A FAST SEGMENTATION-DRIVEN ALGORITHM FOR ACCURATE STEREO CORRESPONDENCE

Stefano Mattoccia and Leonardo De-Maeztu

University of Bologna, Public University of Navarre

ABSTRACT

Recent cost aggregation strategies that adapt their weights to image content enabled local algorithms to obtain results comparable to those of global algorithms based on more complex disparity optimization methods. Unfortunately, despite the potential advantages in terms of memory footprint and algorithmic simplicity compared to global algorithms, most of the state-of-the-art cost aggregation strategies deployed in local algorithms are extremely slow. In fact, their execution time is comparable and often worse than those of global approaches. In this paper we propose a framework for accurate and fast cost aggregation based on segmentation that allows us to obtain results comparable to state-of-the-art approaches much more efficiently (the execution time drops from minutes to seconds). A further speed-up is achieved taking advantage of multi-core capabilities available nowadays in almost any processor. The comparison with state-of-the-art cost aggregation strategies highlights the effectiveness of our proposal.

Index Terms— Stereo vision, local algorithms, cost aggregation, segmentation, adaptive weights

1. INTRODUCTION

Depth from stereo is a widely researched topic, extensively reviewed in [1, 2], aimed at inferring depth from two images of the same scene simultaneously acquired from two different viewpoints. Finding homologous points in the two images is a challenging task and many algorithms have been proposed in the last decades.

According to [1, 2] most approaches perform four steps (*cost computation*, *cost aggregation*, *disparity optimization* and *refinement*) and algorithms can be roughly classified in *local* approaches and *global* approaches. The former class mainly relies on cost aggregation and in most cases ignores disparity optimization deploying, on a point basis, a simple *Winner Takes All* (WTA) strategy. On the other hand, global algorithms minimize iteratively on the whole image an energy function made of two terms, a point-wise matching cost that enforces photometric consistency and a smoothness term that takes into account the evidence that scenes are piecewise smooth. These algorithms typically do not perform cost aggregation focusing on disparity optimization. However, although very effective, global algorithms are in most cases computationally expensive and have a very large memory footprint. These drawbacks render these algorithms not suited to most practical applications.

In this paper we propose a fast local algorithm based on segmentation that allows us to obtain results comparable to state-of-the-art approaches based on adapting weights in a fraction of the time required by its original counterpart. Our proposal casts an accurate algorithm based on segmentation within a framework that enables us to exploit very efficiently the adapting weight strategy at a coarse level. To this aim we use a strategy that brings within an efficient and traditional *correlative approach* [1, 2] based on incremental calculations schemes [3, 4] the effectiveness of the adapting weight strategy exploiting the additional cues provided by segmenting both images. Moreover, we further reduce the execution time of our proposal exploiting multi-core capabilities available nowadays in almost any general purpose and embedded processor.

2. RELATED WORK

In local algorithms robustness to noise is increased aggregating costs within a *support* region, a small area centered in the examined points (one in the reference and one in the target image). In most cases the underlying implicit assumption made by these algorithms is that each point within the support has the same depth of the central point (i.e. *frontal-parallel* assumption). However, in a sensed scene this assumption is violated near depth discontinuities and within non frontal-parallel surfaces. Simpler local algorithms ignore both problems; this enables very fast implementation while more advanced local approaches adapt the weight assigned to each point within the support according to the image content. In these approaches, weights are used to model the probability that two pixels belong to the same object according to the frontal-parallel assumption. Since the disparity of the pixels in the stereo pair is not known beforehand, weights are computed using the information available in the stereo pair (e.g color, segmentation).

The adapting weight strategy allows to deal with depth discontinuities but top-performing algorithms based on this approach [5, 6, 7] are very slow due to the computational

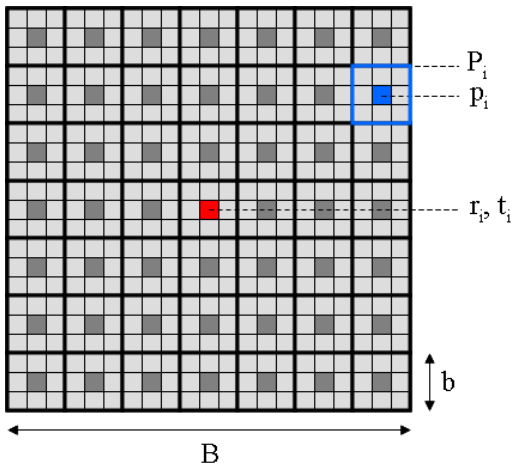

Fig. 1. Support of reference and target images concerned with the central point depicted in red. We partition the support of size $B \times B$ in non overlapping blocks of equal size $b \times b$. In this specific case, the support, of size 21×21, is split in 49 blocks of size 3×3.

