COST-VOLUME FILTERING-BASED STEREO MATCHING WITH IMPROVED MATCHING COST AND SECONDARY REFINEMENT

Jianbo Jiao[*], *Ronggang Wang*[*], *Wenmin Wang*[*], *Shengfu Dong*[*], *Zhenyu Wang*[*], *Wen Gao*[*,†]

[*]Digital Media R&D Center, Peking University Shenzhen Graduate School
[†]National Engineering Laboratory for Video Technology, Peking University
jianbojiao@sz.pku.edu.cn, {rgwang, wangwenmin, dongsf, wangzhenyu}@pkusz.edu.cn, wgao@pku.edu.cn

ABSTRACT

Recent cost-volume filtering-based local stereo methods have achieved comparable accuracy with global methods. However, there are still some significant outliers existing in the final disparity map. In this paper, we propose a cost-volume filtering-based local stereo matching method that employs a new combined cost and a novel secondary disparity refinement mechanism. The combined cost is formulated by a modified color census transform, truncated absolute differences of color and gradients. Symmetric guided filter is used for the cost aggregation. Different from traditional stereo matching, a novel secondary disparity refinement is proposed to further remove remaining outliers. Experimental results on Middlebury benchmark show that our method ranks the 5^{th} out of the 144 submitted methods, and is the best cost-volume filtering-based local method. Furthermore, experiments on real world sequences also validate the effectiveness of our proposed method.

Index Terms— Local stereo, cost-volume filtering, matching cost, disparity refinement

1. INTRODUCTION

Stereo matching is one of the most active research areas in computer vision. It is the process of computing a disparity map given a pair of stereo images. As mentioned in [1], a large number of methods for stereo correspondence have been proposed. These approaches can be classified into global and local methods. Global methods usually achieve more accurate disparity map with higher computational complexity, while local methods are more efficient.

In recent years, local methods based on adaptive support-weight [2] have achieved results comparable to that of global methods using graph cuts [3] or belief propagation [4]. The main idea of these local methods is to measure the likelihood between center pixel and its neighbor pixels by means of adaptive support weights. A high weight indicates they are likely to be on the same object thus with similar disparities. However, this type of methods involves high computational complexity, and the complexity is related to the window size used for aggregation. Later, Rhemann et al. proposed a method [5] using guided filter [6] as the aggregation strategy. And the complexity is independent of the matching window size. Rhemann et al. provide a new approach for cost aggregation, which is aggregating by smoothing the cost volume. Besides, several other cost-volume filtering-based methods have been developed recently, and achieved good performance. In [7], a hardware-efficient bilateral filter is proposed for fast aggregation. In [8], domain transform is imported so that the cost aggregation can be performed by using 1-D filters. A recursive bilateral filter [9] is introduced by Yang for aggregation. De-Maeztu et al. give an O(1) method [10] based on a symmetric filter. Although the performance of stereo matching has been improved, there are still some obvious artifacts in the final results of these filtering-based methods. Much effort is taken on cost aggregation improvement, but far less attention is paid to disparity refinement, and the cost measurement.

In this paper, we propose two strategies to further improve the performance of cost-volume filtering-based local methods. Firstly, a new combined cost measure by merging truncated absolute difference of color, gradients and a modified color census transform is proposed to improve the initial matching performance. Secondly, after the traditional disparity refinement step, we propose a secondary refinement approach, which is called “Remaining Artifacts Detection and Refinement” (RADAR) in this paper. By means of RADAR, most of the remaining outliers after traditional post-processing can be further corrected, and a remarkable improvement is achieved. For cost aggregation, we employ the symmetric guided filter proposed in [5] and [10]. Experimental results on Middlebury benchmark [11] demonstrate the effectiveness of our method, which is one of the best local stereo matching methods. The performance of our method is the best among cost-volume filtering-based methods on Middlebury dataset. What’s more, our method works well in real world sequences.

