An Improved Stereo Matching Algorithm Based on Guided Image Filter

Ruidong Gao, Yun Chen, Lina Yan
School of instrumentation Science and Opto-electronics Engineering, Beihang University, Beijing 100191, China
rxdj1990@163.com

***Abstract*—Stereo matching is a challenging issue in computer vision field. To address the poor accuracy behavior of local algorithms, we propose an improved stereo matching algorithm based on guided image filter. Firstly, we put forward a combined matching cost by incorporating the absolute difference and improved color census transform (ICCT). Secondly, we use the guided image filter to filter the cost volume, which can aggregate the costs fast and efficiently. Then, in the disparity computing step, we design a modified dynamic programming algorithm, which can weaken the scanning line effect. At last, the final disparity maps are gained after post-processing. The experimental results are evaluated on the Middlebury stereo dataset, showing that our approach can achieve good results both in low texture and depth discontinuity areas with an average error rate of 5.14%.**

Keywords-stereo matching; census transform; guided image filter; dynamic programming

I. INTRODUCTION

Stereo matching is one of the most active research areas in computer vision. It refers to the process of estimating the scene depth by finding the corresponding points in the binocular or multi-ocular images [1]. It is widely used for visual reality, object recognition, and depth-image based rendering [2]-[5]. Stereo matching algorithms can be classified into local and global algorithms. Local algorithms compute each pixel's disparity value in the light of the intensity values within a window of finite size [6], [7]. However, global algorithms regard stereo matching as an energy minimization problem and obtain global disparity allocation via optimization methods such as dynamic programming (DP) [8], graph cuts [9], and belief propagation [10].

The traditional matching cost computation methods encompass the pixel-based cost, region-based cost, filter-based cost, and the non-parameter transformation-based cost. [11] gave a new idea of combining AD and census transform, achieving good results by exploiting the advantages of different matching costs computation methods. [12] pioneered the use of adaptive weight in the cost aggregation process, greatly improving the accuracy of the local stereo matching algorithm. This method is actually equivalent to the computation of the weight using the bilateral filter, but its computational complexity is high. On this basis, many new self-adaptive weighting methods are proposed. [13] determined the weights by computing the geodesic distance between the window pixel and the central pixel, increasing the matching accuracy through imposing the connectivity constraint. But the computational complexity was not reduced. Guided filter has good edge-preserving smoothing properties like the bilateral filter, and it does not suffer from the gradient reversal artifacts [14]-[16]. To the best of our knowledge, it is one of the best edge-preserving filtering.

To address these problems mentioned above, we combine AD and improved color census transform (ICCT) to compute matching cost. Then, the guided image filter is used to generate the adaptive weight in the process of cost aggregation. Moreover, the disparity value is selected using the modified dynamic programming algorithm to obtain the high-accuracy disparity map quickly and effectively.

II. ALGORITHMS

The proposed method is an efficient correlation and filter-based local method. An overview of the block diagram of the approach is provided in Fig. 1. Firstly, the algorithm transforms the images before calculating the matching costs under each disparity level. Then we use AD + ICCT to measure the similarity between two points and adopt guided filter to compute support weights. Finally, in the post-processing process, we employ three steps including region voting, interpolation and weighted median filter for disparity refinement. The main components will be discussed in details.

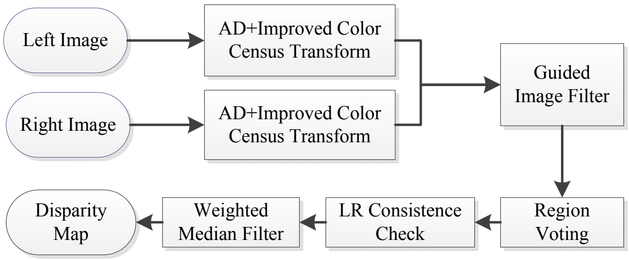

Figure 1. The block diagram of the proposed approach.