complexity of weight computation and cost aggregation for large support windows. Some efficient simplified algorithms have been proposed for [5] and [6] respectively in [8, 9] and [10]. However, the resulting disparity maps compared to the original counterpart are typically less accurate; nevertheless, these simplified algorithms are suited for real-time or near real-time implementation exploiting GPU architectures [10, 9]. In between there are approaches that adapt the shape of the support to the image content according to different strategies (see [1], [11] and [12] for a detailed review of these approaches). Some of these latter methods are fast but their accuracy is clearly outperformed by approaches based and adaptive weights [5, 6, 7].

3. PROPOSED FAST SEGMENTATION-DRIVEN APPROACH

Although the raw Segment Support (SS) approach [7] turned out to be very effective its execution time of several minutes is not suited for most practical applications. Therefore, in this paper we propose to cast this method within a computational framework similar to those proposed in [8]. The Fast Bilateral Stereo (FBS) framework [8] provided a link between traditional fast and inaccurate correlative approaches and the accurate but computationally expensive Adaptive Weight (AW) algorithm [5]. The key idea in FBS was to determine weights according to image content on a block basis and to compute matching costs efficiently and accurately on a point-basis by means of integral images [3] or box-filtering [4].

In this paper we follow this strategy computing weights on a block-basis deploying as additional cues the segmented

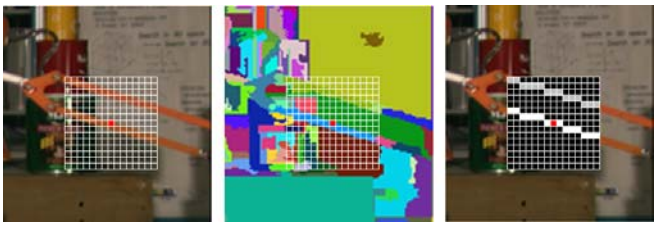

Fig. 2. Segmentation-based weight computation: (left) reference or target image, (center) corresponding segment (*Mean-shift*), (right) weight computation. Each block has size $b \times b$. **[Best viewed in color]**

stereo pair. For our experiments we deployed for segmentation the Mean-Shift [13] algorithm but other algorithms can be used as well. As depicted in Figure 1, for each point belonging to the reference and the target image we split the supports of size $B \times B$ in non overlapping and equal regions of size $b \times b$. On each block within the support of each point we compute on a point-basis and very efficiently the matching cost by means of box-filtering [4] and assign to each block a single weight according to the following symmetric (i.e. considering both images) strategy. Within each support of two potential corresponding points r_i in reference image R and t_i in target image T, for each block P_i centered in the central point of the block p_i, we assign weight $w_i = 1$ if p_i is on the same segment of the central point of the support (respectively, r_i for R and t_i for T). Conversely, if p_i does not belong to the same segment of the point at the center of the support (i.e. r_i for R and t_i for T) the cost assigned to the block is given by

$$w_i = \exp\left(-\frac{\Delta^2}{\gamma}\right) \tag{1}$$

being γ a constant determined empirically, and Δ the Euclidean distance between corresponding RGB channels of the point at the center of the support and the point at the center of the examined block. The proposed block-based weight computation strategy is depicted in figure 2; the same procedure applies to reference and target image. Corresponding weights computed on each block of the reference and of the target image are then multiplied so as to assign an overall weight to the matching cost (Truncated Absolute Differences (TAD)) computed on a point basis within each block. The weighted matching cost are summed-up so as to obtain the overall matching cost assigned to each correspondence. It is worth observing that, compared to SS [7], this strategy reduces significantly (by a factor b^2) the number of weight computations and the number of additions and multiplications required to obtain the overall matching cost. On the other hand it also reduces the accuracy in weight computation due to the rough localization of each point within the blocks. However, as will be shown in the next section, computing costs on a point basis allows us to compensate for this behavior and to

obtain much more efficiently results equivalent to SS.

