}, d, f)$ is computed as

$$V_i'(\mathbf{p}, d, f) = \delta\left(S_i^f(T_i(\mathbf{p}, d)) - L_i^f(T_i(\mathbf{p}, d))\right)$$

where δ is 1 at 0 and 0 elsewhere and $T_i(\mathbf{p}, d)$ is the projection pixel $\mathbf{p}$ at disparity d in the supporting camera i. This approach might not detect all the occluded pixels. As an example, using this rendering technique, blocks 2 and 4 of figure 3 are correctly labeled as occludee and occluder. Nevertheless, this technique does not detect the occluded block 3. We propose an approach that identifies all the occluded blocks. The color channels are used to encode the index of each node in $\mathcal{G}^t$. Since a *visibility mask* is assigned to every pixel, the index of a pixel in the block is also encoded using one of the color channel. When $S_i^t(\mathbf{r})$ and $L_i^t(\mathbf{r})$ are different, we can used the color buffer to identify the occluder and occluded blocks having the smallest disparities. Since a continuous mesh representation is used, all blocks (and pixels) between those two along the epipolar line must be boths occluder and occludee; the blocks must be split and the visibility masks associated to their pixels must be updated (see Fig. 2). Note that when using our camera configuration, this rendering process can be sped up by replacing it by a line drawing using depth buffers.

4.2. Stereo matcher

At each iteration of our algorithm, a disparity map is computed. Many algorithms are capable of computing non-uniform disparity maps, for instance belief propagation [47], reweighted message passing [22], dynamic programming on tree[50, 27], graph-cut [4] and maximum flow formulation with a linear smoothing term [37].

When belief propagation is used, the local evidences can be initialized using the values of the previous iteration. In this case, our algorithm is an occlusion modeling extension of the hierarchical belief propagation presented in [12] where the uniform decomposition is replaced by our non-uniform one. Note that bipartite scheduling is no longer possible, but the distance transform can still be used.

Note that search space reduction techniques that reduce the number of disparity labels that must be examined could also be used [51, 38].

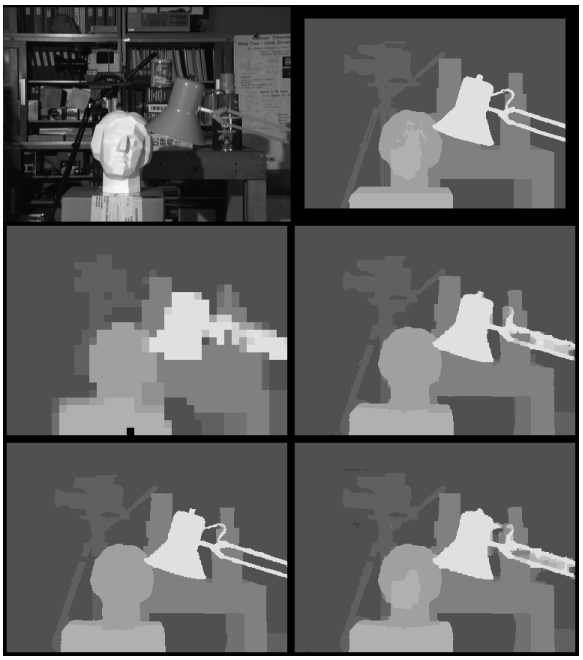

Figure 5. Reference images for the Head and Lamp (top left) and ground truth (top right). Disparity map obtained from NU-GEO-MF after first (middle left) and last iterations (middle right). The result of applying border-cut to NU-GEO-MF (bottom left) and the result of GEO-MF (bottom right) are shown as well.

5. Experimental results

In all our experiments, the matching cost function was the same for all algorithms, that of [24] which is based on [3]. We used color images but only gray scale ones are shown here. As for the pixel smoothing term, we used the experimentally defined smoothing function that also comes from [24]:

$$s(\mathbf{p}, \mathbf{r}, f(\mathbf{p}), f(\mathbf{r})) = \lambda\, h(\mathbf{p}, \mathbf{r})\, l(f(\mathbf{p}), f(\mathbf{r}))$$

where h is defined as

$$h(\mathbf{p}, \mathbf{r}) = \begin{cases} 3 & \textit{if } |I_{\textit{ref}}(\mathbf{p}) - I_{\textit{ref}}(\mathbf{r})| < 5 \\ 1 & \textit{otherwise} \end{cases}$$

with $l(d, d') = |d - d'|$ for the maximum flow with a linear smoothing term (MF) [37] and $l(d, d') = \delta(d - d')$ for graph-cut (BNV)[4]. A pixel disparity is considered erroneous if it differs by more than one disparity step from the ground truth. This error measurement is used by two comparative studies for 2-camera stereo [48, 41]. Note that a