The rest of this paper is organized as follows: in section

This work was partly supported by the grant of National Science Foundation of China 61370115, and Shenzhen Peacock Plan.

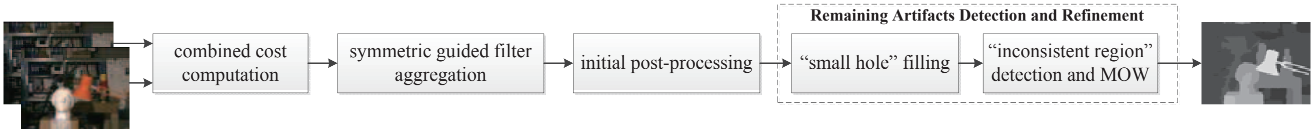
combined cost computation
symmetric guided filter aggregation
initial post-processing
Remaining Artifacts Detection and Refinement
"small hole" filling
"inconsistent region" detection and MOW

Fig. 1: Overview of the proposed method.

2, the proposed local stereo matching method with combined cost function and RADAR scheme are demonstrated in details; section 3 gives the experimental results on both Middlebury dataset and real world sequences; and finally, conclusion is presented in section 4.

2. PROPOSED METHOD

This section demonstrates our proposed cost-volume filtering-based method. First, a cost volume is formulated by our proposed combined cost. Then, a symmetric guided filter is employed for cost-volume filtering. Finally, we propose a RADAR-aided refinement scheme to further improve the accuracy of disparity map. An overview of the proposed method is shown in Fig. 1.

2.1. Combined cost computation and aggregation

Modified color census transform: Motivated by color census transform [12], we propose a modified color census transform (MCCT), which takes full advantage of the color components. As RGB color-space is sensitive to radiometric changes, the image is firstly converted to Gaussian color model [13] space. Then the difference between two pixels p and q is measured by the Euclidean distance $D_G(p,q)$, and the mean value of all these distances in the window centered at p is denoted by $D_m(p)$. The MCCT is formulated as follows:

$$MCCT(p) = \bigotimes_{q \in N(p)} \xi(D_m(p), D_G(p,q)) \tag{1}$$

$$\xi(a,b) = \begin{cases} 1, & b < a \\ 0, & otherwise \end{cases} \tag{2}$$

where operator $\otimes$ denotes a bit-wise catenation, and $N(p)$ represents the neighbor pixel set of p. Hamming distance is used to calculate the difference between the two bit strings generated by MCCT:

$$h(p,d) = Hamming(MCCT_L(p), MCCT_R(p-d)) \tag{3}$$

where d denotes the disparity of two corresponding pixels in left and right iamges. At last, a robust exponential function is used to normalize the cost:

$$C_{MCCT}(p,d) = 1 - \exp(-\frac{h(p,d)}{\lambda_{MCCT}}) \tag{4}$$

A comparison of MCCT and traditional color census [12] is shown in Fig. 2. The window in the last two columns is the census mask (bit string), and white region in the window represents "1". As shown, the proposed MCCT has a better representation of the image structures.

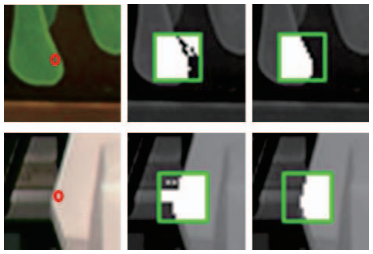

Fig. 2: Comparison of color census and MCCT. Left-to-right: image patches; color census; MCCT.