4. EXPERIMENTAL RESULTS

The Middlebury stereo evaluation website [14] provides a convenient framework for evaluating the accuracy of the reconstruction by the percentage of bad pixels in the computation of four disparity maps using four stereo pairs named Tsukuba, Venus, Teddy and Cones. For each disparity map, Middlebury provides three statistics: the error rates *NOCC* (all points except for occluded areas), *DISC* (only points near depth discontinuities, not including occluded areas) and *ALL* (all points for which the true disparity is known). In this paper, following a similar evaluation procedure proposed in [11], error rates on all image points including occlusions (*ALL*) have not been taken into account since the tested algorithms do not explicitly handle disparity retrieval for occluded points due to the adopted WTA strategy. Since the focus of this paper is both on accuracy and execution time, the execution time was measured with similar processors (i.e. Core Duo @2.5 GHz for our experiments and Intel Core Duo @2.14 GHZ in [11]). In our experiments, the measured execution time for FSD does not include the segmentation algorithm, that for Teddy accounts for less than 2 seconds (for both images).

According to the described evaluation methodology, Table 1 shows the results for our proposal FSD and top-ranked algorithms according[1] to [11]: FBS [8], SS [7], SB [15], AW [5], REL [16] and VW [17]. The parameters of our algorithm have been optimized to produce the best possible results for blocks of size 3×3 and 5×5 with $B = 45$ and 7×7 with $B = 49$. The table shows that the proposed algorithm using $b = 3$ and $B = 45$ outperforms[2] (in terms of overall sum of NOCC and DISC error) all the evaluated state-of-the-art local stereo matching approaches. Moreover, the Table also shows that our approach has an execution time dramatically reduced compared to SS (about 33 seconds for FSD with $b = 3$ vs 39 minutes required by SS according to [11]), being this time comparable to faster approaches (e.g VW and FBS). It wort observing that by using larger blocks (e.g. 5×5 and 7×7) the execution time further decreases, while the performance remains comparable to state-of-the-art approaches. Therefore, the block size parameter allows the user to trade accuracy for speed; this feature might be useful in certain application scenarios (e.g videosurveillance or robotics). Figure 4 shows the disparity maps concerned with Tsukuba for the examined algorithms. Finally, although we do not report experimental results due to the lack of space, we point out that compared to FBS and AW, the proposed FSD algorithm, deploying an explicit segmentation step, allows us to obtain accurate results also with greyscale images.

[1]Complete results adn disparity maps available at: www.vision.deis.unibo.it/spe/SPEresults.aspx

[2]Parameters of the FSD algorithm: $\gamma = 22.6$, TAD threshold $= 45$. Parameters for segmentation[13]: spatial bandwidth $S_b = 5$, range bandwidth $R_b = 2$

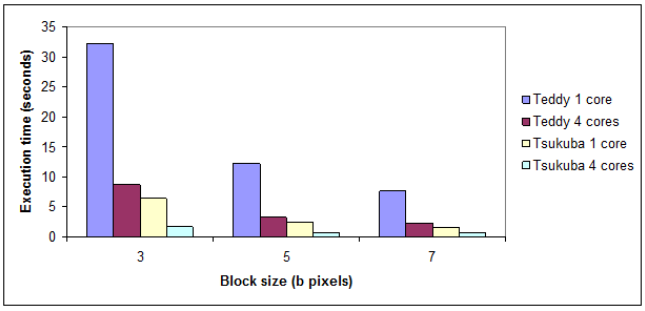

Fig. 3. Execution time for the proposed FSD approach on Teddy and Tsukuba with 1 core and 4 cores.

We also exploited for the proposed FSD algorithm the multi-core capabilities available in most modern processors using the OpenMP library[3]. Figure 3 reports the execution time for FSD in the three different configurations examined deploying 1 core and 4 cores on the Teddy and Tsukuba stereo pairs. Observing the figure, we can notice that FSD obtains significant speed-ups taking advantage of parallel execution and the execution time is reduced to few seconds.

5. CONCLUSIONS

In this paper we have proposed an accurate yet efficient algorithm for accurate stereo correspondence. Our proposal casts an effective algorithm based on segmentation within an efficient framework for weight computation enabling to obtain accuracy equivalent to state-of-the-art approaches with execution times comparable to those of faster, less accurate algorithms. A further speed-up was obtained exploiting multi-core capabilities available in most modern processors.

6. REFERENCES

[1] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2010.

[2] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. Jour. Computer Vision*, vol. 47, no. 1/2/3, pp. 7–42, 2002.

[3] F. Crow, "Summed-area tables for texture mapping," *Computer Graphics*, vol. 18, no. 3, pp. 207–212, 1984.

