Stereo Matching Method Using Slant-based Cost Aggregation

Daeyoon Jeon[1], Sangyoon Lee[1], and Hyunki Hong[,2]*

[1] *Dept. of Digital Imaging, Integrative Technology, Chung-Ang University / Seoul, Korea*
[2] *School of Integrative Engineering, Chung-Ang University / Seoul, Korea*
**Corresponding Author (honghk@cau.ac.kr)*

Abstract: Stereo matching methods based on color segmentation usually obtain the initial matching cost for each pixel and interpolate the disparity value with the local gradient of image patches. It is assumed that the image regions may be approximated with local surfaces in the initial matching cost aggregation. However, color segmentation and disparity plane estimation have high computational loads. This paper presents an efficient real-time stereo matching method to improve initial cost aggregation. The initial matching costs are computed using the direction of arbitrary slopes in matching cost volume without color segmentation and plane estimation procedures.

Keywords: Stereo Matching, Slanted Aggregation, Real-Time Stereo Vision

Received Aug. 23, 2016; accepted for publication Sep. 08, 2016; published online Nov. 30, 2016. DOI: 10.15323/techart.2016.08.3.4.45 / ISSN: 2288-9248.

1. Introduction

Stereo vision is a technique that can be used to estimate the 3D depth of a real scene in a manner similar to the structure of the human visual system. The goal is to determine a precise disparity that indicates the difference in the location of the corresponding pixels [1]-[10].

Stereo algorithms are described according to four individual components in stereo matching: matching cost computation, cost aggregation, disparity computation, and disparity refinement.

This paper presents an efficient cost aggregation method in the assumption that the stereo input images are composed of local surface regions. In the proposed aggregation method, the matching costs are integrated along arbitrary slopes in cost space, which is implemented with shift operations in CUDA memory architecture. On the contrary, previous disparity based stereo matching methods have high computational load because color segmentation and plane estimation procedures are needed[4],[6]. In addition, an input image is down-sampled for real-time performance and the final disparity map is up-sampled using control points with reliable disparity value. Yoon proposed a method adjusts the support-weights of the pixels in a given support window based on color similarity and geometric proximity to reduce the image ambiguity [7]. In addition, real-time stereo matching methods based on CUDA architecture have been introduced actively [8],[9].

2. Proposed Method

A. Initial Matching Cost and Support Region

The initial matching costs of the stereo view are computed by combining the absolute difference (AD) and census transform methods as shown in Equation (1) [3]. The AD method calculates the absolute difference between the brightness values of interest pixel p in the reference image (left) and in the target image (right) using Equation (2). Census transform compares the brightness value of the neighborhood pixel with that of an interest point and the relative brightness distribution is converted into binary strings. More specifically, if the brightness value of the neighborhood pixel is higher than the counterpart of the central pixel, then it returns 1. Otherwise, it returns 0. The bit-string on the left reference image is computed, and the bit strings on the right image are calculated within the search range of the maximum disparity d. In addition, it can efficiently calculate the similarity of two bit-strings through hamming distance. In a previous study [3], initial matching costs by the AD and census methods were combined with exponential functions. The proposed method computes the initial matching cost $C(p,\ d)$ with simple multiplication and minimal operation for computational efficiency as shown in Equation (1) and (2). $C_{AD}(p,d)$ and $C_{Census}(p,d)$ denote cost-volume values by AD and Census methods. Here, λ_{AD}, w_{AD}, λ_{Census} and w_{Census} are set to 10, 0.55, 10, and 0.45, respectively.

$$C(p,d)=\frac{\min(C_{AD}(p,d),\ \lambda_{AD})}{\lambda_{AD}}*w_{AD}+\frac{\min(C_{Census}(p,d),\ \lambda_{Census})}{\lambda_{Census}}w_{Census} \quad (1)$$

$$C_{AD}(p,d)=\frac{1}{3}\sum_{i=b,g,r}\left|I_i^{Left}(p)-\ I_i^{Right}(p-d)\right| \quad (2)$$

In the initial matching cost computation, the brightness values of neighbor pixels of an interest point are examined [2, 3]. The support region with similar color distribution as the interest point is built within the fixed region. The lengths $\{h_p^-,\ h_p^+,\ v_p^-, v_p^+\}$ of the vertical and horizontal arms are determined by the center point p. The endpoint p_1 of the horizontal and vertical directions with respect to the given pixel p is determined by Equations (3)–(5).

