Improving the accuracy of fast dense stereo correspondence algorithms by enforcing local consistency of disparity fields

Stefano Mattoccia
University of Bologna
Dipartimento di Elettronica, Informatica e Sistemistica (DEIS)
Advanced Research Center on Electronic Systems (ARCES)
www.vision.deis.unibo.it/smatt

Abstract

Accurate, dense 3D reconstruction is an important requirement in many applications, and stereo represents a viable alternative to active sensors. However, top-ranked stereo algorithms rely on iterative 2D disparity optimization methods for energy minimization that are not well suited to the fast and/or hardware implementation often required in practice. An exception is represented by the approaches that perform disparity optimization in one dimension (1D) by means of scanline optimization (SO) or dynamic programming (DP). Recent SO/DP-based approaches aim to avoid the well known streaking *effect by enforcing vertical consistency between scanlines deploying aggregated costs, aggregating multiple scanlines, or performing energy minimization on a tree. In this paper we show that the accuracy of two fast SO/DP-based approaches can be dramatically improved by exploiting a non-iterative methodology that, by modeling the coherence within neighboring points, enforces the local consistency of disparity fields. Our proposal allows us to obtain top-ranked results on the standard Middlebury dataset and, thanks to its computational structure and its reduced memory requirements, is potentially suited to fast and/or hardware implementations.*

1. Introduction

Inferring dense accurate 3D reconstruction is an important requirement in many computer vision applications, and depth from stereo is a well known approach to accomplish this task by means of two or more synchronized cameras. In spite of the efforts made in the last decades [17], accurate localization of corresponding points in two or more cameras of the stereo rig (i.e. the *correspondence problem*), is still an important and open problem. According to [17], most dense stereo algorithms perform four steps: *cost computation*, *cost aggregation*, *disparity optimization* and *disparity refinement*. In *local algorithms*, the focus is on cost aggregation and disparity optimization is typically a simple winner takes all (WTA) strategy. Conversely, in *global algorithms*, cost aggregation is often ignored and the focus is on disparity optimization. Local algorithms are potentially faster than global approaches, and their typically limited memory requirements renders these approaches suited to devices with constrained resources (e.g. FPGA, embedded devices). In recent years, for both categories, significant improvements have been proposed. However, it noteworthy that top-ranked algorithms[1] tackle the correspondence problem by deploying 2D disparity optimization methods that enforce the smoothness assumption in vertical and horizontal directions. Unfortunately, in spite of the advent of powerful disparity optimization techniques such as *graph cut* (GC) and *belief propagation* (BP) [18], top-ranked algorithms [13, 24, 26, 25, 27, 3, 1, 22, 19] are typically slow and, mainly due to their iterative nature and their huge memory requirements, unsuited to devices with constrained resources. However, *scanline Optimization* (SO) [17, 10] and *dynamic Programming* (DP) [17, 23, 21], a particular class of disparity optimization methods constrained to subsets of the stereo pair (i.e. 1D), allow for efficiently obtaining reasonable, accurate results. SO/DP-based algorithms are fast [17, 10, 23, 21, 5] and well suited to embedded devices or FPGAs as reported in [7]. A different methodology, referred to as *locally consistent* (LC), aimed at improving the disparity fields provided by local algorithms, was proposed in [15]. The LC approach is an non-iterative technique that enforces the local consistency of disparity fields by explicitly modeling the mutual relationships among neighboring points. In this paper, we show that enforcing local consistency of disparity fields by means of the LC technique enables us to dramatically improve the performance of two fast stereo algorithms based on the SO and DP disparity optimization methods.

[1] http://vision.middlebury.edu/stereo/eval/

2. Related work

In this section, we briefly review disparity optimization methods (focusing our attention on approaches that perform 1D energy minimization) and the LC technique.

2.1. Methods for efficient disparity optimization

Although local algorithms based on *adaptive weights* methods [28, 12, 16], reviewed and evaluated in [8, 20], yield excellent results, according to the Middlebury evaluation site, most of the 10 top-performing stereo algorithms [13, 24, 26, 25, 27, 3, 1, 22, 19, 12] rely on global approaches (i.e [13, 24, 26, 25, 27, 3, 1, 22, 19]). Global approaches pose the correspondence problem in terms of a pixel-labeling assignment of disparities searching for the disparity field D that minimizes the following energy function:

$$E(D) = E_{data}(D) + E_{smooth}(D) \quad (1)$$

The data term E_{data} encodes how well the disparity assignment fits with the stereo pair and, often, it is the sum of per-pixel data costs $C(D(p)$ (see [11] for an evaluation of costs typically deployed in stereo) between one point in the reference image R and the supposed correspondent point in the target image T,

$$E_{data}(D) = \sum_{p \in R} C(D(p)) \quad (2)$$

Sometimes (e.g. [26, 23]) the point-wise data cost $C(D(p)$ is replaced by an aggregated cost computed over neighboring pixels (a *support* window centered in each examined point).