[4] M. Mc Donnel, "Box-filtering techniques," *Computer Graphics and Image Processing*, vol. 17, pp. 65–70, 1981.

[3]www.openmp.org

	Tsukuba			Venus			Teddy			Cones			Time / Overall NOCC+DISC
	NOCC	ALL	DISC	NOCC	ALL	DISC	NOCC	ALL	DISC	NOCC	ALL	DISC	
FSD$_{45(3)}$	**3.72**	5.69	**8.94**	**1.15**	2.80	**6.49**	**10.2**	19.4	**20.3**	**4.72**	15.3	**10.8**	32.28 sec / 66.32
FSD$_{45(5)}$	**4.02**	6.01	**9.72**	**1.29**	2.94	**9.35**	**10.5**	19.6	**21.1**	**4.75**	15.4	**11.5**	12.15 sec / 72.21
FSD$_{49(7)}$	**3.83**	5.72	**10.90**	**1.18**	2.79	**8.70**	**10.6**	19.7	**21.6**	**5.92**	16.2	**12.9**	6.6 sec / 75.61
FBS [8]	**2.95**	4.75	**8.69**	**1.29**	2.87	**7.62**	**10.71**	19.8	**20.82**	**5.23**	15.3	**11.34**	32 sec / 67.56
SS [7]	**2.15**	4.04	**7.22**	**1.38**	3.0	**6.27**	**10.5**	19.7	**21.2**	**5.83**	16.4	**11.8**	2358 sec / 67.06
SB [15]	**2.25**	2.86	**8.87**	**1.37**	2.31	**9.4**	**12.7**	20.1	**24.8**	**11.1**	19.2	**20.1**	2 sec / 100.65
AW [5]	**4.66**	6.68	**8.25**	**4.61**	6.18	**13.3**	**12.7**	21.6	**22.4**	**5.5**	16.0	**11.9**	1221 sec / 83.32
Rel [16]	**5.08**	6.94	**17.9**	**3.92**	5.5	**13.9**	**18.9**	27.0	**29.9**	**11.3**	20.7	**18.3**	803 sec / 121.2
VW [17]	**3.12**	4.86	**12.4**	**2.42**	3.87	**13.3**	**17.7**	25.9	**25.5**	**21.2**	29.6	**27.3**	26 sec / 94.68

Table 1. Measured accuracy and measured execution time for the proposed Fast Segmentation-driven (FSD) approach proposed with three configurations ($B = 45, b = 3$, $B = 45, b = 5$ and $B = 49, b = 7$), FBS [8], SS [7], SB [15], AW [5], REL [16] and VW [17] according to the Middlebury web site [14] and (in boldface) according to [11].

[5] K.J. Yoon and I.S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. PAMI*, vol. 28, no. 4, pp. 650–656, 2006.

[6] Asmaa Hosni, Michael Bleyer, Margrit Gelautz, and Christoph Rhemann, "Local stereo matching using geodesic support weights," in *ICIP*, 2009.

[7] F. Tombari, S. Mattoccia, and L. Di Stefano, "Segmentation-based adaptive support for accurate stereo correspondence," in *Proc. IEEE Pacific-Rim Symposium on Image and Video Technology (PSIVT'07)*, 2007.

[8] 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.

[9] Christian Richardt, Douglas Orr, Ian Davies, Antonio Criminisi, and Neil A. Dodgson, "Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid," in *ECCV (3)*, 2010, pp. 510–523.

[10] Margrit Gelautz Asmaa Hosni, Michael Bleyer, "Near real-time stereo with adaptive support weight approaches," in *3DPVT2010*, 2010.

[11] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda, "Classification and evaluation of cost aggregation methods for stereo correspondence," in *CVPR08*, 2008, pp. 1–8.

[12] M. Gong, R.G. Yang, W. Liang, and M.W. Gong, "A performance study on different cost aggregation approaches used in real-time stereo matching," *Int. Journal Computer Vision*, vol. 75, no. 2, pp. 283–296, 2007.

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

[14] D. Scharstein and R. Szeliski, "Middlebury stereo vision," http://vision.middlebury.edu/stereo/.

[15] M. Gerrits and P. Bekaert, "Local stereo matching with segmentation-based outlier rejection," in *Proc. CRV 2006*, 2006, pp. 66–66.

[16] S.B. Kang, R. Szeliski, and J. Chai, "Handling occlusions in dense multi-view stereo," in *Proc. CVPR 2001*, 2001, pp. 103–110.