A. *Matching cost computation*

Motived by color census transform, we propose an improved color census transform. First, we convert the image from RGB color space to Gaussian color model (GCM) space, because RGB color space is sensitive to radiometric. Then the Euclidean distance is used to measure the difference between two pixels p and q. We define $D_m(p)$ as the mean value of all these distance in the window centered at p. $D_G(q)$ indicates the distance between p and other pixels in the window centered at p. The relation between $D_m(p)$ and $D_G(q)$ is defined as:

$$\xi\left(D_m(p), D_G(q)\right) = \begin{cases} 1, & if\ D_G(q) < D_m(p) \\ 0, & else \end{cases} \qquad (1)$$

For each pixel p in the image I, the census transform encodes the local neighborhood around p to a bit string, representing the set of surrounding pixels with lower distance value than $D_m(p)$. The transform results can be indicated as:

$$T(p) = \underset{q \in N_p}{\otimes} \xi\left(D_m(p), D_G(q)\right) \qquad (2)$$

Where $\otimes$ denotes the act of concatenation and N_p is the neighborhood area of p. For a particular disparity d, the matching cost between a pixel p in the reference image and its corresponding pixel pd in the matching image, is calculated through the Hamming distance. The Hamming distance represents the number of unequal elements in the two bit streams:

$$C_{census}(p,d) = Hamming\left(T_p, T_{pd}\right) = T_p \oplus T_{pd} \qquad (3)$$

For image regions with similar local structures, census transform could bring in matching ambiguities. Thus, we combine the absolute difference in color and the census transform to form a more accurate metric. Equation (4) is used to calculate the AD value.

$$C_{AD}(p,d) = \frac{1}{3}\sum_{i \in R,G,B}\left|I^i_L(p) - I^i_R(p,d)\right| \qquad (4)$$

Where $I^i_L(p)$ denote the intensity values of pixels in the reference image and $I^i_R(p,d)$ represent the intensity values of pixels in the target image. They are both calculated in three color channels.

The final pixel-wise matching cost is defined as (5):

$$C(p,d) = 1 - \exp\left(-\frac{C_{AD}(p,d)}{\lambda_{AD}}\right) + 1 - \exp\left(-\frac{C_{Census}(p,d)}{\lambda_{Census}}\right) \qquad (5)$$

Where λ_{AD} and λ_{Census} are normalizing parameters.

B. *Guided filter based on adaptive weights*

According to the above definition, we can compute every cost for assigning disparity d at pixel $p = (x, y)$. Then we use a three dimension array C, which is usually called the cost volume or disparity space image (DSI), to store all these costs. Each cost can be indexed by $C(p,d)$. As pixel-to-pixel comparison is too sensitive to noise, it is common to aggregate the matching costs in a support region usually defined by a window. To improve the accuracy of matching results, the method of adaptive weights was developed and has been widely used for its great effectiveness. In [12], the aggregation step was formulated as:

$$\overset{\square}{C}(p,d) = \frac{\sum_{q \in N_p, \bar{q}_d \in N_{\bar{p}d}} \omega(p,q)\,\omega(\bar{p}_d, \bar{q}_d)\,C(q,d)}{\sum_{q \in N_p, \bar{q}_d \in N_{\bar{p}d}} \omega(p,q)\,\omega(\bar{p}_d, \bar{q}_d)} \qquad (6)$$

Where N_p and N_{p_d} are the set of pixels in the support window of corresponding pixels p and p_d respectively. $\omega(p,q)$ and $\omega(\bar{p}_d, \bar{q}_d)$ are the designated weights in the reference and target windows. The denominator is used to normalize the weights. As combining the support weights in both windows only helps to improve correspondence search slightly, we can omit the term of $\omega(\bar{p}_d, \bar{q}_d)$. Thus the following equation is obtained.

$$\overset{\square}{C}(p,d) = \sum_{q \in N_p} W_{p,q} C(q,d) \qquad (7)$$

In this way, cost aggregation is used to compute the weighted average in a support window for every pixel. It can be implemented by smoothing the cost volume with a filter. So the weights are defined by the filter kernels. In fact, the method in [12] is equivalent to the bilateral filtering. Here we use the newly developed guided filter to compute weights. Guided filter gives a weight between two pixels p and q, according to their statistical analogy of the reference intensities based on the averages and the variances of several squared windows. For color guidance images, the weight is described as follow:

$$W_{p,q} = \frac{1}{|\omega|^2}\sum_{k \in \omega_p \cap \omega_q}\left(1 + \frac{\left(\mathbf{I}(p) - \boldsymbol{\mu}_k\right)^{\mathrm{T}}\left(\mathbf{I}(q) - \boldsymbol{\mu}_k\right)}{\sigma_k{}^2 + \varepsilon\mathrm{U}}\right) \qquad (8)$$

