Near real-time stereo based on effective cost aggregation

Federico Tombari
federico.tombari@unibo.it

Stefano Mattoccia
stefano.mattoccia@unibo.it

Luigi Di Stefano
luigi.distefano@unibo.it

Elisa Addimanda
elisa.addimanda@studio.unibo.it

Department of Electronics Computer Science and Systems (DEIS), University of Bologna
Advanced Research Center on Electronic Systems (ARCES), University of Bologna

Abstract

Recent research activity on stereo matching has proved the efficacy of local approaches based on advanced cost aggregation strategies in accurately retrieving 3D information. However, accuracy is typically achieved at expense of computational efficiency, with best methods being far from meeting real-time requirements. On the other side, basic real-time local algorithms relying on a rectangular correlation window suffer from significant ambiguity along depth borders and untextured areas. This work proposes a novel local approach aimed at maximizing the speed-accuracy trade-off by means of an efficient segmentation-based cost aggregation strategy.

1. Introduction

Stereo matching algorithms are currently classified between local and global methods [15]. Typically, local methods are simple and fast [2, 5, 8, 10] while global ones can achieve a higher degree of accuracy in retrieving disparity information [12, 16, 22]. Recently, many stereo matching algorithm relying on image segmentation and aimed at improved accuracy have been proposed [1, 6, 9, 11, 12, 16–18, 21, 22, 24]. The great majority of these methods are global, and a subset of them [1, 12, 16, 22] represents currently the most accurate methods on the Middlebury Stereo Evaluation website [14], which is the standard benchmark platform for the stereo community. Anyway, the computational burden they require is far from meeting real-time or near real-time requirements.

As reported in a recent paper [19], local approaches that are state-of-the-art in terms of accuracy are based on segmentation [18] or adaptive weights [23], but are far from being computationally efficient. Indeed, apart from GPU or hardware-based implementation, typically only aggregation strategies based on sets of rectangular windows [2, 5, 10, 20] can afford real-time or near-real-time processing, this implying a notably reduced accuracy of retrieved disparities. Exceptions are represented by methods [6, 7], whose aggregation strategies rely on segmentation and that exhibit interesting trade-offs between accuracy and computational efficiency [19]. Moreover, between those methods for which a GPU implementation has been proposed [8], no one so far deploys segmentation.

The idea which motivates this work is to propose a novel aggregation strategy deploying segmentation aiming at high efficiency and at the same time as accurate as to improve the results of fast local stereo algorithms. This lead us to devise a method which improves significantly the performance-cost trade-off, yielding a level of accuracy comparable to that of segmentation-based methods and capable to meet near-real time processing requirements.

2. Cost aggregation strategy

Let I_r and I_t be respectively a reference and a target image of a stereo pair, and let $p \in I_r, q \in I_t$ be a pair of points at disparity d for which correspondence is being evaluated. The proposed aggregation scheme deploys a *variable support*, that is at each correspondence (p, q) the set of points around p and q on which the local similarity measure (or local cost) is computed depends on the local characteristics of the images. Similarly to most stereo matching algorithms deploying segmentation, the proposed aggregation strategy relies on the assumption that disparity varies smoothly within points lying on the same segment (this is true in practice especially if images are over-segmented). Thus, the idea is to shape the

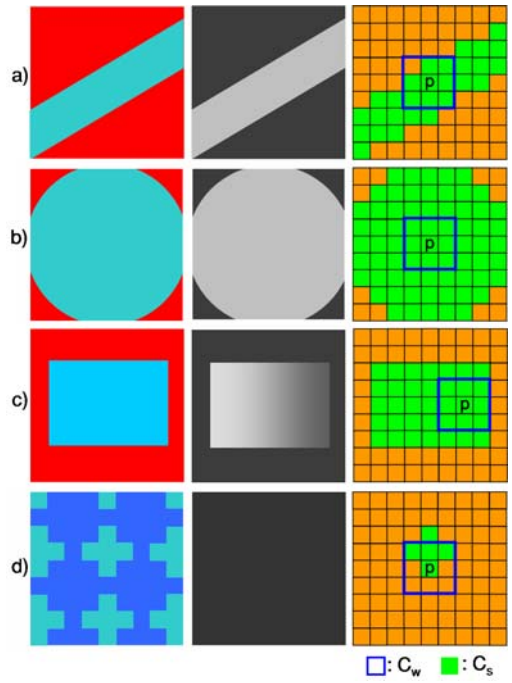

Figure 1. Examples of the behaviour of the proposed aggregation cost

variable support at each correspondence based on information derived from image color segmentation. This is achieved by computing for each correspondence (p, q) at disparity d an aggregation cost defined as:

$$C_s(p,q,d) = \sum_{p_i \in S_p} min\left(\delta\left(p_i, q_{i,d}\right), T_r\right) \quad (1)$$

where S_p is the segment on which p lies, $\delta(p,q)$ is the computationally efficient L_1 distance between the RGB components of p and q:

$$\delta(p,q) = |R_p - R_q| + |G_p - G_q| + |B_p - B_q| \quad (2)$$

and T_r is a fixed threshold. In practice, C_s represents the sum of the truncated absolute differences (TAD) over the segment on which p lies. The use of the truncation value T_r is a very basic M-estimator to enhance robustness toward outliers (in our experiments, T_r is set to 35).

C_s can be efficiently pre-computed by means of a single image scan for each possible disparity within the disparity range. Moreover, it tends to be notably accurate along depth borders since disparity edges tend to coincide with color edges on real images. Furthermore, within low-textured regions segments tends to be very big, which results in a high SNR and hence good robustness of C_s toward matching ambiguities. However, relying only on the segmentation cue might lead to mistakes, since C_s tends to assign the same disparity value to all points belonging to the same segment. This leads to mistakes for those points lying at slightly different depths from the majority of elements of a segment, e.g. on slanted surfaces. Furthermore, it also tends to decrease matching distinctiveness along highly-textured regions, where segments tend to be particularly small. Hence, we modify (1) to include also a corrective term based on a squared correlation window:

$$C_{aggr}(p,q,d) = \frac{C_s(p,q,d)}{n(S_p)} + \alpha \cdot \frac{C_w(p,q,d)}{(2r+1)^2} \quad (3)$$

where C_w is the TAD over the squared window $W_p(r)$ of radius r and centered on p:

$$C_w(p,q,d) = \sum_{p_i \in W_p(r)} min\left(\delta\left(p_i, q_{i,d}\right), T_r\right) \quad (4)$$

Cost C_{aggr} includes a normalization of the two terms C_s, C_w by the total number of points in, respectively, S_p and W_p. This is useful because, while the area of $W_p(r)$ is fixed, the number of points in each segment, $n(S_p)$, varies with p: thus, the normalization stage allows to weight equally each pixel included in C_{aggr}. It is important to point out that, thanks to the use of incremental schemes [4,13] the complexity of the calculation of term C_w amounts to only 4 elementary operations for each point and disparity, and it is independent on the choice of parameter r. Overall this results in a particularly efficient aggregation strategy.

Fig. 1 depicts graphically the behaviour of the proposed aggregation strategy in 4 different cases. In the figure, the first column shows the reference colour image, the second column shows the expected disparity map and the third column illustrates the behaviour of the proposed aggregation strategy. In particular, cost C_s assures that the variable support is shaped according to local chromatic cues. This is particularly useful along depth borders (case a) and within low-textured regions (b). Cost C_w, instead, adds a further weight for those points that are close to p (i.e. spatially more correlated). Generally the role of cost C_w is to increase the robustness of term C_s for those points violating the segmentation assumption, e.g. for bordering regions along slanted surfaces (case c). In addition, it is particularly effective along highly-textured regions (case d), where segments tend to reduce to a few pixels.