Algorithm	Error (whole image)	Error (mask)
KZ1	2.3	-
KZ1'	1.3	-
ASYM-KZ1	1.3	-
REL-DP	1.9	-
NAKA-BNV	1.7	-
HYBRID-IDP	1.7	-
GEO-BNV pt	2.2	1.5
GEO-BNV	2.5	1.6
GEO-MF	2.9	2.0
BC (average)	1.1	-
NU-GEO-MF	2.4	1.5
NU-GEO-BNV	2.5	1.7

Table 1. Percentages of error of the different algorithms for Head and Lamp scene, using 5 images.

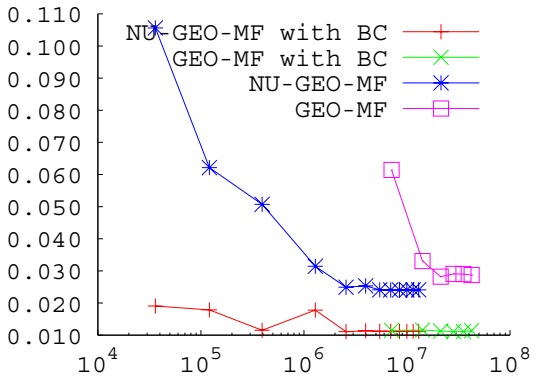

Figure 6. Variation of the error rate as a function of the number of elementary operations.

Algorithm	Smoothing parameter							
	1	2	4	8	16	32	64	128
NU-GEO-MF	2.61	2.51	2.51	2.51	2.71	3.04	3.27	5.90

Table 2. Resistance to change of the smoothing parameter for the Head and Lamp scene. The parameter increases by a factor of 128, while the error rate varies by less than 3.29%.

better smoothing term and cost function are now available [55]. We did not used them to simplify comparisons.

As in [7], we keep a single *visibility history mask* for each pixel $\mathbf{p}$ regardless of the disparity d. The Eq. 4 becomes

$$H_i(\mathbf{p}, t) = \prod_{0 \leq k \leq t} V_i'(\mathbf{p}, f^k(\mathbf{p}), f^k).$$

This saves memory but the convergence is no longer guaranteed. We simply stop iterating when $\mathcal{G}^t = \mathcal{G}^{t+1}$ and $H(\mathbf{p}, t) = H(\mathbf{p}, t-1)$ for all $p \in \mathcal{P}$.

Recently, a new type of post-processing algorithms named border-cut (BC) was proposed[8]. Rather than associating a disparity label to every pixel, it associates a position to every disparity discontinuity. This method obtains sharp and well-located disparity discontinuities starting from the output of a wide range of stereo matchers. We provide results with and without this post-processing step. In our experiments, we used the dataset from the Multiview Image Database from the University of Tsukuba.

5.1. Head and Lamp scene

This dataset is composed of a 5×5 image grid. Each image has a resolution of 384×288 (see Fig. 5). The search interval was between 0 and 15 pixels and we used 16 disparity steps. Some disparity maps are shown in Fig. 5 and error percentages are given in Table 1. Since we use a visibility framework that preserved the ordering constraint, we also computed the error after removing the pixels breaking the ordering constraint, in particular part of the arm of the lamp. We use the same mask as in [7], the one determined by re-projecting the ground truth in each supporting camera. We provide results for our non-uniform framework using the stereo matcher BNV and MF. We label them NU-GEO-BNV and NU-GEO-MF. The non-hierarchical versions are labeled GEO-BNV and GEO-MF and come from [7]. The entries ASYM-KZ1 and KZ1' come directly from [53]. KZ1 and REL-DP come from [24] and [16] respectively. HYBRID-IDP and NAKA-BNV come from [6]. The error rate associated with border-cut (BC) comes from [8] and is the average of the error rates obtained using different initializations. GEO-BNV and GEO-BNV pt come from [6]. We also compared with GEO-MF and our implementation of GEO-MF, that achieved a lower error rate than presented in [7]. The error rate of NU-GEO-MF is 2.4% while GEO-MF obtained a higher error rate of 2.9%. The number of iterations is respectively 12 and 6 for NU-GEO-MF and GEO-MF. The total number of elementary operations is 1.5 million for NU-GEO-MF and 7.1 million for GEO-MF.[1] The running time for NU-GEO-MF and GEO-MF are 55 and 161 seconds respectively using an AMD Athlon(tm) 64 Processor 3500+. Figure 6 shows the number of elementary operations giving the error rate for NU-GEO-MF and GEO-MF. After 7 iterations, NU-GEO-MF achieved a lower error rate than GEO-MF at a fraction of the cost of computing a single iteration of GEO-MF. The border-cut algorithm was run using the disparity maps obtained after each iteration of GEO-MF and NU-GEO-MF. Using both initialization, the error rates after a few iterations are similar to those obtained in [8]. The parameters for the border-cut are those used in [8].