Where, $\mathbf{I}(p)$, $\mathbf{I}(q)$ and $\boldsymbol{\mu}_k$ are 3×1 vectors as they have R, G, B channels. $\sigma_k{}^2$ and $\mathbf{U}$ are 3×3 matrixes. $\boldsymbol{\mu}_k$ and $\sigma_k{}^2$ are the mean and co-variance matrix of $\mathbf{I}$ in a square window with dimensions $r \times r$ centered at pixel k. The number of pixels in this window is denoted with $|\omega|$. ε is a smoothness parameter and $\mathbf{U}$ is an identity matrix. Guided filter has an edge-preserving property which can be easily understood by investigating the above equation. Parameter ε controls the strength of smoothing and can be tuned experimentally. To show the property of guided filter intuitively, we visualized the filter weights for some image regions in Fig. 2. In order to observe different image areas clearly, we gave the partial enlarged views.

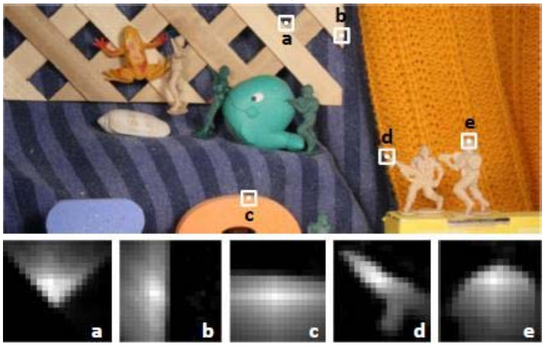

Figure 2. The demonstrations of generating adaptive weight by using guided filter.

The weights are high in regions which are self-similar to the central pixel and low otherwise, agreeing with the local image structure.

In the real implementation, instead of computing $W_{p,q}$ explicitly, the guided filter can be performed in the following equivalent way:

$$\tilde{C}\left(p,d\right)=\bar{\mathbf{a}}_{p}I\left(p\right)+\bar{b}_{p} \tag{9}$$

$$\mathbf{a}_{k}=\left(\sigma_{\mathrm{k}}{}^{2}+\varepsilon\mathrm{U}\right)^{-1}\left(\frac{1}{|\omega|}\sum\nolimits_{i\in\omega_{k}}\mathbf{I}\left(i\right)C\left(i,d\right)-\boldsymbol{\mu}_{k}\overline{C}\left(k,d\right)\right) \tag{10}$$

$$b_{k}=\overline{C}\left(k,d\right)-a_{k}^{\mathrm{T}}\mu_{k} \tag{11}$$

$\mathbf{I}$ is the reference image and acts as the role of guided image. $\bar{\mathbf{a}}_{p}$ and $\bar{b}_{p}$ are the average of $\mathbf{a}_{k}$ and b_{k} in the square window ω_{k} which involves p. $\overline{C}\left(k,d\right)$ is the mean of input cost slice $C\left(\cdot,d\right)$ in ω_{k}.

C. Disparity selection

Dynamic programming (DP) is a technique used to solve a complex problem by breaking it down into several sub-problems. It solves each sub-problem each time, which greatly reduces the computational complexity. DP-based algorithms formulate stereo correspondence as a least-cost path finding problem. Given an image scanning line $S_{y}=\left\{p\left(\cdot,y\right)\right\}$, DP finds an optimal path through a 2D slice $C\left(\cdot;y;\cdot\right)$ of the 3D cost-volume. The optimal path is equivalent to a disparity assignment function $d\left(p\right)$ that minimizes the global energy function:

$$E\left(d\right)=E_{data}\left(d\right)+E_{smooth}\left(d\right) \tag{12}$$

Here, the data term comes directly from the aggregated matching costs, which is defined in (13). The smoothness term is defined in (14):

$$E_{data}\left(d\right)=\sum_{\left(x,y\right)}C\left(x,y,d\right) \tag{13}$$

$$E_{smooth}\left(d\right)=\sum_{\left(x,y\right)}\lambda\left|d\left(x,y\right)-d\left(x-1,y\right)\right| \tag{14}$$