The smoothness term $E_{smooth}(D)$ penalizes disparity changes modeling the interaction between each point p and its neighboring pixels $q \in \mathcal{N}(p)$. In global approaches, $\mathcal{N}(p)$ includes points in vertical and horizontal directions (typically, the four nearest neighbors of p on the pixel grid), while in SO/DP-based approaches, the smoothness constraint is enforced only in one direction (typically $\mathcal{N}(p)$ includes only one point along a scanline). The former disparity optimization methods are referred to as 2D, while the latter are referred to as 1D. Clearly, 2D approaches perform better as they enable us to enforce *inter* and *intra* scanline smoothness assumptions.

Some approaches use additional terms for penalizing occlusions, enforcing consistency or visibility between images, and often segmentation is deployed as a further additional constraint [10]. Top-ranked stereo algorithms minimize energy function (1), performing a 2D disparity optimization. However, since this turns to be a $\mathcal{NP}$-hard problem, global approaches [13, 26, 25, 3, 1, 22, 19] under particular hypotheses [18] on (1) rely on efficient energy minimization strategies based on GC or BP. Nevertheless, GC and BP are often too slow for practical use. Moreover, their iterative computational structure and their memory requirements render these approaches unsuited to devices with limited resources. The algorithmic optimization strategies proposed in [6] significantly reduce the running time as well the memory requirements of the BP algorithm. However, the effectiveness of this approach is not comparable to that of the original BP approach deployed by most top-ranked algorithms.

Conversely, SO [17] and DP [2, 17] approaches perform a 1D optimization restricted to individual scanlines that efficiently minimizes (1) in polynomial time. SO minimizes (1) in a single phase, while DP minimizes it in two phases enforcing the *ordering* constraint. The memory requirement for both approaches is quite low; for SO it is proportional to the disparity range $d_{max} - d_{min}$, while for DP it is proportional to $W \times (d_{max} - d_{min})$ (being W the image width). However, the 1D optimization adopted by basic SO and DP approaches leads to the well-known *streaking* effect. To overcome this problem, the key idea is to enforce constraints incoming from regions outside the individual scanline. Three methodologies have been proposed. The first methodology [23], referred to as RTGPU, uses as a data term the cost computed over a vertical support by means of the adaptive weight approach [28] in order to include cues from neighboring scanlines. RTGPU runs in real time, computing the aggregated cost on a GPU and the DP-based optimization on the CPU. The second methodology [10] avoids the streaking effect by combining SOs, computed along multiple directions (typically 8 or 16). This method, referred to as C-Semiglobal, is the best performing SO/DP based approach on the Middlebury evaluation site and it has a running time of few seconds. Compared to the original Semiglobal approach [9], C-Semiglobal includes a disparity refinement step based on *mean shift* segmentation [4]. The third methodology avoids the streaking effect by performing 1D disparity optimization on a tree rather than on a scanline. In [21] DP-based optimization is performed on a tree structure made of the *most important* edges. This approach was improved by replacing edges with segments: [14] deploys the mean shift algorithm while [5] deploys a fast line segmentation algorithm. Algorithm [14] has a running time of about 10 seconds while [21] and [5] run in near real time.

2.2. The LC technique

A different methodology aimed at enforcing the local consistency of a dense disparity field was proposed in [15]. The LC approach explicitly models the mutual relationships among neighboring pixels of a dense disparity field provided by a local algorithm deriving a posterior probability for each disparity hypothesis. Disparity coherence is enforced by performing a *local optimization* restricted to

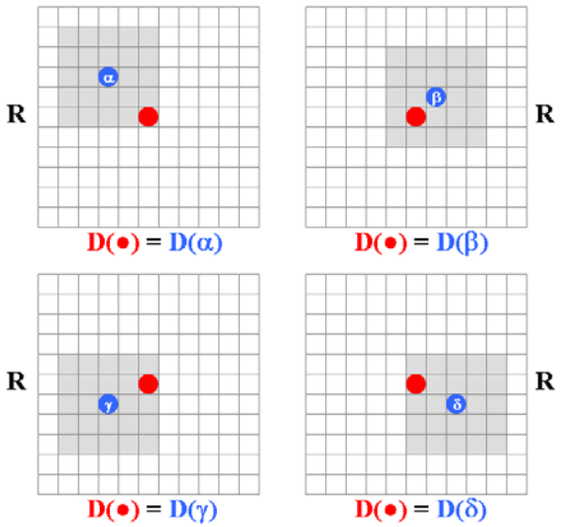

Figure 1. In blue are reported, for reference image R, four points examined by a local algorithm that deploys a 5×5 squared support. The local algorithm sets for the four blue points $\alpha, \beta, \gamma, \delta$ four potentially different disparity hypotheses (i.e. $D(\alpha), D(\beta), D(\gamma), D(\delta)$). Therefore, within each support, the same red point implicitly assumes four potentially different disparity hypotheses (according to the blue points). In the figure we show only 4 out of 25 possible configurations resulting with 5×5 supports. **[Best viewed in color]**

neighboring points.