Combined matching cost: In addition to MCCT, we add two other cost components of truncated absolute difference of color and gradients, which are calculated respectively as follows:

$$\begin{aligned} C_{ADc}(p,d) &= \min(\frac{1}{3}\sum_{i=R,G,B}\|I_i^L(p) - I_i^R(p-d)\|, \lambda_{ADc}), \\ C_{GDx}(p,d) &= \min(\|\nabla_x I_L(p) - \nabla_x I_R(p-d)\|, \lambda_{GD}) \end{aligned} \tag{5}$$

where λ_{ADc} and λ_{GD} are the truncated values, and ∇_x is the derivative in x direction. The gradient in y direction is also employed, denoted as C_{GDy}. The final combined matching cost is formulated by merging the above mentioned four cost components:

$$\begin{aligned} C(x,y,d) = \alpha \cdot C_{MCCT} + \beta \cdot C_{ADc} + \gamma \cdot C_{GDy} \\ + (1 - \alpha - \beta - \gamma) \cdot C_{GDx} \end{aligned} \tag{6}$$

where α, β, γ are the weights for different cost components, adjusting the four components' contribution to the total cost.

Guided filter based aggregation: The combined cost for each pixel at each disparity level is stored in a cost volume. And the cost aggregation is implemented with guided filter [6].In order to preserve both edges in left and right images, we employ the symmetric guided filter proposed in [5] and [10].

After the cost volume is aggregated by symmetric guided filter, the "winner-takes-all" strategy is used for disparity selection, i.e., selecting the disparity label with the lowest cost. Then the initial disparity map is generated.

2.2. Disparity refinement with "RADAR"

There are still many outliers in the initial disparity map, thus a novel RADAR-aided disparity refinement pipeline is pro-

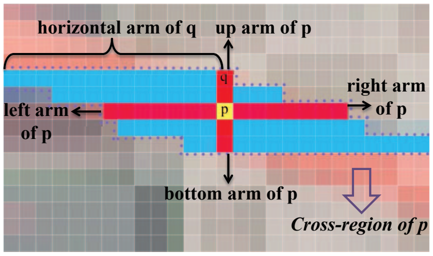

Fig. 3: Cross-region of pixel p. When pixel q belongs to the up arm of p, the horizontal arm of q is set to the support region, and all of these arms compose the cross-region.

posed to improve the accuracy. The pipeline consists of an initial post-processing and the secondary RADAR scheme.

Initial post-processing: In order to find the inconsistent pixels in left and right images, the left-right consistency check (LRC) is employed. A pixel p is labeled as outlier if it violates the following constraint:

$$|d_L(p) - d_R(p - d_L(p))| < 1 \tag{7}$$

where d_L, d_R are the disparities of pixels in the left and right images respectively.

Once the outliers are detected, we use a cross-region based voting technique [14] to correct them. The voting operation is done iteratively to be more robust. The cross region of a pixel p is shown in Fig. 3. More details about the voting method can be found in [14].

After the cross-region voting, we use the nearest reliable pixel in the scan-line to update the remaining outliers labeled by LRC. A weighted median filter with bilateral filter weights [5] is employed to remove the streak-like artifacts.

Remaining Artifacts Detection and Refinement: Some error regions still exist after the initial post-processing. These regions are mainly caused by the drawback of LRC. If these artifacts exist in both left and right images, they can't be detected by just employ the LRC. Thus a secondary refinement scheme named "Remaining Artifacts Detection and Refinement" (RADAR) is proposed in this paper.

According to our observation, remaining artifacts are mainly composed of some "small holes" and outliers around object boundary. "Small holes" are dark regions with disparities much smaller than their neighbors. They can be detected by comparing disparities with their neighbors. After detecting hole-pixel, we use the most appropriate disparity in its neighborhood (both horizontal and vertical) to update it. Commonly, the hole-pixel p is updated by the pixel with smaller disparity (background pixel), but if the pixel on the smaller disparity side of p is also invalid (hole-pixel), it should be updated by the pixel on the other side, as shown in follows:

$$d_p^* = \begin{cases} \max\{d_p^l, d_p^r\}, & d_p^l \cdot d_p^r \leq d_{thres}^2 \\ \min\{d_p^l, d_p^r\}, & d_p^l \cdot d_p^r > d_{thres}^2 \end{cases}, \tag{8}$$
$$d_{thres} = \rho \cdot d_{\max}$$