The smoothness term implicitly assumes that the disparity changes smoothly and imposes penalty for abrupt change of disparity. We modified the traditional dynamic programming (DP) algorithm and proposed the WTA guidance-based DP algorithm. The proposed DP algorithm will be described in details below:

Consider the scan line y in the reference image. Each pixel $p=\left(x,y\right)$ is visited from left to right. Its energy corresponding to all disparities d is computed. At the same time, minimum energy and the position of the corresponding disparity are saved. This can be given by:

$$\begin{aligned}M\left(x,y,d\right)=C\left(x,y,d\right)+&\\ \min_{d'\in\left[0,d_{\max}\right]}\left(M\left(x-1,y,d'\right)+\lambda\left|d-d'\right|\right)&\end{aligned} \tag{15}$$

Where d' denotes the disparity of $p'=\left(x-1,y\right)$, which is the neighboring pixel of p. The dynamic programming algorithm mentioned above has a computation load of $O\left(WD^{2}\right)$ for each scan line, where W denotes the image width and D denotes the disparity range. So it is not suited for real-time applications. To reduce the computational load, we consider the disparity smoothness assumption. By assuming that the disparity of p's neighboring pixel is close to that of p and limiting the value of d' within $\left\{d-1,d,d+1\right\}$, we obtain the modified equation as follows:

$$\begin{aligned}M\left(x,y,d\right)=C\left(x,y,d\right)+&\\ \min_{d'\in\left[d-1,d,d=1\right]}\left(M\left(x-1,y,d'\right)+\lambda\left|d-d'\right|\right)&\end{aligned} \tag{16}$$

The computational complexity of the modified algorithm is $O\left(WD\right)$, but the disparity computed by the simplified algorithm changes too slowly in the depth discontinuity areas. This may blur the depth edges and cause the scan line effect. To avoid the over-smoothing phenomenon, the original disparity image from the WTA method is used to guide the dynamic programming process. The core idea is to provide an additional disparity candidate for the dynamic programming process by using WTA method. The original disparity image d_{0} is computed by (17). For the pixel

$p=(x,y)$, let $d_0(x-1,y)$ be the fourth disparity candidate of p' :

$$M(x,y,d)=C(x,y,d)+\min_{d'\in[d-1,d,d+1,d_0(x-1,y)]}\left(M(x-1,y,d')+\lambda|d-d'|\right) \quad (17)$$

Because the original disparity image d_0 has been computed at the beginning, the computational complexity of the modified dynamic programming algorithm (WTA-DP) is not increased, which is still d_0 .

D. Disparity post-processing

There are some mismatches in the original disparity images, so it needs to be post-processed. First, left-right consistency check is adopted to detect occluded points. Let $d_L(p)$ denotes the left disparity image and $d_R(p)$ denotes the right disparity image. If the disparity of the point is not consistent with that of its corresponding point, $d_L(p)\neq d_R(p-d_L(p))$. Then we regard the point p as the occluded point and label its disparity value as zero. Next, we search the scan line, which point p locates on, for the first non-occluded point at its left and right directions, Afterwards the occluded point is filled by selecting the smaller disparity value as the disparity of the occluded point. Finally, the weighted median filter is used to smooth the disparity image and obtain the final disparity image.