In local algorithms, once a disparity for one point is determined, implicit assumptions concerned with the points within the support window are made. That is, deploying *frontal-parallel* support, as typically made by most local approaches, each element of the support is assumed at the same disparity of the central point. The LC approach explicitly models this behavior by enforcing that the resulting disparity field is piecewise smooth. Let's assume that for each point of the reference image, the disparity is determined by a local algorithm that deploys a 5×5 support window (i.e. the matching cost for the central point of the 5×5 window is the sum of the matching costs within the support). Once the optimal disparity for the central point has been selected (typically by means of a WTA strategy in local approaches), all the points within the 5×5 support are implicitly assumed at the same disparity of the central point. For example, in Figure 1 is depicted this behavior for the reference image R deploying a 5×5 frontal-parallel support window. In this case the same red point is included in 25 different support windows. However, for simplicity, in Figure 1 we report only 4 out of 25 possible configurations. The figure shows that for the 4 different blue points $\alpha, \beta, \gamma, \delta$ the same red point is potentially assumed at 4 different disparity $D(\alpha), D(\beta), D(\gamma), D(\delta)$ by the 5×5 support windows deployed in this example. More precisely, with a $M \times N$ support window, each point of the reference image (e.g. the red point in Figure 1) is subject to $M \times N$ disparity hypotheses by neighboring points (e.g. blue points in Figure 1). Given a certain support window size, the set of points that are allowed to pose a disparity hypothesis for the same red point p is called the *active support* of point p (we note that the active support for point p, in this example, is given by the 5×5 square window centered in p). Moreover, it is noteworthy that a similar behavior can be modeled in the target image T since, once a correspondence has been established, the same disparity hypothesis for each point of the support window is also made in T. However, in this case, the active support in T is distributed according to the disparity hypothesis established for each correspondence (i.e. this means that points within the active support of point p' in the target image might not set any disparity hypothesis for p'). Each time that one red point is included by the support of a blue point the belief of the disparity hypothesis implicitly made for the red point is encoded by the *plausibility* [15] of that event. Plausibility is defined according to the image content as the posterior probability of the three joint events as depicted in Figure 2.

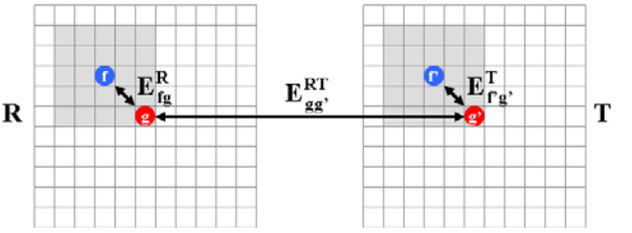

Figure 2. The three events involved in the computation of plausibility for the red points in R and T given a certain disparity hypothesis for the blue points in R and T. **[Best viewed in color]**

That is, modeling local surfaces as frontal-parallel, the posterior probability $P^R_{f \rightharpoonup g}(D(f))$ of a specific disparity hypothesis $D(f)$ implicitly made by the blue point f on the red point g is related to the spatial proximity between (f, g), (f', g') and to the color proximity between (f, g), (f', g'), (f, f'). Color and spatial proximity are encoded according to the Euclidean distance (see [15] for details). For each point of reference and target image, the plausibility of each disparity hypothesis made by the points belonging to the active support are gathered. Thus, once every element belonging to the active support $S(g)$ of g of the reference image has propagated its plausibility towards g, $\Omega^R(g|d)$ encodes for g the accumulated plausibility of each disparity hypothesis $d \in [d_{min}, d_{max}]$. Formally, for R and T the accumulated plausibility for disparity hypothesis d is:

$$\Omega^R(g \mid d) = \sum_{i \in S(g)} P^R_{i \rightharpoonup g}(d) \qquad (3)$$

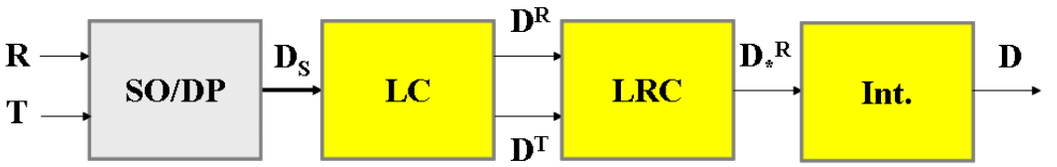

Figure 3. Proposed overall approach.

$$\Omega^T(g' \mid -d) = \sum_{i' \in S'(g')} P^T_{i' \rightarrow g'}(-d) \qquad (4)$$

Equation (3) provides for each point of reference image R and for each disparity hypothesis the degree of consistency with neighboring points belonging to the active support S.

