Fast Edge-Aware Cost Aggregation for Stereo Correspondence

Bingrong Wang, Xianghui Bai, Zhiming Tan, Akihiro Higashi
Fujitsu R&D Center
29/F, Kerry Parkside, 1155 Fangdian Road, Shanghai, China
wangbingrong@cn.fujitsu.com

Abstract

Stereo correspondence computes disparity map needed by many high-level computer vision tasks. Recently local stereo approaches show their ability matching global while keeping efficiency. But it still deserves further research for higher performance. In this paper, we propose a fast edge-aware cost aggregation strategy. It constructs 1D support window based on spatial and gradient information of input image, and uses a two-pass scheme to smooth the disparity space image. We also propose an O(1) implementation independent of window size for this scheme. A similar edge-aware median filter is proposed for the post-processing step. According to the evaluation results on the Middlebury stereo dataset, our approach outperforms the state-of-the-art local methods in accuracy and efficiency.

1. Introduction

Stereo correspondence as an important vision problem estimates disparities from a given stereo image pair. Lots of work on the topic has been surveyed and evaluated in [1][2]. According to [1], most stereo algorithms consist of four steps: matching cost computation, cost aggregation, disparity computation and disparity refinement. They can be categorized into two major classes: local methods and global methods. Unlike most computationally expensive global stereo methods, local methods are usually efficient and easy to implement.

The major challenge in local stereo is to find an appropriate support window. An ideal support window should be large enough to capture sufficient intensity variation for handling textureless regions. At the same time, the window should be small enough not to include pixels of different disparities in order to avoid the well-known edge fattening effect at disparity discontinuities. In order to overcome the challenge, many local stereo methods have been proposed.

In the early development stage the multiple-window methods [3], which selected the lowest-cost support window from a set of pre-defined windows, and variable-window methods [4][5], which computed an optimal support window for each pixel, were proposed. But the results suffer from the edge-fattening problem.

The filter-based methods were introduced by Yoon and Kweon [6], which improved the accuracy much. The method is actually an application of the bilateral filter [7] in the domain of stereo correspondence. The idea is that pixels having a color similar to the center pixel are likely to lie on the same object, and therefore have similar depth (disparity). A pixel inside the support window receives a high support weight if it is close in both color and spatial distance to the central pixel of the window. This strategy considerably reduces the edge fattening problem when using large window sizes and accordingly leads to better quality results. But Yoon and Kweon used a naïve implementation of the bilateral filter, which was slow and diminishes the runtime advantage of local over global methods. Richard et al. [8] realized this shortcoming and suggested an approximate but fast (real-time) implementation of the filter. However, their solution could not even get close to the state-of-the-art results.

Recently, Rhemann et al. [9] utilized the image guided filter [10] to filter cost volume and achieved state-of-the-art results among local methods. Because the guided filter performs a first-order local linear modeling, it is computationally much faster than bilateral filter.

In this paper, we will present an efficient stereo correspondence approach. It has three contributions:

(1) We propose a fast edge-aware cost aggregation based on an iterative two-pass 1D filter. It performs cost aggregation in succession on horizontal and vertical windows.
(2) We also propose a fast implementation for the cost aggregation so that its computation is independent of the window size.
(3) We use similar ideas to design an edge-aware median filter for post-processing.

We will describe the details in the following sections.

2. Algorithms

The outline of our stereo matching algorithms is shown in Figure 1. We select the left image as target in this paper.

We use a similar method with that used in [9] for matching cost computation because it is robust to illumination changes. The difference is that we add truncated absolute difference of the vertical gradient:

$$\begin{aligned} C(x,y,d) = {} & \alpha \cdot \min(||I_R(x-d,y) - I_L(x,y)||, \tau_c) + \\ & \beta \cdot \min(\|\nabla_x I_R(x-d,y) - \nabla_x I_L(x,y)\|, \tau_g) + \\ & (1-\alpha-\beta) \cdot \min(\|\nabla_y I_R(x-d,y) - \nabla_y I_L(x,y)\|, \tau_g) \end{aligned} \quad (1)$$

Here, ∇_x is the horizontal gradient and ∇_y is the vertical gradient. The parameters α and β balances the color, the horizontal gradient and the vertical gradient terms, and τ_c, τ_g are truncating values which control the limit of the matching cost to avoid the later aggregation is polluted by noisy points.