III. Results

To validate the effectiveness of the proposed algorithm, C Language was used to implement the proposed algorithm. The experiments were carried out on the standard stereo image pairs from the recognized Middlebury Platform [17] for testing stereo matching algorithms. This website provides four groups of baseline color image pairs: Tsukuba, Venus, Teddy, and Cones. The quantified matching errors are obtained by comparing the experimental results with the ground truth, thus enabling the algorithm's accuracy to be evaluated objectively. Unless specified description, the parameter setting of all experiments is as follows:

$$\{\lambda_{AD},\lambda_C,\lambda_G,\beta,r,\varepsilon\}=\{45,30,15,50,9,0.0001\} \quad (18)$$

Fig. 3 intuitively shows the accuracy of the proposed algorithm. The experimental results of Tsukuba, Venus, Teddy, and Cones are sequentially given in Fig. 3(a)-(d). In this figure, the first column is the original image. The second column displays the related ground truth. The disparity image from the proposed algorithm is given in the third column. The fourth column presents the mismatched pixel image of the proposed algorithm. In the fourth column, the large white areas are the correctly matched points, whereas, the grey and black areas represent the mismatched points in the occluded areas and the non-occluded areas, respectively. It can be seen that the disparity image of the proposed algorithm is smooth overall, and it is capable of achieving desired matching results in the less-textured areas and disparity edges.