3. Enforcing local consistency in SO/DP

Since our proposal aims at improving the accuracy of fast algorithms for stereo correspondence, we restrict our attention to approaches that rely on 1D techniques for disparity optimization. However, in contrast to previous approaches [10, 9, 21, 14, 5, 23] that embody in the energy terms cues from neighboring scanlines, we enforce, by deploying the LC approach [15], *a posteriori* local consistency of the dense disparity field provided by SO/DP-based algorithms. Our proposal can be summarized as depicted in Figure 3. We assume that a SO/DP-based algorithm computes a dense disparity field D_S, processing two rectified images R and T. As SO/DP-based approaches we have considered in our experiments C-Semiglobal [10] and RTGPU [23]. Among the approaches based on 1D disparity optimization reported on the Middlebury evaluation site, the former is currently the top-ranked, while the latter is one of the fastest approaches. The D_S disparity fields provided by these algorithms are processed according to the LC technique that, for each point of R and T, provides the accumulated plausibilities (3) and (4) for each disparity hypothesis within the disparity range $[d_{min}, d_{max}]$. The accumulated plausibilities (3) and (4) are normalized by the overall plausibility of each point in order to obtain values ranging from 0 to 1. Afterwards, the normalized accumulated plausibilities are cross checked (i.e. multiplied) with their counterpart in the other image according to the disparity range $[d_{min}, d_{max}]$. That is, for reference image R:

$$\Omega^{RT}(g \mid d) = \frac{\Omega^R(g \mid d)}{\sum_{i \in [d_{min}, d_{max}]} \Omega^R(g \mid i)} \cdot \frac{\Omega^T(g-d \mid -d)}{\sum_{j \in [d_{min}, d_{max}]} \Omega^R(g-j \mid -j)} \qquad (5)$$

This step allows us to emphasize consistent disparity hypotheses within the reference R and the target image T. The same process is applied to the plausibilities computed according to the target image T (i.e. $\Omega^{TR}(g' \mid -d)$). Although not essential, we re-normalize the resulting cross-checked plausibility in order to obtain values ranging from 0 to 1. The normalized accumulated plausibility $\Omega^{RT}(g \mid d)$ for reference image concerned with five disparity hypotheses (i.e. $d = 4$,$d = 5$,$d = 6$,$d = 7$,$d = 8$) of the Tsukuba stereo pair is shown in Figure 4. The results reported in the figure are concerned with the disparity field D_S of [10] available on Middlebury. Once we have computed the two normalized plausibility distributions, we simply choose for each point of R and T the most plausible disparity label. This allows us to obtain the two D^R and D^T disparity fields, concerned with the reference and target image respectively, as depicted in Figure 3. Although we obtain two independent disparity fields D^R and D^T, it is noteworthy that the most time consuming task required to obtain accumulated normalized plausibilities (i.e. equations (3) and (4)) is computed only once for the two images. The two disparity fields D^R and D^T are left-right checked (LRC in Figure 3) in order to detect inconsistent disparity assignments. As the resulting disparity field D^R_* contains missing disparity assignments, we fill-in these values in D^R_* by means of a simple interpolation stage (referred to as Int. in Figure 3). The interpolation step scans the image in row order and replaces the missing assignments with the lower values between the two disparities that bound the missing values. Finally, we process the interpolated disparity field with a single iteration of a median filter and a bilateral filter. Although not essential, this latter step allows us to smooth the effect of the very simple interpolation stage. We do not use segmentation (although the SO/DP algorithm that provides the disparity field D_S might use this cue). Moreover, our proposal has some properties that we briefly highlight. LC allows us to obtain two independent depth maps in a single pass by simply accumulating the disparity hypotheses made on R and T. The memory requirements for the same module are quite low: in fact, an efficient implementation of LC would store only $W \times (d_{max} - d_{min}) \times N$ values, N being the height of the active support deployed (in our successive experiments we deploy 39×39 squared active supports). LC is a non-iterative technique and its computational structure

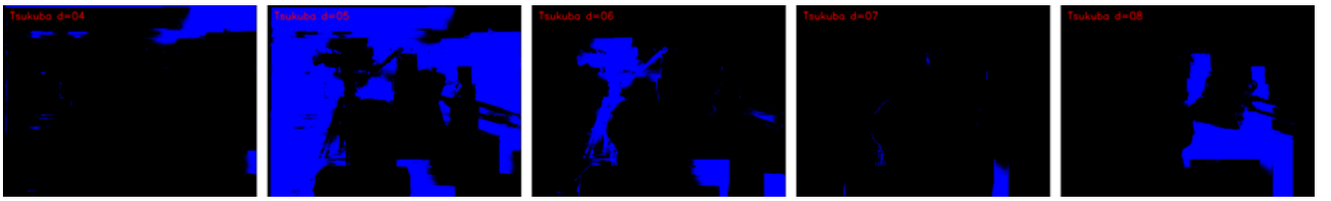

Figure 4. Accumulated plausibility for reference image R concerned with five disparity hypotheses (i.e. from left to right $d = 4$,$d = 5$,$d = 6$,$d = 7$,$d = 8$) of the Tsukuba stereo pair deploying the disparity field D_S of [10] available on Middlebury.

is very simple and regular. This fact, combined with the efficiency of the 1D disparity optimization methods deployed to obtain the initial D_S disparity field, might allow us to exploit the powerful processing capabilities provided by state-of-the-art computing architectures (multicore CPU, GPU, SIMD instructions, FPGA) as already done for SO/DP approaches (e.g. [7],[23]). However, it important to note that the experimental results reported in this paper do not take advantage of these capabilities and are concerned with our unoptimized C++ implementation of the LC approach and the disparity fields D_S for [23] and [10] available on the Middlebury site. Finally, we observe that, although we have restricted our attention to fast dense stereo algorithms based on SO and DP, our proposal could be potentially applied to any dense disparity field.