3. Further comments

The proposed aggregation strategy bears some resemblances with that proposed in [6], where for each

Algorithm	Accuracy	Tsukuba	Venus	Teddy	Cones	Art	Books	Dolls	Laundry	Moebius	Reindeer	MDS
Variable Windows	86.7	96.23	91.99	87.4	94.34	80.81	80.04	87.22	76.68	87.29	84.63	0.3
Proposed	86.4	97.04	96.47	89.33	95.08	78.72	81	85.64	74.89	84.88	80.48	18.9
Segmentation Based	83.3	94.3	93.92	90.35	92.69	76.22	79.86	84.75	61.7	81.09	77.75	5.9
Multiple Windows	82.1	94.42	95.82	85.46	91.18	72.68	78.31	81.36	64.23	80.79	76.66	2.7
Gradient Guided	79.4	92.99	87.66	80.46	88.03	72.17	72.86	83.93	61.48	76.15	78.27	3.2
Shiftable Windows	79.4	93.46	93.4	83.84	90.45	68.08	75.6	77.42	60.03	77.06	74.6	1.2

Table 1. Comparison of accuracy and MDS yielded by the proposed approach with respect to different state-of-the-art local stereo algorithms.

correspondence (p, q) the variable support is defined as the intersection between the points lying on the same segment as p and those belonging to the current correlation window. Nevertheless, if the working assumption that disparity varies smoothly within points lying on the same segment is verified, then the use of all points lying on S_p, rather than just those included in the current correlation window shall yield improved matching robustness and thus less ambiguity. Moreover, to avoid matching ambiguities due to few intersection points, method [6] requires the use of big correlation windows and the inclusion in the local cost with a smaller weight also of the remaining points in the window, which tends to increase inaccuracy. Furthermore, the efficient incremental implementation of the aggregation strategy proposed in [6] sacrifices accuracy for speed and tends to deteriorate the accuracy of the results. Conversely, our proposal can be directly implemented in an efficient way without any loss in accuracy. This results in significant improvements in accuracy and speed, as shown in next section.

The proposed aggregation strategy might be usefully deployed either by a local algorithm or as the initial stage of a global process based on e.g. Scanline Optimization [9] or Belief Propagation [16]. Moreover, it is interesting to note that this aggregation strategy could be symmetrically extended to include information also from the color segmentation of the target image I_t, rather than only that from I_r. This is not investigated here for lack of space, but there are hints that it would result in improved accuracy and lower computational efficiency.

Finally, in our implementation we use Mean Shift [3] to perform segmentation. This method yields accurate segmentation but is not extremely fast: overall in our experiments it accounts for a percentage between 40 and 80% of the total time. As a consequence, the proposed method could be further speeded-up using a faster segmentation method.

4. Experimental results

This section presents a comparison between the proposed method and other state-of-the-art aggregation strategies. Methods are evaluated within the same plain WTA (Winner-Take-All) stereo matching framework. In particular, as a term of comparison we selected state-of-the-art efficient aggregation strategies based on variable support [19], that is, *Segmentation Based* [6], *Shiftable Windows* [2], *Variable Windows* [20], *Multiple Windows* [10]. For what regards this last method, the version based on 9 correlation window is used as representative of the best accuracy-speed trade-off [10]. Methods [18, 23], though being state-of-the-art in accuracy among local algorithms [19], have not been included in our comparison since this paper focuses on real-time or near real-time methods while these methods are far from compelling these requirements (e.g. on the same platform and on Teddy, author's code of [23] runs in 18 minutes against 0.6 seconds of our approach).

For fairness of comparison, algorithms do not employ neither pre-processing nor post-processing such as consistency check and interpolation. Moreover, the local cost function is for all methods the TAD on RGB values, except for *Segmentation based* which deploys the Sum of Absolute Differences on RGB values plus a more complex M-estimator, as originally proposed in [6]. For what concerns the choice of parameters, all parameter values of the algorithms were optimally tuned on the dataset. In particular, for the proposed method the two parameters of the aggregation stage were set as $\alpha = 0.9$, $r = 6$. Finally, all algorithms were implemented in *C*, without any kind of optimization based, e.g., on SIMD instructions and tested on Intel Core Duo 2.14 GHz CPU.

Table 1 shows the results in terms of accuracy and computational requirements yielded by the evaluated algorithms on 10 stereo pairs belonging to the Middlebury dataset [14, 15]. Accuracy is calculated as the percentage of retrieved disparities whose difference with the

ground truth is ≤ 1. Evaluated disparities relate to all points of the disparity map except for occluded regions, since local WTA methods do not explicitly handle occlusions. As for computations, we report the millions of computed disparity per second (MDS) averaged on the whole tested dataset (for those algorithms deploying segmentation, the MDS includes also the overhead time spent for segmentation). To allow for a qualitative evaluation, all stereo pairs and disparity maps can be found on-line [1].

