

# Fast Dense Stereo Matching Using Adaptive Window in Hierarchical Framework

SangUn Yoon, Dongbo Min, and Kwanghoon Sohn

Dept. of Electrical and Electronic Eng., Yonsei University  
134 Shinchon-dong, Seodaemun-gu, Seoul, 120-749, Korea  
khsohn@yonsei.ac.kr

**Abstract.** A new area-based stereo matching in hierarchical framework is proposed. Local methods generally measure the similarity between the image pixels using local support window. An appropriate support window, where the pixels have similar disparity, should be selected adaptively for each pixel. Our algorithm consists of the following two steps. In the first step, given an estimated initial disparity map, we obtain an object boundary map for distinction of homogeneous/object boundary region. It is based on the assumption that the depth boundary exists inside of intensity boundary. In the second step for improving accuracy, we choose the size and shape of window using boundary information to acquire the accurate disparity map. Generally, the boundary regions are determined by the disparity information, which should be estimated. Therefore, we propose a hierarchical structure for simultaneous boundary and disparity estimation. Finally, we propose post-processing scheme for removal of outliers. The algorithm does not use a complicate optimization. Instead, it concentrates on the estimation of a optimal window for each pixel in improved hierarchical framework, therefore, it is very efficient in computational complexity. The experimental results on the standard data set demonstrate that the proposed method achieves better performance than the conventional methods in homogeneous regions and object boundaries.

## 1 Introduction

Stereo matching is one of the classical problems in computer vision. There are many potential applications including the robot navigation, teleconferencing, 3D modeling, and image based rendering, etc. The stereo matching problem is to compute the disparity map for the reference image using two or more images of the same scene. In binocular stereo, it is assumed that two input images are calibrated and rectified in advance, so that epipolar line becomes horizontal, but stereo matching is a difficult problem due to ambiguous region such as the occluded and textureless areas. To solve this problem, a number of algorithms have been proposed [1].

According to a recent taxonomy [1], stereo matching method can be roughly classified into two categories, namely, global and local methods. Global method define the smoothness constraint in order to resolve the ill-posed problem of

stereo matching, so that the problem of homogeneous regions can be handled successfully. Moreover, the discontinuity preserving smoothness constraint has been employed for the stereo model, and the energy function is minimized through various minimization techniques such as regularization method[9] and graph cut [11][12]. However, global methods have high computational complexity due to iterative scheme for energy minimizations and the complex constraints. Therefore, it is not appropriate to apply the method in real-time applications.

Local methods use dissimilarity between color or intensity in the neighboring window. It is implicitly assumed that all pixels in a support window are from similar depth in a scene. These methods can acquire correct results in highly textured regions, but often tend to produce noisy results in large textureless regions. Therefore, to obtain accurate results at depth discontinuities and in homogeneous regions, an appropriate window must be selected for each pixel.

Adaptive window algorithms [2][3][4] try to find optimal window for each pixels in accordance with changing the window size and shape adaptively. For this window cost, Kanade and Okutomi model [2] presented a method to select an appropriate window by evaluating the local variation of intensity and disparity. These methods are, however, highly dependent on the initial disparity estimation and is computationally expensive. Other popular methods are using the multiple windows [5][6][7]. For each pixel, a small number of different windows are evaluated, and the one with the best cost is retained. Generally, window size is constant, but shape is varied. This method has still some problems in homogeneous regions. A compact window algorithm [8] is one of the famous methods. Window cost is the average window error plus bias to larger windows. However, it is not efficient enough for real time implementation. Adaptive support weight approach [10] computes the matching cost based on the photometric and geometric relationship with the pixel under consideration. However, it is not efficient enough for real time approach due to the high computational complexity, either.

We propose a new adaptive window algorithm. The method estimates window sizes and shapes by using boundary information adequately. In order to estimate boundary and disparity information simultaneously, we use the hierarchical framework. The rest of the paper is organized as follows. First, we present the outline of proposed algorithm in section 2. Then, we explain how to obtain a boundary map, decide the direction of edges and post processing. Experimental results on various data sets are shown in section 4, and conclusions are drawn in section 5.

## 2 Outline of Proposed Algorithm

Generally, the performance of local methods depends on how well the window is selected adaptively at each pixel. The window must satisfy the following conditions [2].

- a. A window should be large enough to have sufficient intensity variation.
- b. A window should be small enough to contain pixels at approximately equal disparity only.