4. Experimental results

In order to assess the effectiveness of our proposal, we consider two different representative algorithms deploying 1D and multiple 1D disparity optimization techniques. The two considered algorithms are RTGPU [23] and C-Semiglobal [10]. The former is one of the fastest algorithms on the Middlebury evaluation site. RTGPU delivers disparity fields in real time by deploying the processing capabilities of GPU (for cost aggregation) and CPU (for disparity optimization). The 1D optimization technique deployed is DP and currently it is ranked[2] 61th on Middlebury among 80 approaches. The latter (C-Semiglobal) is currently the top-ranked algorithm among the approaches that rely on 1D disparity optimization. In this case, 16 independent SO along as many different paths are aggregated for each disparity hypothesis. C-Semiglobal uses a disparity refinement step based on segmentation and the overall running time is a few seconds on a standard PC deploying SIMD instructions [10]. Currently, this method is ranked 17th. For what concerns our implementation of the LC approach [15] we deploy a 39×39 active support but we do not exploit GPU, nor SIMD nor multi-core processing capabilities. Parameters of the LC approach according to [15] are reported in Figures 5 and 6.

Table 1 reports[3] the accuracy, according to the three errors NOCC, ALL, DISC (respectively: errors in non occluded regions, errors on the whole image, errors within discontinuity regions), defined on the Middlebury evaluation site by the two algorithms considered (i.e. RTGPU and C-Semiglobal) and the top-ranked algorithms. The table also reports on the accuracy of our proposal by deploying the disparity fields provided by RTPGU and C-Semiglobal (referred to as LC(RTGPU) and LC(C-Semiglobal) respectively). Comparing RTGPU and the results obtained by our proposal we notice a dramatic overall improvement for each stereo pair and for each parameter (i.e. NOCC, ALL, DISC). However, improvements are particularly notable on Tsukuba and Venus. Our proposal allows us to gain more than 45 positions (from 61 to 14) on the current Middlebury ranking. Moreover, our proposal outperforms several algorithms based on 2D disparity optimization as well as the original C-Semiglobal.

Observing the same table we notice that our LC(Semiglobal) proposal allows us to significantly improve the overall performance of the already very effective C-Semiglobal approach. In this case, enforcing the local consistency of the C-Semiglobal disparity field renders our proposal the 5th best performing algorithm. This allows us to improve the ranking of 12 positions (from 17 to 5 on the current Middlebury evaluation site). Observing the results reported in the table for C-Semiglobal, we notice that enforcing local coherence allows us to dramatically improve the effectiveness of the original algorithm, especially on Tsukuba and Venus. These results highlight that a simple frontal-parallel support assumption deployed by LC is more suited to images that best fit with this model (i.e. images that are mostly made of frontal-parallel or slanted surfaces such as Tsukuba and Venus).

Figures 5 and 6 show the original disparity fields and the errors according to Middlebury methodology of the two original approaches (RTGPU and C-Semiglobal) on the left. On the right side of both images we report the disparity fields and the errors provided by our proposal. Observing the two figures we can clearly perceive in both cases the

[2]For simplicity we also include our proposals

[3]Additional experimental results in:
http://www.vision.deis.unibo.it/smatt/3DPVT2010.htm

Algorithm	Rank	Tsukuba			Venus			Teddy			Cones		
		NOCC	ALL	DISC	NOCC	ALL	DISC	NOCC	ALL	DISC	NOCC	ALL	DISC
AdaptingBP [13]	1	1.11	1.37	5.79	0.10	0.21	1.44	4.22	7.06	11.8	2.48	7.92	7.32
CoopRegion [24]	2	0.87	1.16	4.61	0.11	0.21	1.54	5.16	8.31	13.0	2.79	7.18	8.01
DoubleBP [26]	3	0.88	1.29	4.76	0.13	0.45	1.87	3.53	8.30	9.63	2.90	8.78	7.79
OutlierConf [25]	4	0.88	1.43	4.74	0.18	0.26	2.40	5.01	9.12	12.8	2.78	8.57	6.99
LC(C-Semiglobal)	**5**	1.08	1.57	5.86	0.13	0.25	1.86	5.56	11.0	13.9	2.86	8.31	7.50
SubPixDoubleBP [27]	6	1.24	1.76	5.98	0.12	0.46	1.74	3.45	8.38	10.0	2.93	8.73	7.91
WarpMat [3]	7	1.16	1.35	6.04	0.18	0.24	2.44	5.02	9.30	13.0	3.49	8.47	9.01
Undr+OvrSeg [1]	8	1.89	2.22	7.22	0.11	0.22	1.34	6.51	9.98	16.4	2.92	8.00	7.90
GC+SegmBorder [22]	9	1.47	1.82	7.86	0.19	0.31	2.44	4.25	5.55	10.9	4.99	5.78	8.66
AdaptOvrSegBP [19]	10	1.69	2.04	5.64	0.14	0.20	1.47	7.04	11.1	16.4	3.60	8.96	8.84
GeoSup [12]	11	1.45	1.83	7.71	0.14	0.26	1.90	6.88	13.2	16.1	2.94	8.89	8.32
PlaneFitBP	12	0.97	1.83	5.26	0.17	0.51	1.71	6.65	12.1	14.7	4.17	10.7	10.6
SymBP+occ	13	0.97	1.75	5.09	0.16	0.33	2.19	6.47	10.7	17.0	4.79	10.7	10.9
LC(RTGPU)	**14**	1.02	1.68	5.50	0.31	0.63	3.17	6.36	12.1	14.3	4.14	10.0	10.6
AdaptDispCalib	15	1.19	1.42	6.15	0.23	0.34	2.50	7.80	13.6	17.3	3.62	9.33	9.72
Segm+visib	16	1.30	1.57	6.92	0.79	1.06	6.76	5.00	6.54	12.3	3.72	8.62	10.2
C-Semiglobal [10]	**17**	2.61	3.29	9.89	0.25	0.57	3.24	5.14	11.8	13.0	2.77	8.35	8.20
. . .	**...**	...	...	...	...	...	...	...	...	...	...	...	...
RTGPU [23]	**61**	2.05	4.22	10.06	1.92	2.98	20.3	7.23	14.4	17.6	6.41	13.7	16.5
. . .	**...**	...	...	...	...	...	...	...	...	...	...	...	...