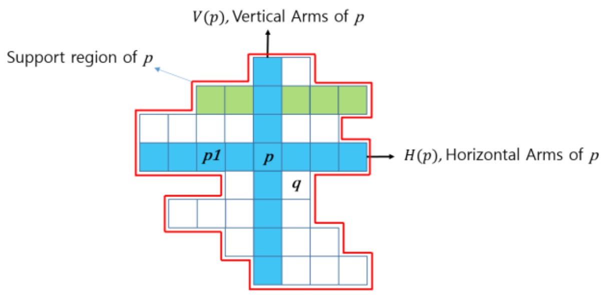

Fig. 1. Support Regions and Arm Parameters [3]

$$D_c(p_1, p) < \tau, D_c(p_1, p+1) < \tau_1,$$
$$D_c(p_1, p) = \max_{i=b,g,r} |I_i(p_1) - I_i(p)| \quad (3)$$

$$D_s(p_1, p) < L_1, \qquad D_s(p_1, p) = |p_1 - p| \quad (4)$$

$$D_c(p_1, p) < \tau_2,\ if\ L_2 < D_s(p_1, p) < L_1 \quad (5)$$

Equation (3) computes the color distance $D_c(p_1, p)$. The endpoint p_1 of the given pixel p is determined until it is lower than the given threshold value (τ) of the color distance. Equation (4) denotes the spatial distance between the pixels and the maximum length of the "L_1" arm is considered. The additional thresholds (arm length L_2 and threshold value τ_2) are included as shown in Equation (5). By examining neighbor pixels in both vertical and horizontal directions, we can build an adaptive support region, which is represented as Equation (6) [3].

$$U_L(p) = \bigcup_{q \in V_L(p)} H_L(p) \quad (6)$$

$$U(p, d) = U_L(p) \bigcap U_R(p - d) \quad (7)$$

$$C_1(p, d) = \frac{1}{\|U(p, d)\|} \sum_{q \in U(p,d)} C_0(q, d) \quad (8)$$

Although both color and spatial examination procedures were employed, the support regions may include another region with different disparity values. Thus, the support regions in the reference image and target image are intersectioned to determine the overlapping support regions between two views. Equation (8) computes the matching cost C_1 by aggregating initial matching cost C_0 in overlapped support region with equation (7). As shown in Figure 1, p and q pixels in Equation (7) and (8) represent an anchor pixel and any neighboring pixel in the support region of p, respectively.

B. *Slant-based Cost Aggregation*

In general, color segmentation-based stereo matching assumes that the image regions with similar color have the same disparity values [4]. Figure 2 shows the disparity results by [4]: Venus (left) and Cones (right) ~~image pairs~~ pair images from Middlebury Benchmark[4]. In the third row, the left-side region of the cup has the same disparity because it is a plane in the forward direction with respect to the camera. Likewise, when the plane is parallel to the camera and faces the forward direction, it is called fronto-parallel [4].

In Figure 2, the disparity map results presented in the first row were obtained based on the assumption that image surfaces are composed of fronto-parallel planes. The disparity maps in the third row were computed using an image surface with real disparity gradient information. In order to obtain more accurate disparity information, the 3-D slant of the object's plane region must be calculated. However, in the example of the cone (in the fourth row), the slanted plane region may have different disparity values for all pixels in the support region. The local surface region with the same disparity value may not be detected using the color segmentation procedure.

The proposed method does not directly compute the slopes of the image surface, but determines a range of stereo matching cost aggregates in the disparity space. Previous methods aggregate the initial matching costs of the support region of the interest pixel in the horizontal direction for every disparity level d. In other words, matching costs at the disparity level d of the surface disparity gradient have to be integrated along the scan direction (x axis). Our method performs the initial matching cost aggregation procedure along several typical slopes representing candidate image surfaces in Fig. 3. In CUDA memory architecture, we can integrate the matching cost volume efficiently by accessing adjacent memory address of all active threads.