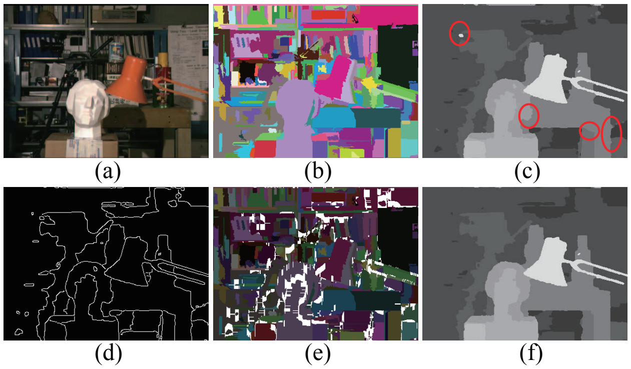

Fig. 4: Inconsistent region detection and MOW. (a) left color image (*Tsukuba*); (b) segmentation after contrast enhancement; (c) input disparity map; (d) edges of (c); (e) detected inconsistent regions (marked as white); (f) the disparity result by MOW.

where d_p^l and d_p^r are the nearest (taking horizontal as an example) pixels' disparities that larger than d_{thres} , and d_{max} is the maximum disparity, while ρ is an empirical penalty of 1/7. The updated disparity is denoted as d_p^*.

As shown in Fig. 4 (c), the other type of artifacts is composed of outliers around object boundary. We name these artifacts as "inconsistent regions", which consist of convex regions (same as "fattening" region [1]) and concave regions. Fig. 4 gives a whole pipeline of the inconsistent region refinement. The inconsistent region is detected by checking whether the edges of disparity map coincide with the boundaries of objects in the scene. Canny edge detector [15] is used to extract the disparity edges. A mean-shift [16] based color segmentation approach is utilized to detect the objects' boundaries. Beforehand, a contrast enhancement operation (histogram equalization on the luminance part of the color image) is performed, and then the image is converted to CIELab space. With this kind of preprocessing, segmentation accuracy is improved, especially on dark regions. If an edge in disparity map does not exactly coincide with the object boundary in the scene, it is labeled as a mismatch edge. Convex regions can be detected by searching the foreground side (larger disparity side) of the mismatch edge, while concave regions the background side (smaller disparity side). Recently, a similar method focusing on fattening region management achieved a remarkable result [17]. Whereas, as can be seen in Fig. 5, at the edge of disparity map, some regions being mislabeled by the method [17]. The circle marked region is segmented into a whole area, ignoring the boundary there.

In order to correct the outliers detected above, we propose a modified OccWeight (MOW) based on the OccWeight presented in [18]. The OccWeight method replaces a pixel's disparity by choosing the most likely one in a fixed window around it, and the likelihood is expressed by an occlusion-aided weight. However, a fixed window is not robust. Hence we use the adaptive window as shown in Fig. 3 to improve the accuracy. In addition, the disparity inheritance [18] is also

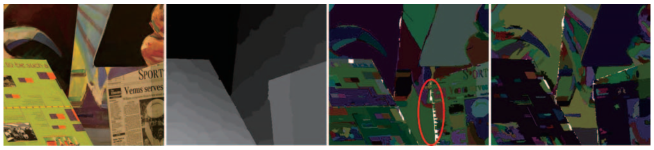

Fig. 5: Comparison of inconsistent region detection. Left-to-right: left color image (*Venus*); input disparity map; segmentation result and detected fattening regions (marked as white) by method in [17]; the approach proposed in this paper.