Table 1. Accuracy (errors NOCC, ALL, DISC) according to the methodology defined on the Middlebury evaluation site [17] for top-ranked algorithms. In boldface, the results provided by the two considered algorithms, **C-Semiglobal** [10] and **RTGPU** [23], and their locally consistent versions, **LC(C-Semiglobal)** and **LC(RTGPU)**, proposed in this paper.

improvements brought in enforcing the local consistency of the original disparity fields.

Current C++ implementation of our proposal, without a specific optimization, has a running time on a standard PC (Intel Core 2 Quad CPU at 2.49 GHz) of 15 seconds (on Teddy and Cones). For RTGPU, the further overhead required to obtain the initial disparity field D_S would be negligible while for C-Semiglobal it would be a few seconds according to [10].

5. Conclusions

In this paper we have shown that the effectiveness of fast stereo algorithms based on 1D disparity optimization techniques can be dramatically improved by enforcing the local consistency of disparity fields. The overall approach proposed requires a few seconds on a PC and allows us to obtain top-ranked results on the standard Middlebury dataset. Moreover, thanks to its overall computational structure, our proposal is potentially suited to fast and/or hardware implementations.

References

[1] Anonymous. Stereo matching based on under- and over-segmentation with occlusion handling. *IEEE Trans. PAMI (submitted)*, 2009. 1, 2, 6

[2] S. Birchfield and C. Tomasi. Depth discontinuities by pixel-to-pixel stereo. *Int. J. Comput. Vision*, 35(3):269–293, 1999. 2

[3] M. Bleyer, M. Gelautz, C. Rother, and C. Rhemann. A stereo approach that handles the matting problem via image warping. In *CVPR09*, pages 501–508, 2009. 1, 2, 6

[4] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. PAMI*, 24:603–619, 2002. 2

[5] Y. Deng and X. Lin. A fast line segment based dense stereo algorithm using tree dynamic programming. In *Proc. European Conf. on Computer Vision (ECCV 2006)*, volume 3, pages 201–212, 2006. 1, 2, 4

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70(1):41–54, 2006. 2

[7] S. Gehrig, F. Eberli, and T. Meyer. A real-time low-power stereo vision engine using semi-global matching. In *CVS09*, pages 134–143, 2009. 1, 5

[8] M. Gong, R. Yang, W. Liang, and M. Gong. A performance study on different cost aggregation approaches used in real-time stereo matching. *Int. Journal Computer Vision*, 75(2):283–296, 2007. 2