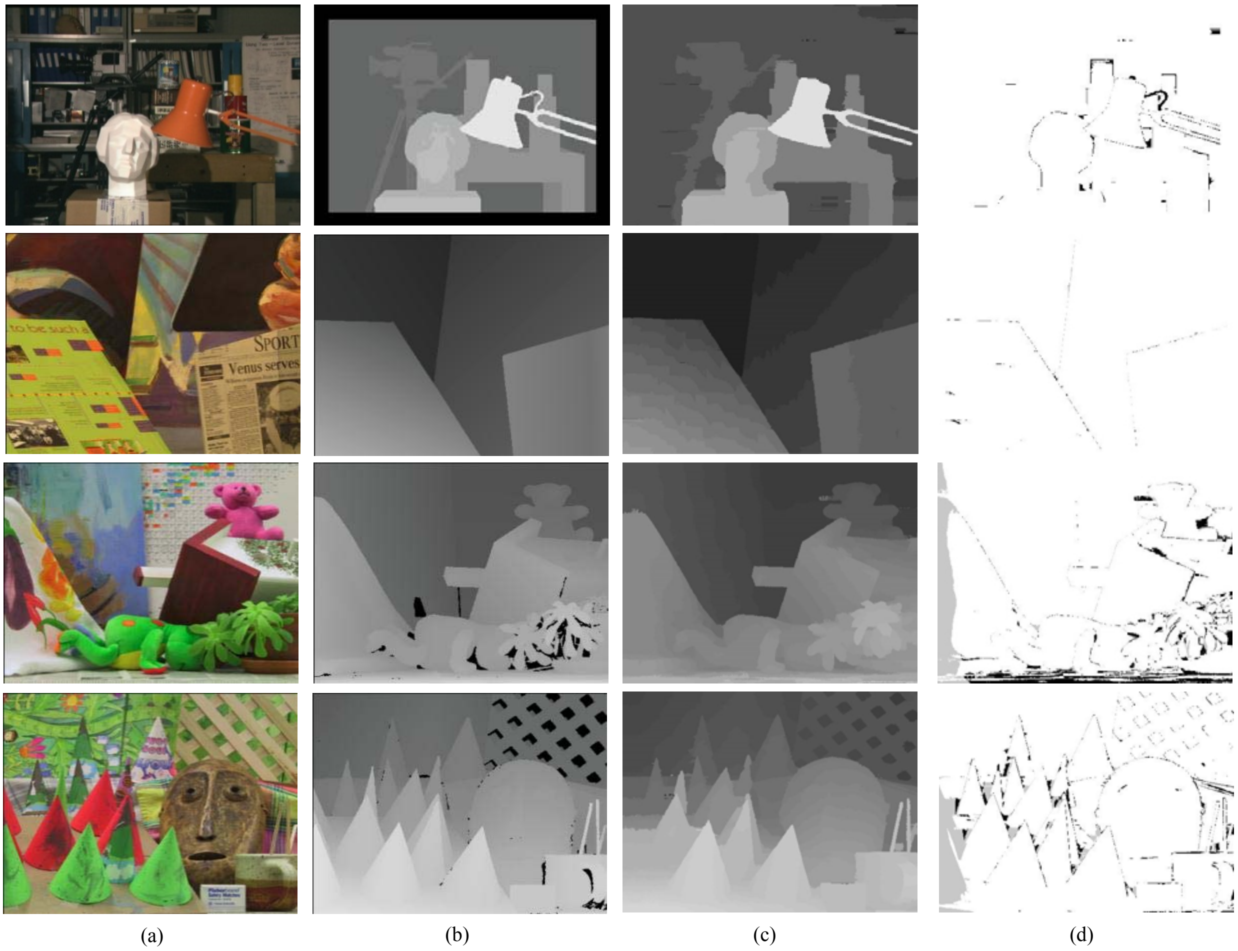

Figure 3. The comparison of the results. (a) The four reference images, from top to bottom, they are Tsukuba, Venus, Teddy, and Cones. (b) Ground Truth maps. (c) Matching results using the proposed method. (d) Error maps using the proposed method.

TABLE I. AVERAGE PERCENTAGE OF BAD PIXELS WITH DISPARITY ERROR THRESHOLD OF 1 USING THE MIDDLEBURY BENCHMARK

Algorithms	Tsukuba			Venus			Teddy			Cones			Average Error
	nooocc	all	disc	nooocc	all	disc	nooocc	all	disc	nooocc	all	disc	
Proposed	**1.41**	**1.74**	**6.86**	**0.17**	**0.33**	**1.87**	**5.62**	**10.7**	**15.1**	**2.64**	**7.86**	**7.37**	**5.14**
DTAggr-P [18]	1.75	2.10	7.09	0.24	0.45	2.59	5.70	11.5	13.9	2.49	7.82	7.30	5.24
HEBF [19]	1.10	1.38	5.74	0.22	0.33	2.41	6.54	11.8	15.2	2.78	9.28	8.10	5.41
GlobalGCP [20]	0.87	2.54	4.69	0.46	0.53	2.22	6.44	11.5	16.2	3.59	9.49	8.9	5.60
ASSW [21]	1.81	2.17	7.58	0.32	0.51	3.73	7.02	12.5	17.4	3.21	8.40	8.99	6.16

The obtained disparity image can be compared with the ground truth to evaluate the accuracy of the algorithm objectively by defining the accuracy as the proportion of the mismatched pixels. And the mismatched pixel is defined as the point whose absolute difference with the ground truth is larger than 1:

$$B = \frac{1}{N}\sum_{(x,y)}\left(\left|d_C\left(x,y\right)-d_T\left(x,y\right)\right| > \delta_d\right) \qquad (19)$$

Where $d_c\left(x,y\right)$ denotes the computed disparity image, $d_T\left(x,y\right)$ represents the true disparity map, N denotes the total number of pixels, and δ_d denotes the specified error threshold. The point whose matching difference with the true disparity map is over 1 pixel is regarded as the mismatched point. Table I provides the percentage data of mismatched pixels for the proposed algorithm when $\delta_d = 1$. From the table, it can be seen that the matching accuracy of the proposed algorithm is higher than the compared stereo matching algorithms.

IV. CONCLUSION

A novel local stereo matching algorithm which can obtain the disparity image accurately is proposed in this paper. A combined matching cost based on AD and improved color census transform is adopted to overcome the disadvantages of the single matching cost. We perform the adaptive cost aggregation by filtering the cost volume using the guided image filter. Moreover, a modified dynamic programming algorithm is used to weaken the scanning line effect from the obtained disparity image. The experimental results on Middlebury platform show that our method performs well among the local stereo matching methods.

ACKNOWLEDGMENT

This research is funded by the National Natural Science Foundation of China (NSFC) under grants No. 61375025.

REFERENCES

[1] D. Scharsterein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," International Journal of Computer Vision, vol. 47, no. 1, pp: 7-42, 2002.

[2] S. P. Zhu, Z. Li and Y. Yu, "Virtual view synthesis using stereo vision based on the sum of absolute difference," Computers and Electrical Engineering, vol. 40, no. 8, pp. 236–246, November 2014.