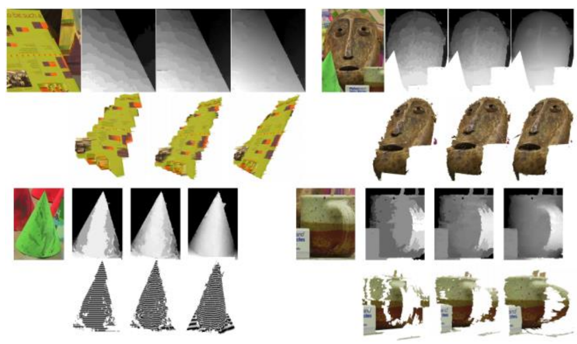

Fig. 2. Disparity Results by [4]: Venus and Cones

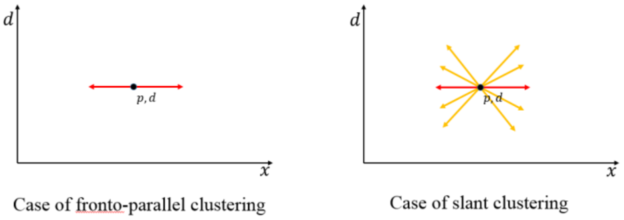

Fig. 3. Initial Matching Cost Aggregation Directions

The proposed system examines interest disparity levels that represent candidate image surfaces instead of every image plane with all surface gradients. As shown in Fig. 4, the matching cost integration is performed via three steps:

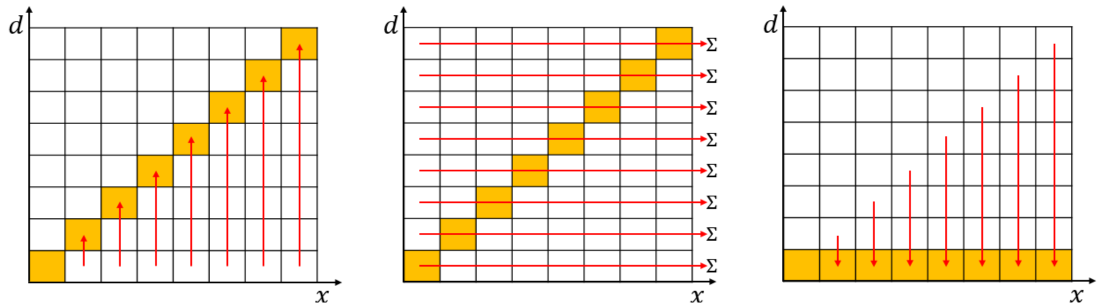

Fig. 4. Matching Cost Integration Procedure: Cost Volume Pushing, Integration, and Pulling

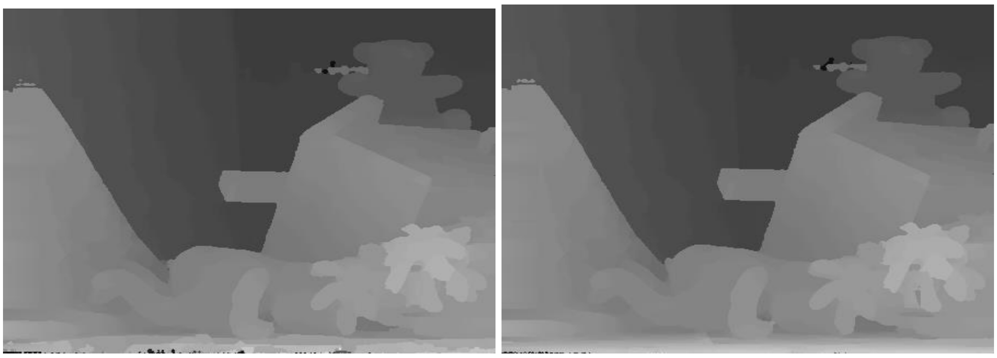

Fig. 5. Comparison of Disparity Map Results by Slant-based Aggregation; Before (left) and After (right)

cost volume pushing, integration, and pulling. The arbitrary slope directions in the matching cost space are pre-set and the computed disparity level positions are marked in yellow. The yellow marked matching costs are pulled with simple memory address subtraction, are integrated, and then the cost values are returned to their original positions.

In the experiment, slope adjustment offsets are set to seven directions: 0, 1, 6, 12, -1, -6, and -12 units. Here, 0 means that the interest surface is fronto-parallel. The offset is determined by the pair-wise pair based on the pixel of interest along the disparity level. For example, if the offset is 6, *d* increases by one step while *x* increases by as much as 6. The aggregation procedure is performed in both horizontal and vertical directions. In this way, the proposed method aggregates the initial matching costs within a random offset range, rather than finding the real slopes of image surfaces.