[9] H. Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Proc.*

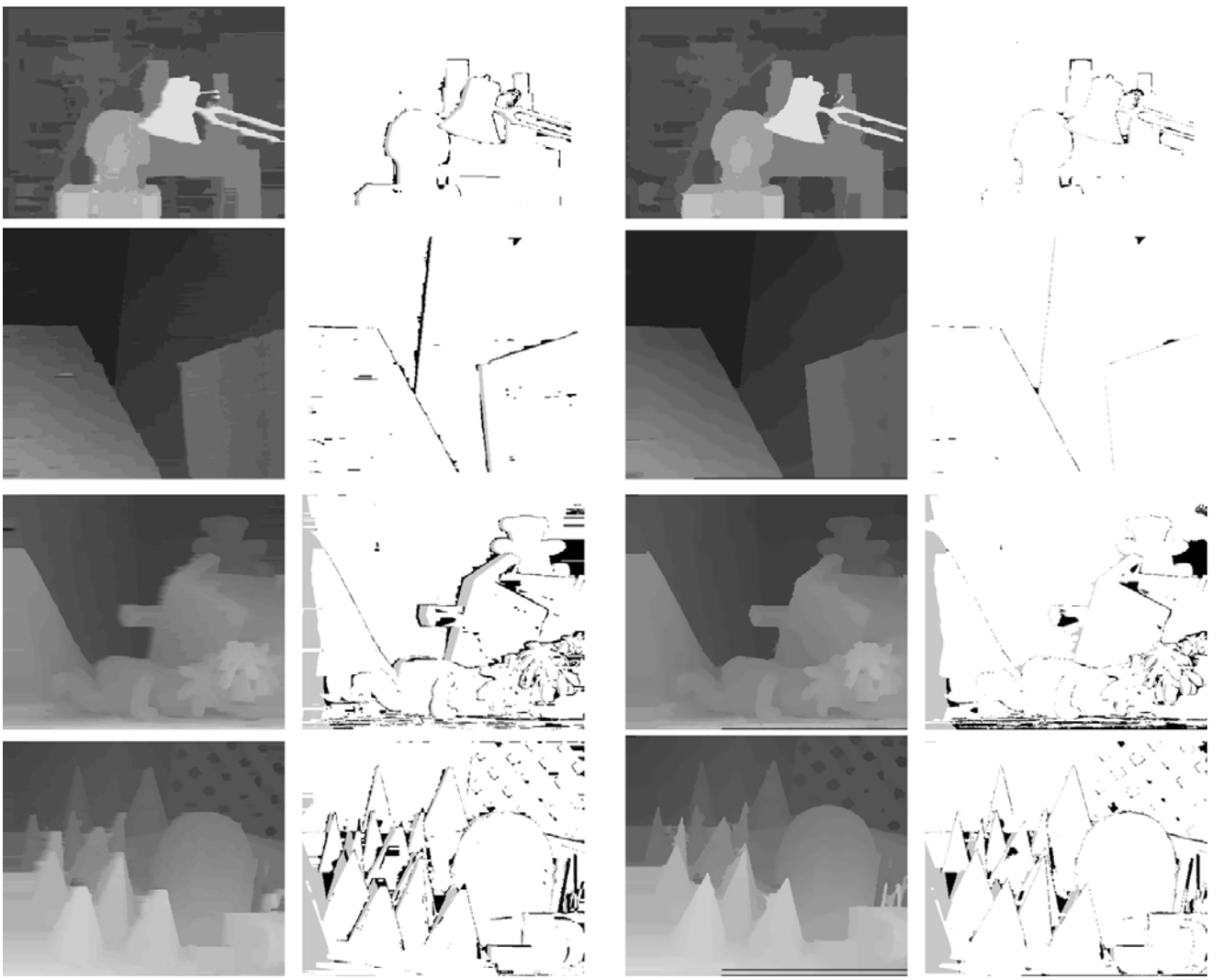

Figure 5. (Left) Disparity fields and errors of the original **RTGPU** approach [23] (Right) Disparity fields and errors of the proposed **LC(RTGPU)** approach. Parameters of the LC technique according to [15]: $\gamma_s = 15.5$, $\gamma_c = 28$, $\gamma_t = 12.5$, $\rho = 66$.

Conf. on Computer Vision and Pattern recognition (CVPR 2005), volume 2, pages 807–814, 2005. 2, 4

[10] H. Hirschmuller. Stereo processing by semi-global matching and mutual information. *IEEE Trans. on PAMI*, 2(30):328–341, 2008. 1, 2, 4, 5, 6, 8

[11] H. Hirschmuller and D. Scharstein. Evaluation of stereo matching costs on images with radiometric differences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(9):1582–1599, 2009. 2

[12] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann. Local stereo matching using geodesic support weights. In *ICIP*, 2009. 2, 6

[13] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR '06*, pages 15–18, 2006. 1, 2, 6

[14] C. Lei, J. Selzer, and Y. Yang. Region-tree based stereo using dynamic programming optimization. In *CVPR06*, pages II: 2378–2385, 2006. 2, 4

[15] S. Mattoccia. A locally global approach to stereo correspondence. In *3DIM2009*, pages 1763–1770, Kyoto, Japan, 2009. 1, 2, 3, 4, 5, 7, 8

[16] S. Mattoccia, S. Giardino, and A. Gambini. Accurate and efficient cost aggregation strategy for stereo correspondence based on approximated joint bilateral filtering. In *Proc. of ACCV2009*, 2009. 2

[17] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. Jour. Computer Vision*, 47(1/2/3):7–42, 2002. 1, 2, 6