[17] O. Veksler, "Fast variable window for stereo correspondence using integral images," in *Proc. CVPR 2003*, 2003, pp. 556–561.

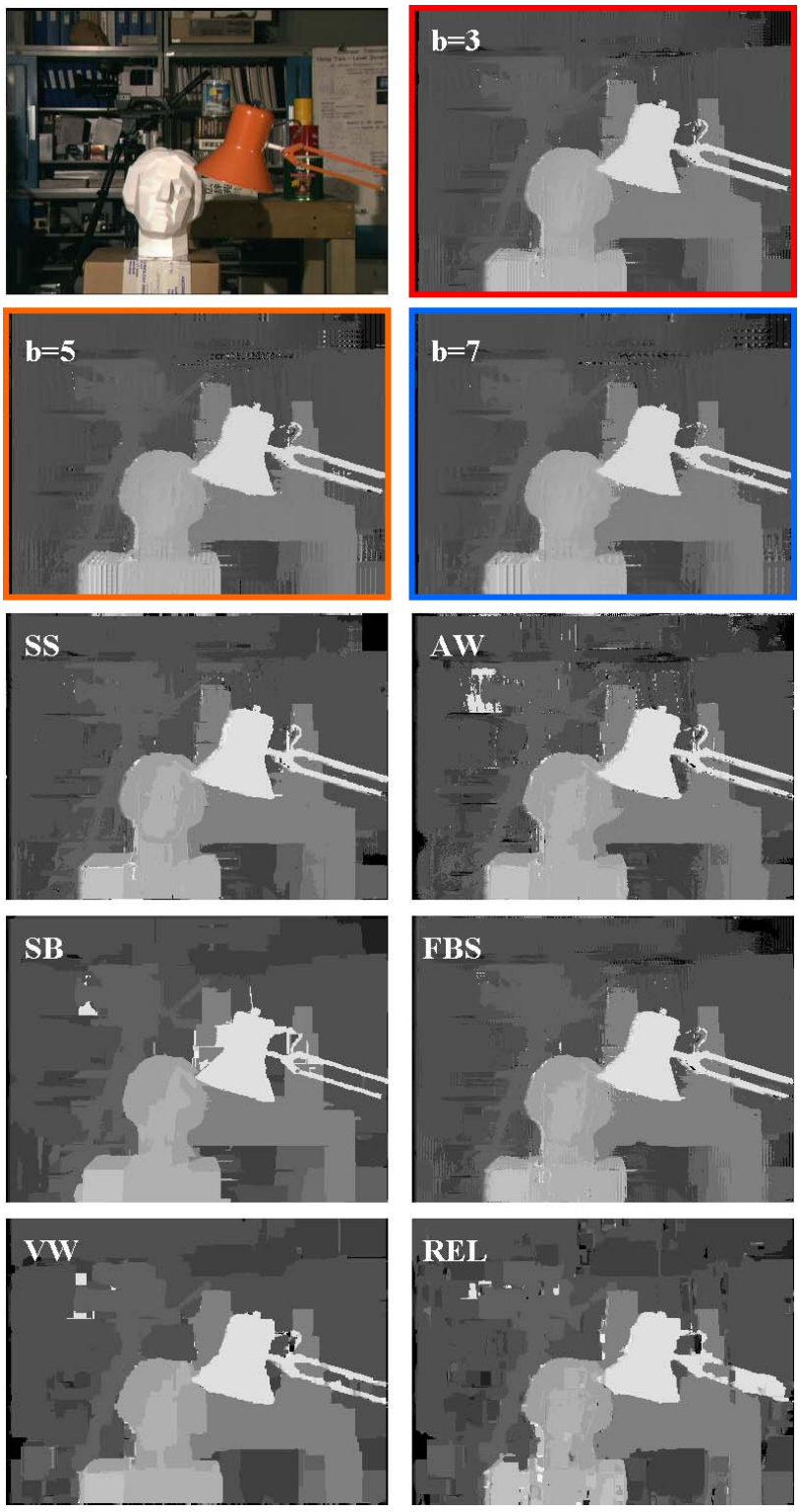

Fig. 4. Disparity maps concerned with the Tsukuba stereo pair for the proposed FSD method (with parameters $B = 45, b = 3$, $B = 45, b = 5$ and $B = 49, b = 7$), SS [7], AW [5], SB [15], FBS [8], VW [17] and REL [16]. **[Best viewed in electronic format]**