Plane based stereo matching methods have to estimate the planar surface with a constant gradient. However, the proposed method can achieve cost aggregation along some directions without plane estimation. The minimum integration values is obtained along the slope similar to the real image surface. That means the slope with minimum aggregated value can be determined as the slope of real surface.

In Equation (9), we can compute the final minimum cost in the aggregation by examining every slope offset for *p* and *d*. In Figure 5, the disparity map results of the floor and other planar areas have been improved using the proposed method.

$$C_{final}(p,d) = \underset{s \in offset}{\operatorname{argmin}}\, C_s(p,d) \qquad (9)$$

C. Optimization

After slant-based aggregation, unreliable disparity values were obtained due to occlusions, repetitive structures, and texture-less regions. These regions can be determined through a left-right consistency (LRC) check with Equation (10). The edge propagation (EDP) method was then employed as an optimization process with color continuity and edge information [10]. The EDP method propagated the disparity values to the peripheral regions, considering the color differences of pixels and the edge costs of regions as Equation (11). When the cost value of an adjacent pixel is small, the disparity values of neighboring pixels are propagated. Here, pl is the adjacent pixel in the direction being propagated and $\Delta(p, pl)$ is the average RGB value of the interest pixel with neighborhoods. As the cost difference between pixels increases, Equation (11) returns a value close to 0. The final cost volume is sequentially propagated in the corresponding direction by Equation (12). Here, $C_{new}(p, d)$ is the seed cost volume, and $C'_{new}(p, d)$ is the final slant-based aggregation cost volume.

$$D_L(x, y) = D_R(x - D_L(x, y), y) \quad (10)$$

$$\alpha(\mathrm{p}, \mathrm{pl}) = \exp\left(-\frac{1}{\sigma_r} - \frac{\Delta(p, pl)}{\sigma_s}\right) \quad (11)$$

$$C'_{new}(p, d) = C_{new}(p, d) + \alpha(\mathrm{p}, \mathrm{pl}) \cdot C'_{new}(pl, d) \quad (12)$$

The proposed method employs WTA(WinnerTakesAll) to determine a disparity value in the cost volume, and then the LRC check for outlier detection. In addition, unstable disparity values are refined by iterative region voting and proper interpolation procedures [3].

Table 1.
Matching error rates with the number of slant directions

	Tsukuba			**Venus**			**Teddy**			**Cones**		
	nocc	all	disc	nocc	all	disc	nocc	all	disc	nocc	all	disc
3	2.95	3.31	8.83	0.26	0.35	3.29	2.72	5.95	8.31	3.17	9.06	8.19
5	3.23	3.70	9.27	0.27	0.39	3.36	2.90	5.93	8.76	2.18	7.30	6.46
7	3.10	3.56	9.48	0.33	0.48	4.07	3.14	5.93	9.13	2.13	7.07	6.32

Table 2.
Middlebury benchmark test results

	Tsukuba			**Venus**			**Teddy**			**Cones**			Aver.
	nocc	all	disc	nocc	all	disc	nocc	all	disc	nocc	all	disc	
AD-Census[3]	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.90	2.42	7.25	6.95	3.97
Proposed Method	**3.23**	-	-	**0.27**	-	-	**2.90**	-	-	**2.20**	-	-	**2.16**
PlaneFitBP[4]	0.97	1.83	5.26	0.17	0.51	1.71	6.65	12.10	14.7	4.17	10.70	10.60	5.78
AW[6]	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.3	18.6	3.97	9.79	8.26	6.67
Real-time BFV[7]	1.71	2.22	6.74	0.55	0.87	2.88	9.90	15.00	19.5	6.66	12.60	13.40	7.65
Real-time GPU[8]	2.05	4.22	10.6	1.92	2.98	20.30	7.23	14.40	17.6	6.41	13.70	16.50	9.82
DCB grid[9]	5.90	7.26	21.0	1.35	1.91	11.20	10.50	17.20	22.2	5.34	11.90	14.90	10.90

3. Experimental Results

The experiment was carried out using a PC with Intel(R) Core(TM) i7-4790 CPU @3.60 GHz, NVIDIA Geforce GTX 980Ti graphics card, and VS 2013, OpenCV 2.4.10 development environment. The proposed algorithm was implemented on a GPGPU-based CUDA 6.5 platform with multi-threads and parallel programming. The performance of previous methods is listed with their original rank, as reported in the Middlebury benchmark [5]. The proposed method provides better results compared to all other methods.