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

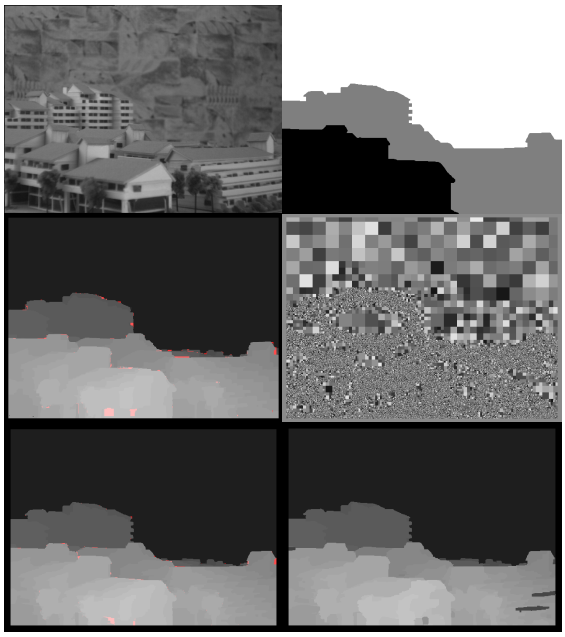

Figure 7. Reference image for the City scene (top left). The *partial* discontinuity ground truth (top right). Disparity map obtained from NU-GEO-MF (middle left) and improved by border-cut (bottom left). Final graph for NU-GEO-MF (middle right). Disparity map obtained by border-cut starting from the result of Hybrid-IDP (bottom right).

The initial block size used in our experiments is 10×10 and the initial disparity map is shown in figure 5. For NU-GEO-MF and NU-GEO-BNV the non-uniform decomposition allow a reduction of approximatively 80% of the problem space. Table 2 shows the stability to change of the smoothing parameter of NU-GEO-MF, giving the error percentage for 8 values of this parameter.

5.2. City scene

This dataset contains 81 640×480 images in a 9×9 grid (Fig. 7). We only used 5 images in a cross configuration. Each disparity map was computed using 44 disparity steps and the search interval was between 0 and 43 pixels. The initial block size was 30×30. A *partial* discontinuity ground truth that comes from [8] is shown in Fig. 7. A discontinuity location is considered erroneous if it differs by more than one pixel from the ground truth. The results are presented in Table 3 and some disparity maps are shown in Fig. 7. We also show the result obtained using Hybrid-IDP [6] with the border-cut post-processing. The latter algorithm is fast, however it introduces artifacts that are not eliminated by border-cut. The disparity maps obtained by NU-GEO-MF before and after border-cut are

Algorithm	Before B-C (best γ)	After B-C (fixed γ)
NU-GEO-MF	18.8	11.8
GEO-MF	23.2	10.4
GEO-BNV	15.4	11.4
NU-GEO-BNV	15.6	12.6
Hybrid-IDP	28.2	14.7

Table 3. Percentage of error in the discontinuity location, according to the *partial* ground truth, of the different algorithms for City scene, before and after border-cut.

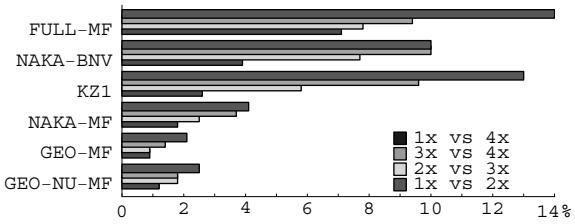

Figure 8. Resistance to baseline change for 6 algorithms for the Santa scene; each bar represents a percentage of incompatible pixels between depth maps obtained for two different baselines.

shown in Fig. 7. The percentage of pixels with a difference in disparity greater than one for GEO-MF and NU-GEO-MF is only 0.5%. After applying border-cut on each disparity map, this difference drops to .2%. These different pixels are highlighted in Fig. 7 by saturating the red channel. Although the disparity maps obtained are almost identical, the reduction in graph sizes is significant. The final non-uniform decomposition of the disparity map is shown in Fig. 7. The reduction of the search space is approximatively 75%.