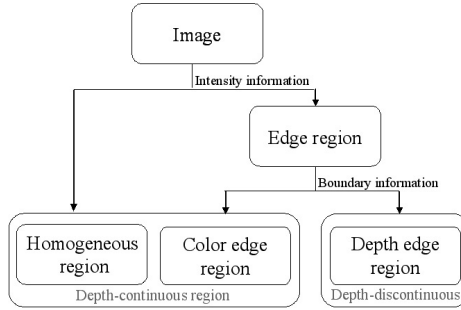


Fig. 1. Classification of Textureless and boundary region

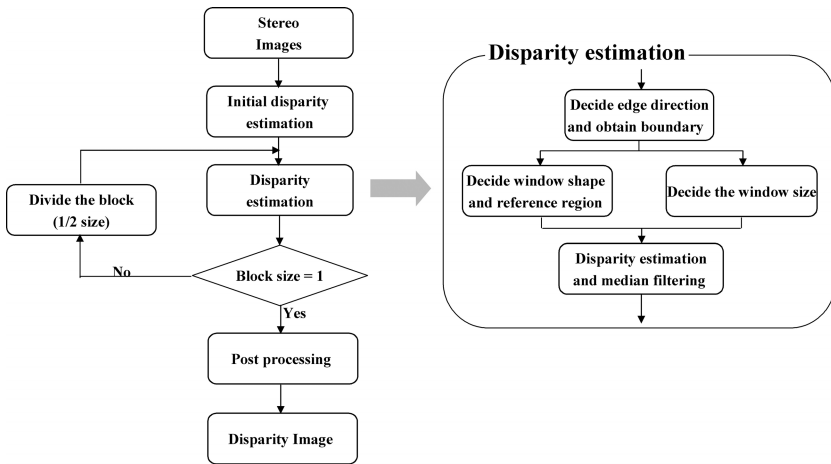


Fig. 2. The flow chart of proposed algorithm

As window size is increased from small to large, the results range from accurate disparity boundaries but noisy in low textured areas, to more reliable in low textured areas but blurred disparity boundaries. It is very difficult to obtain the optimal window for each pixel, which are reliable in both low textured areas and object boundaries. We classify an image into the depth-discontinuous and continuous region, and different approaches are used in each region as shown in Fig. 1.

To solve the matching problem in homogeneous regions, a window size increases until the window includes sufficient texture information or encounters the boundary map. Fig. 2 shows the flow chart of the proposed algorithm. The key technique in the process is the estimation of optimal window and boundary map. In boundary regions, the window is defined as 9 different shape models. Fig. 3 shows the 9 window models for each direction. Each window is decided according to the direction of the corresponding edge. The proposed algorithm is different from multiple window method [5]: Multiple window method changes the



Fig. 3. Definition of 9 window models

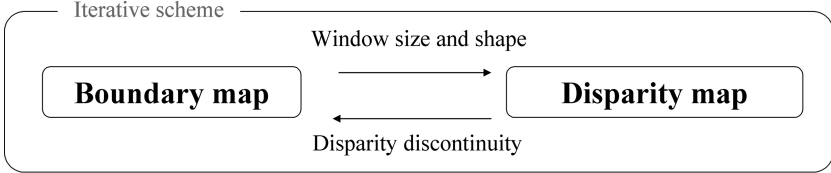


Fig. 4. Relation of disparity and boundary information

location of pixel and select the best position in the window. On the other hand, the proposed window method chooses one among the 9 window models by using direction of an edge. Moreover, unlike the multiple window using fixed window shape, window shape is different according to the edge direction. It makes the algorithm robust to object boundary regions.

The boundary information is necessary for optimal window selection. According to the boundary information, an image can be divided into the homogeneous and object boundary regions, direction of each pixel is calculated at boundary regions. Generally, the boundary regions are determined by the disparity information, which should be estimated, as shown in Fig. 4. Thus, to estimate the boundary and disparity information simultaneously, we use a hierarchical framework.

Fig. 5 shows the shape and size of window in 'tsukuba' image. We can confirm that an optimal window is estimated at each pixel adequately. The sum of absolute difference (SAD) is used as the matching cost and the disparity is finally selected by Winner-Takes-All(WTA) method in Eq (1).

$$E(x, y, d) = \frac{1}{||N(x, y)||} \sum_{(x', y') \in N(x, y)} |I_l(x', y') - I_r(x' + d, y')|$$

$$\hat{d}(x, y) = \arg \min_d E(x, y, d), \tag{1}$$

where  $I_l$  and  $I_r$  are input images and  $N(x, y)$  represent an optimal window for each pixel. Finally, we obtain final disparity map through the proposed post-processing schemes.