Table 1 shows the disparity map results according to the number of slant directions. In the case of "cones" image with many image planes, more accurate disparity results were obtained by considering more slant directions. Table 2 compares the disparity error results according to the Middlebury benchmark. Since each image has a different scene structure, the optimal surface number was different in every scene. However, in the case of five slant directions, we obtained the best matching performance on average. The input image resolution was 512×384 and the maximum disparity distance was set to 64. The proposed method computed the final disparity map in 95 ms (10.52 frames/s) without further program optimization.

4. Conclusion

This paper presents a novel stereo matching method using slant direction in cost space. The proposed method can achieve cost aggregation along some directions without plane estimation and color segmentation. This process is implemented efficiently in CUDA memory architecture. The experiment results show that the proposed method can obtain better matching performance than previous methods.

References

[1] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *Int. J. Comput. Vision*, vol. 47, pp. 7–42, 2002.

[2] K. Zhang, J. Lu, and G. Lafruit, "Cross-Based Local Stereo Matching Using Orthogonal Integral Images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073–1079, 2009.

[3] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On Building an Accurate Stereo Matching System on Graphics Hardware," *In Computer Vision Workshops*, pp. 467–474, 2011.

[4] M. Bleyer, C. Rhemann, and C. Rother, "PatchMatch Stereo-Stereo Matching with Slanted Support Windows," *In BMVC*, vol. 11, pp. 1–11, 2011.

[5] Middlebury Stereo Vision, http://vision.middlebury.edu/stereo/.

[6] Q. Yang, C. Engels, and A. Akbarzadeh, "Near Real-Time Stereo for Weakly-Textured Scenes," *Proc. of MBVA British Machine Vision Conf.*, pp. 72.1–72.10, 2008.

[7] K. Yoon and I. Kweon, "Adaptive Support-Weight Approach for Correspondence Search," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 28, no. 4, pp. 650–656, 2006.

[8] K. Zhang et al., "Real-Time Accurate Stereo with Bitwise Fast Voting on CUDA," *Proc. of IEEE Int. Conf. on Computer Vision Workshops*, pp. 794–800, 2009.

[9] J. Kowalczuk, E. T. Psota, and L. C. Perez, "Real-Time Stereo Matching on CUDA Using an Iterative Refinement Method for Adaptive Support-Weight Correspondences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 1, pp. 94–104, 2013.

[10] X. Sun, X. Mei, S. Jiao, M. Zhou, Z. Liu, and H. Wang, "Real-Time Local Stereo via Edge-Aware Disparity Propagation" *Pattern Recognition Letters*, vol. 49, pp. 201–206, November 2014.

Biographies

Daeyoon Jeon received his B.S. degree in Software Engineering from Hansei University, Gyeonggi, Korea in 2015. He is currently pursuing an M.A. degree in the Department of Digital Imaging, Graduate School of Integrative Technology at Chung-Ang University, Seoul, Korea. His research interests include computer vision, stereo vision, and parallel programming.

Sangyoon Lee received his B.S. degree in Electronic Engineering from Hoseo University, Asan City, Korea, in 2014. He received his M.S. degree from the Department of Imaging Science and Arts, Graduate School of Advanced Imaging Science, Multimedia and Film (GSAIM) at Chung-Ang University, Seoul, Korea, in 2016. His research interests include stereo vision, computer vision, and augmented reality.

Hyunki Hong received his B.S., M.S., and Ph.D. degrees in Electronic Engineering from Chung-Ang University, Korea, in 1993, 1995, and 1998, respectively. From 1998 to 1999, he worked as a researcher at the Automatic Control Research Center, Seoul National University, Korea. From 2000 to 2014, he was a professor in GSAIM at Chung-Ang University. Since 2014, he has been a professor at the School of Integrative Engineering, Chung-Ang University. His research interests include stereo vision, computer vision, and augmented reality.