5.3. Santa scene

This dataset contains 81 images in a 9×9 grid (see figure 10). We only used 5 images in a cross-shaped configuration. As the baseline increases, the amount of occlusion in the scene increases as well. To measure the level of resistance to change of the baseline, we used the incompatibility metric introduced in [7]. Figure 8 contains bar charts of the percentages of pixels incompatible between the depth maps obtained for two baselines. Images were reduced by a factor of 2 to achieve a resolution of 320×240 and 23 disparity steps were used. In addition to GEO-MF and NU-GEO-MF, results from 4 algorithms coming from [7] were included. Our hierarchical approach is slightly less resistant to changes of baseline. The reduction in search space for baselines 1x to 4x is 65%, 62%, 29% and 30% respec-

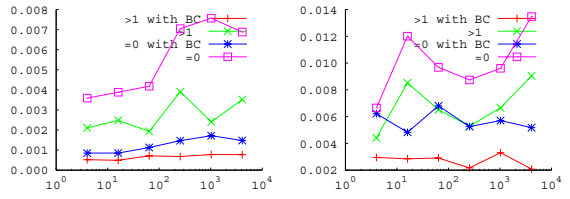

Figure 9. Ratio of incompatible pixels between GEO-MF and NU-GEO-MF as a function of the initial block size for the city (left) and the Santa scenes (right). The block size varies from 4 to 4096 pixels.

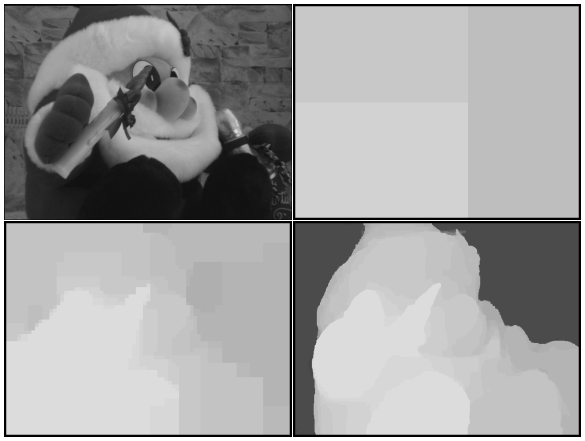

Figure 10. Reference image for the Santa scene and disparity maps after different iteration of NU-GEO-MF.

tively. The small reduction for the largest baselines is explained by the geometric inconsistencies introduced by by non-lambertian surfaces. Indeed, these surfaces move significantly when the baseline is large. Figure 10 shows disparity maps obtained using initial blocks of 130×130 pixels on the full size image using 45 disparity steps. The incompatibility between GEO-MF and NU-GEO-MF for different initial block sizes is shown in figure 8 for both the city and Santa scenes. A pixel is considered incompatible when its disparities are different ($= 0$) or differ by more than one (> 1).

5.4. Conclusion

We proposed a new and flexible hierarchical stereo algorithm that features a non-uniform spatial resolution. The visibility computation and refinement of the disparity map are integrated into a single iterative framework. The disparity map is represented as a graph and the nodes are split using the visibility information. The levels of refinement is increased automatically where they are needed most to preserve accurate localization of object boundaries. Our framework does not add extra constraints to the cost function and is very indifferent to the choice of the initial block size. As has been shown, our algorithm allows major speedups and preserves the quality of the final solution. While two graph-theoretic stereo matchers are used in our experiments, our framework is general enough to be applied to many others.

As for future work, the extension of our approach to full volumetric reconstruction, where occlusion becomes the dominant problem, should be investigated.

References

[1] 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.

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

[3] S. Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(4):401–406, 1998.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cut. In *Int. Conf. on Comput. Vision*, pages 377–384, 1999.

[5] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs. A maximum likelihood stereo algorithm. *Comput. Vis. Image Underst.*, 63(3):542–567, 1996.

[6] M.-A. Drouin, M. Trudeau, and S. Roy. Fast multiple-baseline stereo with occlusion. In *3-D Digit. Imag. and Model.*, pages 540–548, June 2005.

[7] M.-A. Drouin, M. Trudeau, and S. Roy. Geo-consistency for wide multi-camera stereo. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 351–359, June 2005.

[8] M.-A. Drouin, M. trudeau, and S. Roy. Improving border localization of multi-baseline stereo using border-cut. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 511–518, 2006.

[9] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 24(8):1127–1133, 2002.