adopted. In a cross-region (adaptive) window of pixel p, the weight of its neighboring pixel (q) is calculated as follows:

$$sw(p,q) = \begin{cases} \exp(-\frac{\Delta c_{pq}}{\phi_c} - \frac{\Delta s_{pq}}{\phi_s}), & if\ q \notin R_f \\ 0 & , \ otherwise \end{cases} \tag{9}$$

where Δc_{pq} and Δs_{pq} are the color distance and spacial distance between p and q, and ϕ_c, ϕ_s are parameters used to normalize the color and spacial distances, respectively. And R_f is the set of inconsistent regions. Then the updated disparity is calculated as :

$$\begin{aligned} &d^*(p) = \arg\max_{d \in D}(\sum_{q \in AW_p} sw(p,q) \times m(q,d)), \\ &m(q,d) = \begin{cases} 1, & if\ d(q) = d \\ 0, & otherwise \end{cases} \end{aligned} \tag{10}$$

where D is the set of disparity candidates, and AW_p is the set of pixels in the adaptive window of p. By employing the MOW, outliers at inconsistent regions are corrected (Fig. 4(f)).

Finally, we smooth the disparity map with a median filter to remove remaining noises. We compare the RADAR-aided disparity refinement with the refinement (denoted as MDC) proposed in [17] and the referenced method OccWeight [18]. For MDC, we use the fattening detection method in [17], while for OccWeight, the region-detection method in this paper is employed. In addition, the RADAR-only (RADAR-o) item is also evaluated, which is without the initial post-processing. All of these methods are based on the same initial disparity map, i.e., based on our proposed combined cost and aggregation. For the evaluation, we choose the Middlebury dataset (*Tsukuba*, *Venus*, *Teddy*, and *Cones*)[11], and the evaluation measures "Nonocc", "All", "Disc" are used here, representing the non-occluded regions, all regions, and regions near discontinuities, respectively. For each index, average values of the four images' are calculated. The results are shown in Fig. 6. As can be seen, the refinement pipeline proposed in this paper performs best among all these methods.

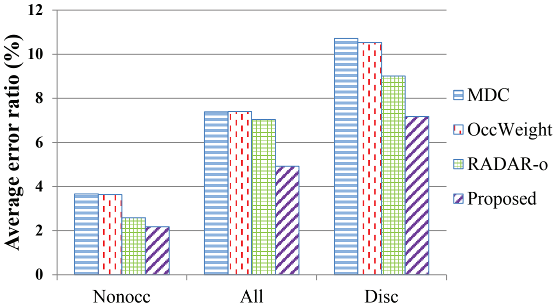

Fig. 6: Comparison of different refinement approaches.

3. EXPERIMENTAL RESULTS

This section shows an experimental evaluation on the proposed method. Here, the evaluation is performed over two test sets. One is the Middlebury dataset [11], while the other is some real world sequences. The parameters used in the experiment are chosen empirically and kept constant, which are shown as follows:

$$\{\lambda_{MCCT}, \lambda_{ADc}, \lambda_{GD}, \alpha, \beta, \gamma, \phi_c, \phi_s\} = \\ \{55, \tfrac{7}{255}, \tfrac{2}{255}, 0.011, 0.15, 0.1, 15, 10.5\}$$

in which α, β, γ are obtained based on 35 images with ground truth on Middlebury datasets according to an optimization process aiming at obtaining the minimum average error ratio on the datasets, while the rest parameters are chosen according to [5] and [18].

3.1. Middlebury dataset

The experimental results on four test images (*Tsukuba*, *Venus*, *Teddy*, and *Cones*) from Middlebury online benchmark [11] of our proposed method are shown in Fig. 7. Our proposed method obtains competitive performance with the state-of-the-art methods, and ranks the 5^{th} out of 144 methods by the time we submit. To the best of our knowledge, our method is the top performer of cost-volume filtering-based local methods. Besides, the proposed method greatly outperforms the original GuidedFilter method [5], which ranks the 34^{th} on the benchmark.

Our method is also compared with some other filtering-based local methods and "ADCensus" [14] (the top performed local method) on Middlebury. In addition, result for CostFilter [5] using our proposed combined cost (CostFilterwCC) is also listed. The comparison results are listed in Table 1, and error percentages in different regions for the four images are presented. Error threshold is set to the default 1.0. Meanwhile, subpixel threshold 0.75 is also chosen, and the rank on it can be seen in the last column (Rank*) of Table 1.