[3] S. P. Zhu, L. Y. Li and Z. K. Wang, "A novel fractal monocular and stereo video codec with object-based functionality," EURASIP Journal on Advances in Signal Processing, vol. 2012, Article number 227, pp. 1-14, October 2012.

[4] S. P. Zhu and Y. Gao, "Noncontact 3-D coordinates measurement of cross-cutting feature points on the surface of a large-scale workpiece based on the machine vision method," IEEE Transactions on Instrumentation and Measurement, vol. 59, no. 7, pp. 1874-1887, July 2010.

[5] S. P. Zhu, J. C. Fang, R. Zhou, J. H. Zhao and W. B. Yu, "A new noncontact flatness measuring system of large 2-D flat workpiece," IEEE Transactions on Instrumentation and Measurement, vol. 57, no. 12, pp. 2891-2904, December 2008.

[6] S. P. Zhu and Z. Li, "Local stereo matching using combined matching cost and adaptive cost aggregation," KSII Transactions on Internet and Information Systems, vol. 9, no. 1, pp. 224-241, January 2015.

[7] S. P. Zhu, Y. S. Hou, Z. K. Wang and K. Belloulata, "Fractal video sequences coding with region-based functionality," Applied Mathematical Modelling, vol. 36, no. 11, pp. 5633-5641, November 2012.

[8] J. K. Kim, K. M. Lee, and B. T. Choi, "A dense stereo matching using two-pass dynamic programming with generalized ground control points," IEEE International Conference on Computer Vision and Pattern Recognition, vol. 2, pp: 1075-1082, June 2005.

[9] N. Papadakis and V. Caselles, "Multi-label depth estimation for graph cuts stereo problems," Journal of Mathematical Imaging and Vision, vol. 38, no. 1, pp: 70-82, September 2010.

[10] F. Besse, C. Rother, and A. Fitzgibbon, "PMBP: Patch match belief propagation for correspondence field estimation," International Journal of Computer Vision, vol. 110, no. 1, pp: 2-13 October 2013.

[11] X. Mei, X. Sun, and M. Zhou, "On building an accurate stereo matching system on graphics hardware," IEEE International Conference on Computer Vision Workshops, pp. 467-474, November 2011.

[12] K. Yoon, S. Kweon, "Locally adaptive support weight approach for visual correspondence search," IEEE International Conference on Computer Vision and Pattern Recognition, vol. 2, pp: 924-931, June 2006.

[13] A. Hosni, M. Bleyer, and M. Gelautz, "Local stereo matching using geodesic support weights," IEEE International Conference on Image Processing, pp. 2093-2096, August 2009.

[14] K. He, J. Shun, and X. Tang, "Guided image filter," IEEE Pattern Analysis and Machine Intelligence, vol. 35, no. 6, pp: 1397-1409, June 2013.

[15] A. Hosni, C. Rhemann, and M. Bleyer, "Fast cost-volume filtering for visual correspondence and beyond," IEEE Pattern Analysis and Machine Intelligence, vol. 35, no. 2, pp: 504-511, February 2013.

[16] Q. Yang, P. Ji, D. Li, S. Yao, and M. Zhang, "Fast stereo matching using adaptive guided filtering," Image and Vision Computing, vol. 32, no. 3, pp: 202-211. March 2014.

[17] D. Scharstein, R. Szeliski, "Middlebury Stereo Vision Page," http://vision.middlebury.edu/stereo/.

[18] C. Pham, J. Jeon, "Domain transformation-based efficient cost aggregation for local stereo matching," IEEE Transactions on Circuits and systems for Video Technology, vol. 23, no. 7, pp: 1119-1130, July 2012.

[19] Q. Yang, "Hardware-efficient bilateral filtering for stereo matching," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 5, pp: 1026-1032, May 2014.

[20] L. Wang, R. Yang, "Global stereo matching leveraged by sparse ground control points," IEEE International Conference on Computer Vision and Pattern Recognition, pp. 3033-3040, June 2011.

[21] Y. Xu, Y. Zhao, M. Ji, "Local stereo matching with adaptive shape support window based cost aggregation," Applied optics, vol. 53, no.29, pp: 6885-6892, 2014.