Then the proposed cost aggregation approach is used on the initial disparity space image *C(x,y,d)*. The details will be presented in Section 2.1. And an *O*(1) implementation of cost aggregation independent of window size is proposed in this section. In the next step, the Winner-Takes-All (WTA) strategy is applied for the dis-

parity selection.

The same process is carried out on the right image in order to detect occlusions. Next we designed a post-processing procedure to optimize the disparity map. The process is described in Section 2.3.

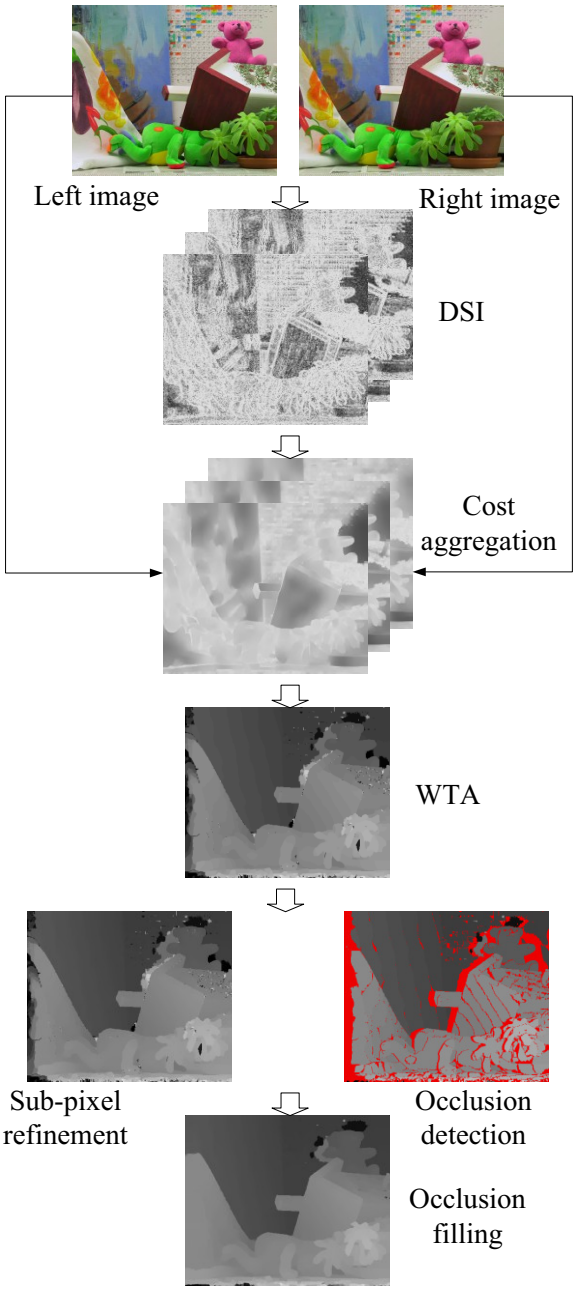

Figure 1. Outline of the proposed stereo correspondence algorithm. We can obviously see that our cost aggregation method filters the initial disparity space image (DSI) with edges preserved well. And the post-processing process including sub-pixel refinement, occlusion detection and filling effectively optimize the disparity map.

2.1. Proposed fast edge-aware cost aggregation

As we briefly mention in the introduction, the key of cost aggregation is finding a well-suited support window to contain as many pixels as possible at the same depth. However, without depth information beforehand, the support region for a pixel can only be adequately derived from the raw input images. Just like gradient is used to detect edge, we can judge whether two pixels stay in the same object, or at the same depth by the accumulation of the gradients of the pixels between them. Under that idea, we propose a local support window construction method.

Unlike the most existing window-based methods fixing the window size (e.g. SSD [1], bilateral filter [6], guided filter [9], etc.), our method uses a shape-adaptive window to reliably capture local image structure. The process is based on an iterated two-pass 1D filter, as shown in Figure 2. 1D filter has two advantages: 1) because considering horizontal or vertical bounds every time, it can adapt the local shape better; 2) It has a lower computation complexity compared with 2D filter.