From Table 1, when error threshold is 1.0, the method proposed in this paper is the best cost-volume filtering-based method, and the second best local method, being close behind "ADCensus" [14]. However, when errors being evaluated at subpixel level (0.75), our method performs best in the selected methods. Subpixel evaluation means the disparity can be a floating number, instead of being limited to integer, and it is useful in practical applications. But it's worth noting that our method has no regard of subpixel, which means all of the disparities are estimated at integer level. In subpixel

Table 1: Quantitative evaluation of the proposed method compared with other local methods on Middlebury.

Algorithm	Rank	*Tsukuba*			*Venus*			*Teddy*			*Cones*			Rank*
		nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	
ADCensus [14]	2	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.9	2.42	7.25	6.95	27
Proposed	5	1.15	1.42	6.23	0.15	0.27	1.89	5.39	10.6	14.7	2.01	7.37	5.88	10
HEBF [7]	27	1.10	1.38	5.74	0.22	0.33	2.41	6.54	11.8	15.2	2.78	9.28	8.10	20
CostFilterwCC	30	1.38	1.74	7.38	0.15	0.42	2.12	6.28	11.6	16.6	2.54	7.96	7.46	15
DTAggr-P [8]	32	1.75	2.10	7.09	0.24	0.45	2.59	5.70	11.5	13.9	2.49	7.82	7.30	29
CostFilter [5]	34	1.51	1.85	7.61	0.20	0.39	2.42	6.16	11.8	16.0	2.71	8.24	7.66	16
P-LinearS [10]	45	1.10	1.67	5.92	0.53	0.89	5.71	6.69	12.0	15.9	2.60	8.44	6.71	44
RecursiveBF [9]	58	1.85	2.51	7.45	0.35	0.88	3.01	6.28	12.1	14.3	2.80	8.91	7.79	30

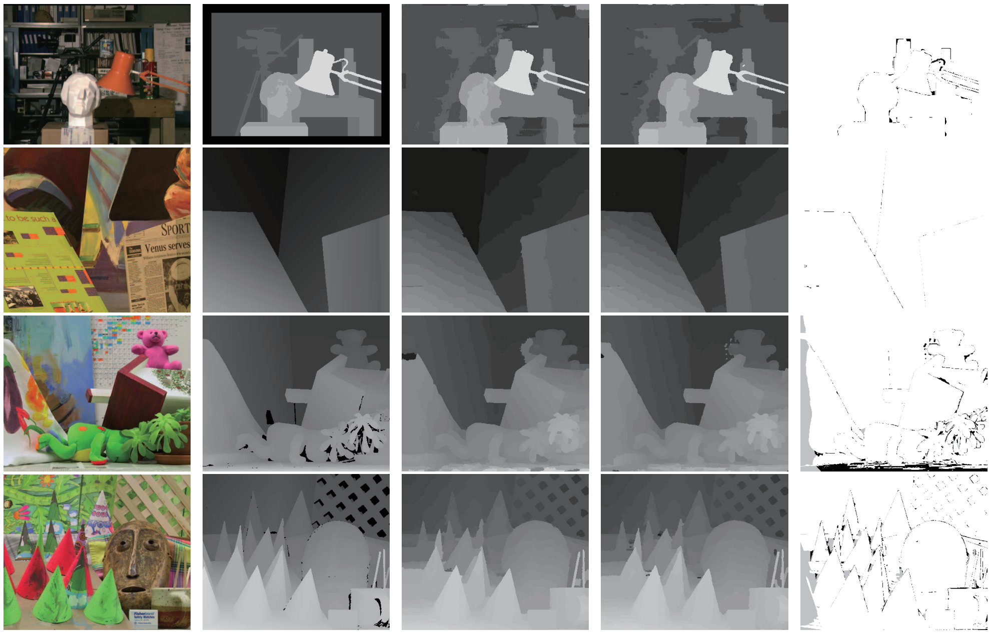

Fig. 7: Experimental results on Middlebury dataset. Top-to-bottom: *Tsukuba*, *Venus*, *Teddy*, *Cones*, respectively. Left-to-right: color images; ground truth; results of CostFilter [5]; results of our method; error maps of our results with error threshold equals 1.0.

level, the performance of our method only has a slight decline (rank from 5 to 10), which proves the robustness of our proposed method. Besides, the result of CostFilterwCC shows the effectiveness of our proposed combined cost.