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

[11] O. D. Faugeras and R. Keriven. Complete dense stereovision using level set methods. In *Europ. Conf. on Comput. Vision*, pages 379–393, 1998.

[12] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70(1), 2006.

[13] M. Fradkin, M. Roux, H. Maitre, and U. M. Leloglu. Surface reconstruction from multiple aerial images in dense urban areas. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, volume 1, pages 262–267, June 1999.

[14] A. Fusiello, V. Roberto, and E. Trucco. Efficient stereo with multiple windowing. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, 1997.

[15] M. Goesele, S. M. Seitz, and B. Curless. Multi-view stereo revisited. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, 2006.

[16] M. Gong and Y.-H. Yang. Fast stereo matching using reliability-based dynamic programming and consistency constraint. In *Int. Conf. on Comput. Vision*, 2003.

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

[18] S. Intille and A. F. Bobick. Disparity-space images and large occlusion stereo. In *Europ. Conf. on Comput. Vision*, pages 179–186, 2002.

[19] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *Fifth European Conference on Computer Vision*, pages 232–248, 1998.

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

[21] S. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multiview stereo. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, 2001.

[22] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. Technical Report MSR-TR-2005-38, Microsoft Research, 2005.

[23] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions via graph cuts. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 508–515, 2001.

[24] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Europ. Conf. on Comput. Vision*, 2002.

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

[26] K. Kutulakos and S. Seitz. A theory of shape by space carving. *Int. J. Comput. Vision*, 38(3):133–144, 2000.

[27] C. Lei, J. Selzer, and Y.-H. Yang. Region-tree based stereo using dynamic programming optimization. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 2378–2385, 2006.

[28] J. Lotti and G. Giraudon. Adaptive Window Algorithm for Aerial Image Stereo. In *the IEEE Conf. on Comput. Vision and Pattern Recogn.*, pages 701–703, 1994.

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

[30] 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.

[31] Y. Nakamura, T. Matsuura, K. Satoh, and Y. Ohta. Occlusion detectable stereo -occlusion patterns in camera matrix-. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, 1996.

[32] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline using dynamic programming. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 7(2):139–154, 1985.

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

[34] J. Park and S. Inoue. Hierarchical depth mapping from multiple cameras. In *Int. Conf. on Image Anal. and Process.*, volume 1, pages 685–692, 1997.

[35] J. Park and S. Inoue. Acquisition of sharp depth map from multiple cameras. *Signal Processing: Image Commun.*, 14:7–19, 1998.

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

[37] S. Roy. Stereo without epipolar lines : A maximum-flow formulation. *Int. J. Comput. Vision*, 34(2/3):147–162, 1999.

[38] S. Roy and M.-A. Drouin. Non-uniform hierarchical pyramid stereo for large images. In *VMV*, pages 403–410, 2002.

[39] M. Sanfourche, G. L. Besnerais, and F. Champagant. On the choice of the correlation term for multi-baseline stereovision. In *Proc. of the IEEE Conf. on British Computer Vision*, September 2004.

[40] D. Scharstein and R. Szeliski. www.middlebury.edu/stereo.

[41] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision*, 47(7), 2002.

[42] H. Schultz. Terrain reconstruction from oblique views. In *ARPA Image Understanding Workshop*, pages 1001–1008, november 1994.

[43] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. *Int. J. Comput. Vision*, 35(2):151–173, 1999.

[44] C. Strecha, R. Fransens, and L. V. Gool. Combined depth and outlier estimation in multi-view stereo. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 2394–2401, 2006.

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

[46] J. Sun, Y. Li, S. B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 399–406, 2005.

[47] J. Sun, N. Zheng, and H. Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 25(7):787–800, July 2003.

[48] R. Szeliski and R. Zabih. An experimental comparison of stereo algorithms. In *Vision Algorithms: Theory and Practice*, pages 1–19. Springer-Verlag, 1999.

[49] O. Veksler. Fast variable window for stereo correspondence using integral images. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, 2003.

[50] O. Veksler. Stereo correspondence by dynamic programming on a tree. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, June 2005.

[51] O. Veksler. Reducing search space for stereo correspondence with graph cuts. In *the British Mach. Vision Conf.*, volume 2, pages 709–718, 2006.

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

[53] Y. Wei and L. Quan. Asymmetrical occlusion handling using graph cut for multi-view stereo. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, pages 902–909, June 2005.

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

[55] K.-J. Yoon and I. S. Kweon. Stereo matching with symmetric cost functions. In *IEEE Conf. on Comput. Vision and Pattern. Recogn.*, volume 2, pages 2371–2377, 2006.