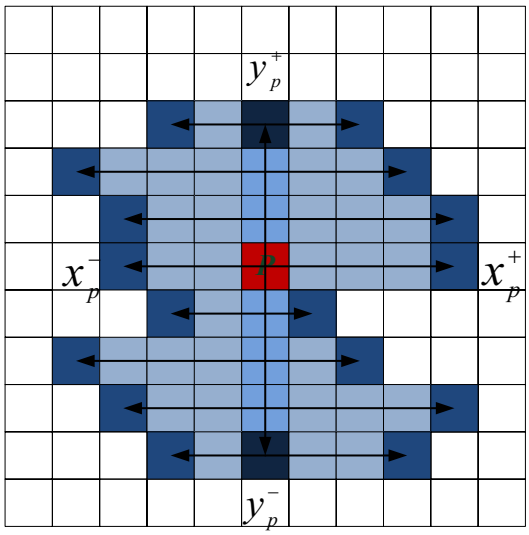

Figure 2. Two-pass cost aggregation for the center pixel p. The first pass filters the cost in the horizontal direction. And the result is filtered by the second pass in the vertical direction. All those pixels influencing the cost of the pixel p are marked in color.

We first look at the horizontal pass. In x direction, we decide the horizontal coordinate x_p^- of the left bound of a pixel (x_p,y_p) as

$$x_p^- = \min\{x^- \mid 0 \le x^- < x_p \text{ and } T^-(x^-, x_p, y_p) \le r\}, \quad (2)$$

where T^- is a judgment function whether two pixels lie at the same depth and r controls its confidence level. T^- is defined as

$$T^-(x^-, x_p, y_p) = (x_p - x^-) + \sigma \sum_{u=x^-+1}^{x_p} ||\nabla_x I(u, y_p)||. \quad (3)$$

Here, T^- consists of two parts: spatial distance and accumulation of absolute values of gradients. The gradients of different channels will be computed respectively and added together. The parameter σ balances the distance and the gradients' sum terms.

The horizontal coordinate x_p^+ of the right bound is computed by the corresponding formula:

$$x_p^+ = \min\{x^+ \mid x_p < x^+ < w \text{ and } T^+(x^+, x_p, y_p) \le r\}, \quad (4)$$

where w is the image width. T^+ is defined as

$$T^+(x^+, x_p, y_p) = (x^+ - x_p) + \sigma \sum_{u=x_p+1}^{x^+} ||\nabla_x I(u, y_p)||. \quad (5)$$

Thus the aggregated cost is average over the horizontal segment:

$$C_h(x_p, y_p, d) = \frac{\sum_{u=x_p^-}^{x_p^+} C(u, y_p, d)}{x_p^+ - x_p^- + 1}. \quad (6)$$

The vertical cost aggregation is carried out on the result of the horizontal aggregation. The lower bound y_p^- and the upper bound y_p^+ are calculated in similar methods. Thus a two-pass filter is completed. Figure 3 shows four examples of support windows. From the figure, we can see that edges of objects are preserved well.

Figure 3. Samples of edge-aware support windows. The center pixel is marked in blue rectangle. The red represents support window.

In order to have better filtering effects, our two-pass aggregation can be done iteratively. Our experiments iterate the process twice, which trades off accuracy and efficiency well. Note that the second aggregation should have a smaller window size by decreasing the confidence level *r* because the disparity space image becomes smooth after the first aggregation. We let *r* for the second iteration be half of the first one: $r_1 = r_0 / 2$.

Instead of summing raw matching costs in (6) directly, we propose an efficient *O*(1) implementation to accelerate the aggregation over irregularly shaped regions. Given the pixelwise raw matching cost *C*(*x*,*y*,*d*), we first build a horizontal integral image $S_h(x,y,d)$:

$$\begin{aligned} S_h(x,y,d) &= \sum_{u=0}^{x} C(u,y,d) \\ &= S_h(x-1,y,d) + C(x,y,d) \end{aligned}. \quad (7)$$

Here $S_h(x,y,d)$ can be iteratively computed from $S_h(x\text{-}1,y,d)$ with only one addition. Thus we can rewrite the equation (6) into

$$C_h(x_p, y_p, d) = \frac{S_h(x_p^+, y_p, d) - S_h(x_p^- - 1, y_p, d)}{x_p^+ - x_p^- + 1}. \quad (8)$$

Obviously our new filtering formula has nothing to do with the support window. When $x_p = 0$, $C_h(x_p,y_p,d)=0$. The vertical aggregation can be of course computed by this fast strategy, too.