3.2. Real world sequences

In this experiment, we choose four real world sequences as the test set: the *BookArrival* sequence from HHI 3D video database [19], the *Balloons* sequence from FTV [20], and *Cafe* and *Newspaper* sequences obtained from GIST [21]. For each test sequence, we randomly extract a frame with its corresponding view as the test image pair. Besides, some competitive filtering-based methods discussed previously are selected for comparison, which are RecursiveBF [9], CostFilter [5], DTAggr-P [8], and HEBF [7]. The parameter settings of these methods are the same as recommended in these papers. Experimental results are shown in Fig. 8.

From the visual results we can see that, being compared with other methods, our method has better edge-preserved performance, such as the lion in *BookArrival* sequence and objects in *Balloons* sequence. What's more, our disparity results are well aligned to image borders, e.g., the coat on the left border of *BookArrival* and *Newspaper* sequences, which is an important feature in practical applications, such as virtual view synthesis and 3D reconstruction. The experimental results on real word sequences again prove the effectiveness of our proposed method.

In our proposed method, the mainly time-consuming parts are the aggregation step and the refinement step, accounting for 60.61% and 24.49% of the total time. However, both of them as well as the cost computation can be parallel on the GPU for a large extent of acceleration.

4. CONCLUSION

In this paper, we propose a secondary refinement scheme and a combined cost to improve the performance of local filtering-based stereo matching. The secondary refinement scheme, namely RADAR, mainly focuses on handling remaining artifacts after traditional disparity refinement. In the combined cost, a modified color census transform (MCCT) is proposed combined with truncated AD and gradients. The experimental results show that our proposed method achieves the state-of-the-art performance and is the best cost-volume