[18] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for

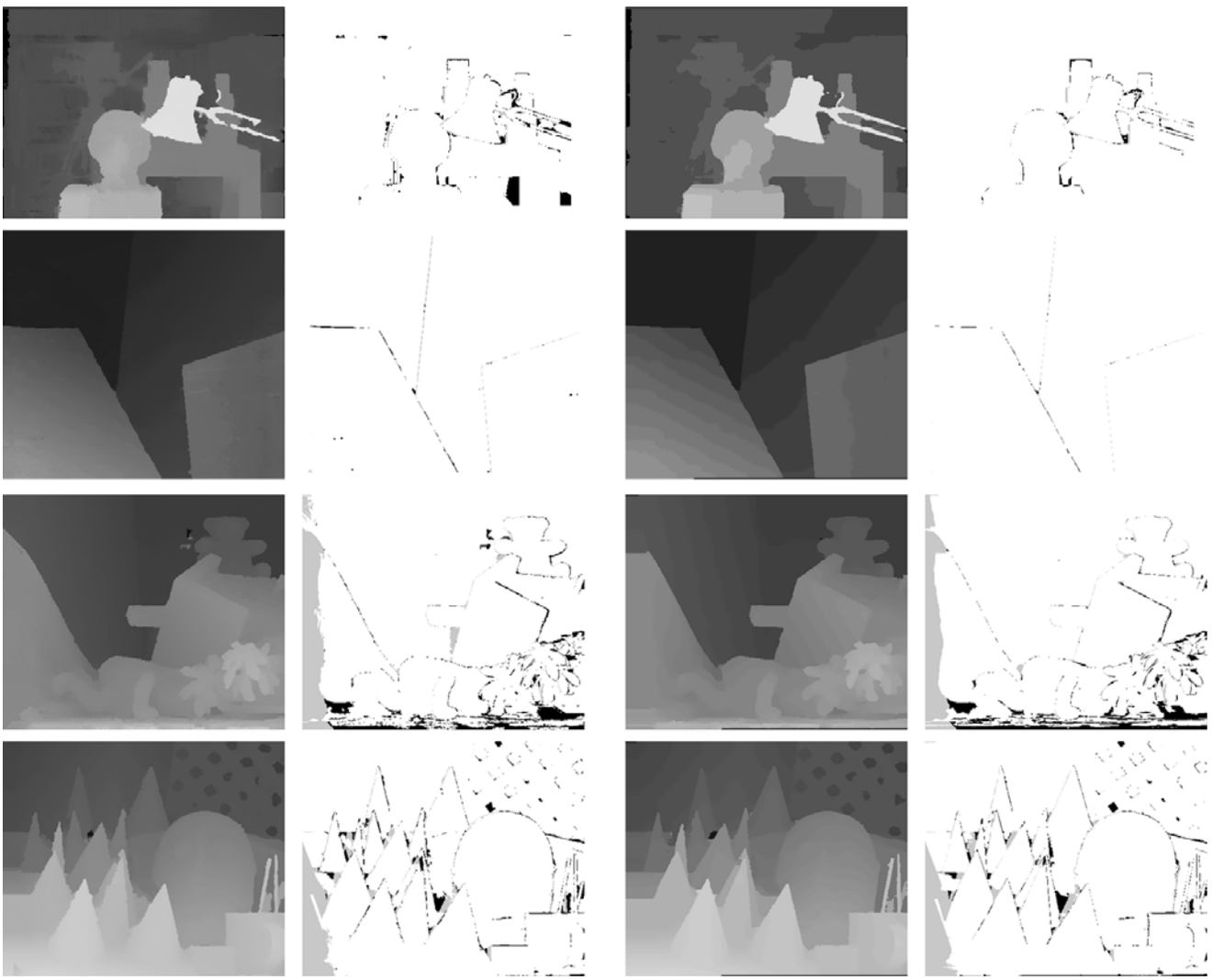

Figure 6. (Left) Disparity fields and errors of the original **C-Semiglobal** approach [10] (Right) Disparity fields and errors of the proposed **LC(C-Semiglobal)** approach. Parameters of the LC technique according to [15]: $\gamma_s = 16$, $\gamma_c = 27.5$, $\gamma_t = 11$, $\rho = 62$.

markov random fields with smoothness-based priors. *IEEE Trans.PAMI*, 30(6):1068–1080, 2008. 1, 2

[19] Y. Taguchi, B. Wilburn, and C. Zitnick. Stereo reconstruction with mixed pixels using adaptive over-segmentation. In *CVPR08*, 2008. 1, 2, 6

[20] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *CVPR08*, pages 1–8, 2008. 2

[21] O. Veksler. Stereo correspondence by dynamic programming on a tree. In *CVPR '05*, pages 384–390, 2005. 1, 2, 4

[22] W. W. Chen, M. Zhang, and Z. Xiong. Segmentation-based stereo matching with occlusion handling via region border constrains. *CVIU (submitted)*, 2009. 1, 2, 6

[23] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nister. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. In *3DPVT '06*, pages 798–805, 2006. 1, 2, 4, 5, 6, 7

[24] Z.-F. Wang and Z.-G. Zheng. A region based stereo matching algorithm using cooperative optimization. In *CVPR*, 2008. 1, 2, 6

[25] L. Xu and J. Jia. Stereo matching: An outlier confidence approach. In *ECCV 2008*, pages 775–787, 2008. 1, 2, 6

[26] Q. Yang, L. Wang, R. Yang, H. Stewénius, and D. Nistér. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Trans. PAMI*, 31(3):492–504, 2009. 1, 2, 6

[27] Q. Yang, R. Yang, J. Davis, and D. Nistér. Spatial-depth super resolution for range images. In *Proc. of CVPR2007*, pages 1–8, 2007. 1, 2, 6

[28] K. Yoon and I. Kweon. Adaptive support-weight approach for correspondence search. *IEEE Trans. PAMI*, 28(4):650–656, 2006. 2