From the table it can be inferred that *Variable Windows* and the proposed approach outperform neatly all the other methods in terms of accuracy on the evaluated dataset, yielding comparable results. Nevertheless, for what concerns computations *Variable Windows* results to be the slowest method, while our approach is the fastest one, being almost two orders faster than *Variable Windows* and more than 3 times faster than *Segmentation based*. Hence it is clear that overall our approach yields the best accuracy-speed trade-off. It is interesting to point out that processing time for our method is around 0.2 s for *Tsukuba* (320×240, 16 disp., i.e. working at 5 fps.) and around 0.6 s for *Teddy* and *Art* (respectively 450×675, 60 disp. and 463×370, 75 disp.), thus achieving near real-time performance.

5. Conclusion and future works

An efficient aggregation strategy based on color image segmentation has been proposed. Overall, the proposed approach showed the capabilities to improve the accuracy of fast local methods and can be regarded as an interesting trade-off between accuracy and speed. Further optimizations of the proposed approach, e.g. based on GPU or SIMD instructions, might succeed in achieving even faster - perhaps real-time - performance.

References

[1] M. Bleyer and M. Gelautz. A layered stereo matching algorithm using image segmentation and global visibility constraints. *Jour. Photogrammetry and Remote Sensing*, 59:128–150, 2005.

[2] A. Bobick and S. Intille. Large occlusion stereo. *IJCV*, 33(3):181–200, 1999.

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

[4] F. Crow. Summed-area tables for texture mapping. *Computer Graphics*, 18(3):207–212, 1984.

[5] L. Di Stefano, M. Marchionni, and S. Mattoccia. A fast area-based stereo matching algorithm. *Image and Vision Computing*, 22(12):983–1005, 2004.

[6] M. Gerrits and P. Bekaert. Local stereo matching with segmentation-based outlier rejection. In *Proc. Conf. on Computer and Robot Vision*, pages 66–66, 2006.

[7] M. Gong and R. Yang. Image-gradient-guided real-time stereo on graphics hardware. In *Proc. Conf. 3D Digital Imaging and Modeling*, pages 548–555, 2005.

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

[9] H. Hirschmuller. Stereo vision in structured environments by consistent semi-global matching. *IEEE Trans. PAMI*, 30(2):328–341, 2008.

[10] H. Hirschmuller, P. Innocent, and J. Garibaldi. Real-time correlation-based stereo vision with reduced border errors. *IJCV*, 47:1–3, 2002.

[11] L. Hong and G. Chen. Segment-based stereo matching using graph cuts. In *Proc. CVPR*, volume 1, page 7481, 2004.

[12] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Proc. Int. Conf. on Pattern Recognition*, volume 3, pages 15–18, 2006.

[13] M. Mc Donnel. Box-filtering techniques. *Computer Graphics and Image Processing*, 17:65–70, 1981.

[14] *http://vision.middlebury.edu/stereo*. Middlebury Stereo Evaluation.

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

[16] J. Sun, Y. Li, S. Kang, and H. Shum. Symmetric stereo matching for occlusion handling. In *Proc. CVPR*, volume 2, pages 399–406, 2005.

[17] H. Tao, H. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *Proc. ICCV*, volume 1, pages 532–539, 2001.

[18] 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*, 2007.

[19] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Proc. CVPR*, 2008.

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

[21] Y. Wei and L. Quan. Region-based progressive stereo matching. In *Proc. CVPR*, volume 1, page 106113, 2004.

[22] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nistr. Stereo matching with color-weighted correlation, hierachical belief propagation and occlusion handling. In *Proc. CVPR*, volume 2, pages 2347 – 2354, 2006.

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

[24] C. Zitnick and S. Kand. Stereo for image-based rendering using image over-segmentation. *IJCV*, 75(1):49–65, 2007.

[1] Available at: *www.vision.deis.unibo.it/Stereo-FS.asp*