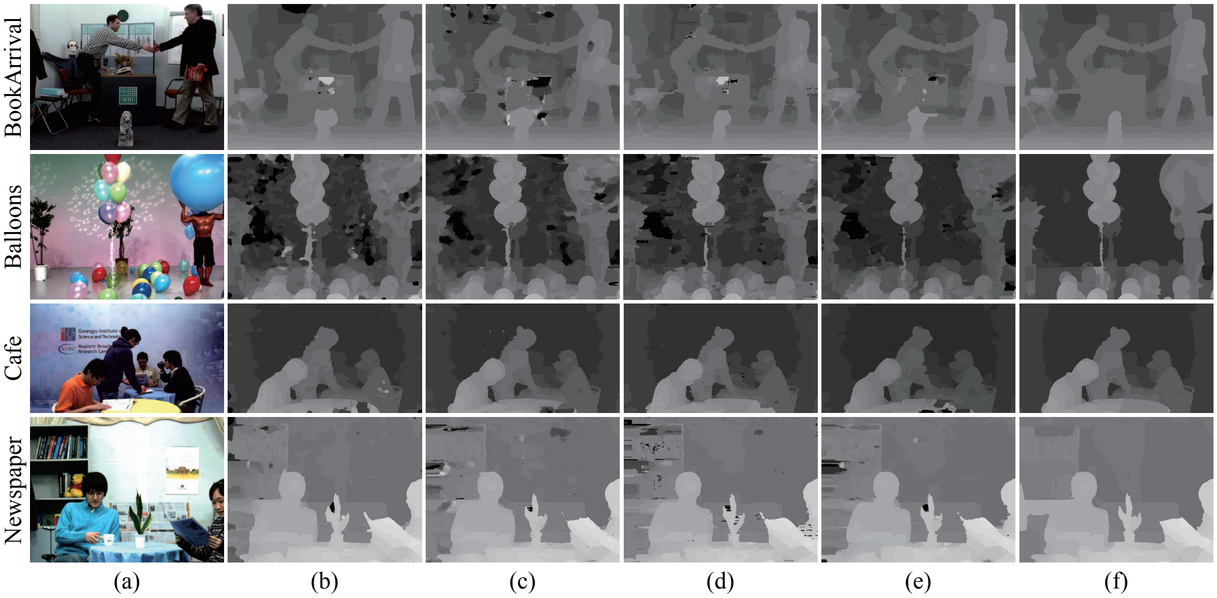

Fig. 8: Experimental results of our method compared with other competitive methods on real world sequences. (a) left frames; (b) results of RecursiveBF; (c) results of CostFilter; (d) results of DTAggr-P; (e) results of HEBF; (f) results of proposed method.

filtering-based local method according to Middlebury benchmark. In addition, experimental results on four representative real world sequences show the effectiveness of our method as well.

5. REFERENCES

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

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

[3] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *ICCV*, 2001, pp. 508–515.

[4] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE PAMI*, vol. 25, no. 7, pp. 787–800, 2003.

[5] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *CVPR*, 2011, pp. 3017–3024.

[6] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE PAMI*, vol. 35, no. 6, pp. 1397–1409, 2013.

[7] Q. Yang, "Hardware-efficient bilateral filtering for stereo matching," *IEEE PAMI*, vol. PP, no. 99, pp. 1–8, 2013.

[8] C. Pham and J.W. Jeon, "Domain transformation-based efficient cost aggregation for local stereo matching," *IEEE CSVT*, vol. 23, no. 7, pp. 1119–1130, 2013.

[9] Q. Yang, "Recursive bilateral filtering," in *ECCV*, 2012, pp. 399–413.

[10] L. De-Maeztu, S. Mattoccia, A. Villanueva, and R. Cabeza, "Linear stereo matching," in *ICCV*, 2011, pp. 1708–1715.

[11] D. Scharstein and R. Szeliski, "Middlebury Stereo Website [Online]," Available: `http://vision.middlebury.edu/stereo/`.

[12] S.C. Pei and Y.Y. Wang, "Color invariant census transform for stereo matching algorithm," in *ISCE*, 2013, pp. 209–210.

[13] J.M. Geusebroek, R. Van den Boomgaard, A.W.M. Smeulders, and H. Geerts, "Color invariance," *IEEE PAMI*, vol. 23, no. 12, pp. 1338–1350, 2001.

[14] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On building an accurate stereo matching system on graphics hardware," in *ICCV Workshops*, 2011, pp. 467–474.

[15] J. Canny, "A computational approach to edge detection," *IEEE PAMI*, vol. 8, no. 6, pp. 679–698, 1986.

[16] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE PAMI*, vol. 24, no. 5, pp. 603–619, 2002.

[17] Y.-C. Wang, C.-P. Tung, and P.-C. Chung, "Efficient disparity estimation using hierarchical bilateral disparity structure based graph cut algorithm with foreground boundary refinement mechanism," *IEEE CSVT*, vol. 23, no. 5, pp. 784–801, 2013.

[18] W. Wang and C. Zhang, "Local disparity refinement with disparity inheritance," in *SOPO*, 2012, pp. 1–4.

[19] I. Feldmann, M. Mller, F. Zilly, R. Tanger, K. Mller, A. Smolic, P. Kauff, and T. Wiegand, "HHI test material for 3d video," *ISO/IEC JTC1/SC29/WG11 MPEG08/M15413*, 2008.

[20] "MPEG-FTV project, Tanimoto Lab at Nagoya University [Online]," Available: `http://www.tanimoto.nuee.nagoya-u.ac.jp/`.

[21] Y.-S. Ho, E.-K. Lee, and C. Lee, "Multiview video test sequence and camera parameters," *ISO/IEC JTC1/SC29/WG11 MPEG2008/M15419*, 2008.