2.2. Post-processing with proposed edge-aware median filter

Our post-processing does two things: 1) obtaining a sub-pixel disparity map; 2) handling occluded pixels.

As shown in Figure 1, on the one hand we apply quadratic polynomial interpolation on the disparity map of the left image for sub-pixel refinement, as shown in [11]. On the other hand, we mark a pixel in the left disparity map as occluded by left-right consistency checking.

For the unoccluded pixels, the sub-pixel disparities are their final values. For the occluded pixels, to fill them, we first assign the lowest disparity value of the spatially closet non-occluded pixels which lie on the same scanline to them, obtaining an initial filling result. Then an edge-aware median filter is applied on the initial filling result to remove the streak-like artifacts in the disparity map. The edge-aware median filter is similar with the one used in the cost aggregation. The difference is that the median value is selected as the new disparity in the window instead of averaging them:

$$d_h(x_p, y_p) = \underset{x_p^- \le x \le x_p^+}{\text{med}} \left(d(x, y_p) \right). \quad (9)$$

Similarly we also take the median value in the vertical direction. In order to get a better result, we set a different value for the parameter σ from that in the cost aggregation. Compared with the weight median filter used in [9], our edge-aware median filter has lower computation complexity and better performance. At last, a normal median filter is used to obtain the final disparity map.

3. Experimental Results

We tried different parameter settings and decided to use the following values for all the experiments: $\{\alpha, \beta, \tau_c, \tau_g, r_0\} = \{0.11, 0.75, 9/255, 2/255, 80\}$. The parameter σ for cost aggregation is 150 and the parameter σ for post-processing is 200. The larger σ brings the smaller support region.

We implemented our method on the graphics card using CUDA. All experiments were conducted on an nVidia GeForce GTX560 Ti graphics with 1G display memory whose CUDA core number is 384.

We evaluated our algorithms on the Middlebury stereo benchmark [1]. Our approach gives quite excellent results ranking 8th out of 140 methods at the time of submission when the error threshold is set to 5.0. And more importantly as far as we know, our method is the best local filter-based stereo method outperforming the original implementation of [6]. Table 1 gives the comparisons between our method and several other local stereo methods. And Figure 4 shows our results and the difference against the ground truth maps.

The average time of our method and its competitors are also shown in Table 1. Runtime data of the methods are taken from [9]. The run time for the method CostFilter [9] was acquired on GeForce GTX480 graphics card with 480 CUDA cores, and for the other methods on Quadro FX5800 with 240 CUDA cores. Considering the distinction of cores, we claim that our approach is the fastest one among these local methods.

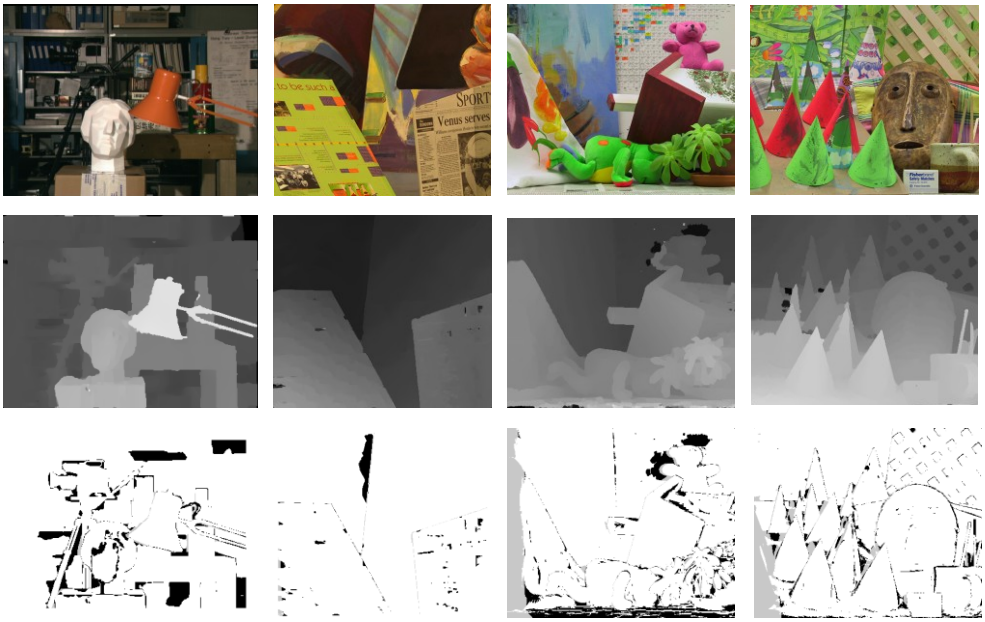

Figure 4. Results on Middlebury dataset. The first row is the left input image (Tsukuba, Venus, Teddy, and Cones, respectively). The second row shows the results computed by our algorithm. The third row shows a comparison against the ground truth by plotting disparity errors larger than 0.5 pixels.

Table 1. Rankings on Middlebury on-line database.

Method	Rank	Avg. Error (%)	Error non-occluded pixels (%)				Avg. Runtime (ms)
			Tsukuba	Venus	Teddy	Cones	
Ours	8	10.9	11.5	2.7	9.75	5.23	52
CostFilter[9]	26	12.8	11.7	6.43	18.1	13.7	65
GeoSup[12]	55	15.9	23.1	7.11	20.4	15.0	16000
DCBGrid [8]	86	18.5	22.8	3.97	24.0	18.2	95.4
AdaptWeight[6]	89	18.1	18.8	8.40	23.9	19.7	8550

4. Conclusions

We have presented a novel efficient cost aggregation method for local stereo matching. It smoothes the disparity space image based on an iterative two-pass 1D filter. The filter has shape-adaptive window size constructed by spatial and gradient information of the original input images. Theories and experiments prove its edge-aware property and the success of cost aggregation. We also propose a constant time implementation for this cost aggregation independent of support window size. We use a similar idea to design an edge-aware median filter for post-processing.

Experimental results on the Middlebury stereo benchmark show our method outperforms the other local stereo matching methods in accuracy and efficiency. Because it is both accurate and fast, it will be useful in many circumstances.

References

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vision (IJCV)*, vol. 47, no. 1, pp. 7-42, May 2002.

[2] M.Z. Brown, D. Burschka, and G.D. Hager, "Advances in computational stereo," *IEEE Trans. Pattern Anal. Mach. Intell (PAMI).*, vol. 25, no. 8, pp. 993-1008, 2003.

[3] A. Fusiello, V. Roberto, and E. Trucco, "Effficient stereo with multiple windowing," *in IEEE Intl. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 858-863, 1997.

[4] Y. Boykov, O. Veksler, and R. I. Zabi, "A variable window approach to early vision," *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, vol. 20, no. 12, pp. 1283-1294, 1998.

[5] O. Veksler, "Fast variable window for stereo correspondence using integral images," *in Proc. IEEE Intl. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, pp. I-556-I-561, 2003.

[6] K.J. Yoon and I.S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, vol. 28, no. 4, pp. 650-656, 2006.

[7] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *in Proc. IEEE Intl. Conf. on Computer Vision*, pp. 839-846, 1998.

[8] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. Dodgson, "Real-time spatiotemporal stereo matching using dual-cross-bilateral grid," *in Proc. IEEE European Conf. Computer Vision (ECCV)*, pp. 510-523, Feb. 2010.

[9] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *in Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.(CVPR)*, Mar. 2011.

[10] K. He, J. Sun, and X. Tang, "Guided image filtering," *in Proc. IEEE European Conf. Computer Vision (ECCV)*, 2010.

[11] Q. Yang, R. Yang, J. Davis, D. Nister, "Spatial-depth super resolution for range images," *in Proc. IEEE Conf. Comput. Vis. and Pattern Recognit. (CVPR)*, 2007.

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