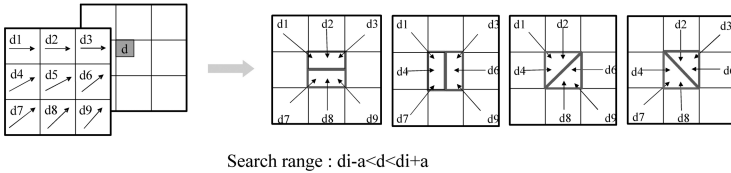
### 3 Implementation

#### 3.1 Improved Hierarchical Framework

In order to obtain boundary and disparity map simultaneously, we use hierarchical framework. The basic idea of a hierarchical approach is to start at the



**Fig. 5.** Adaptive window : adaptive window in Tsukuba image



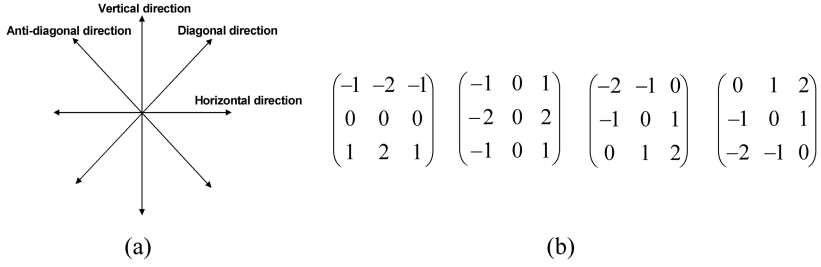
**Fig. 6.** Reference regions in hierarchical framework

highest level with a large measurement window and a maximum search range. Only a set of candidate displacement vectors from the higher level is considered at a lower level in order to decrease the potential displacements to be taken into account. The dimension of the measurement window is decreased as well. In order to keep implementation costs as low as possible, displacement vectors are only estimated with respect to some selected sampling positions at the highest level. However, in contrast with the conventional approach, we decide candidate vectors selectively according to the direction of each block and search the image to obtain optimal disparity near the candidate disparity value, as shown in Fig. 6. It is possible to enhance the accuracy of disparity map and reduce the computational complexity in this way.

### 3.2 Boundary Region Detection

It is possible for an image to classify edge region and homogeneous region using intensity information. However, edge region includes both color edge and depth edge. Since color edge regions are not object boundaries, we divide them according to disparity information as shown in Fig. 1. We use different methods at each class for accurate stereo matching. To estimate the direction of edges, we use the texture information for each block. We define four types of edge directions (horizontal, vertical, diagonal and anti-diagonal ) as shown in Fig. 7 (a). To determine the type of block, we utilize four ‘Sobel’ masks, as shown in Fig. 7 (b). For each region, four types of ‘Sobel’ masks are applied and the directional type with the largest summation of absolute ‘Sobel’ value is selected. After four ‘Sobel’ operations, we sum the absolute ‘Sobel’ values of all the directional types in Eq. (2). We use this information to decide the location of the homogeneous region[13].

$$Sum = Sum_h + Sum_v + Sum_d + Sum_{ad}, \tag{2}$$



**Fig. 7.** Definition of direction and four directional Sobel mask

where  $Sum_h$ ,  $Sum_v$ ,  $Sum_d$  and  $Sum_{ad}$  represent the sums of absolute 'Sobel' values in each direction, respectively.

The region of block is considered as a homogeneous region when the Sum is less than the threshold ( $E_{th}$ ). In order to consider the characteristics of an image, the value of  $E_{th}$  is computed as follows:

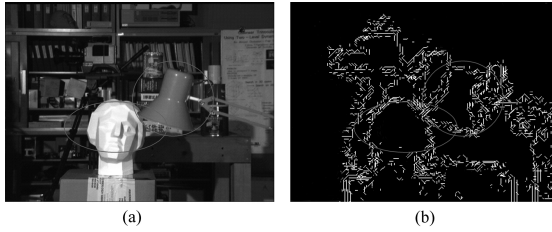
$$E_{th} = \frac{1}{N_x N_y} \sum_{(i,j) \in I_r} ((I_h(i,j) + I_v(i,j) + I_d(i,j) + I_{ad}(i,j)) \times c, \quad c \in [0, 1], \quad (3)$$

where  $I_h(i,j)$ ,  $I_v(i,j)$ ,  $I_d(i,j)$  and  $I_{ad}(i,j)$  are the output values for the each directional 'Sobel' masks.  $N_x$ ,  $N_y$  indicate the image size, respectively. By adjusting the constant value  $c$ , we can control the proportion of the homogeneous region. Fig. 8 shows the results of boundary detection. The proper directions are correctly estimated especially in 'head' and 'lamp'.

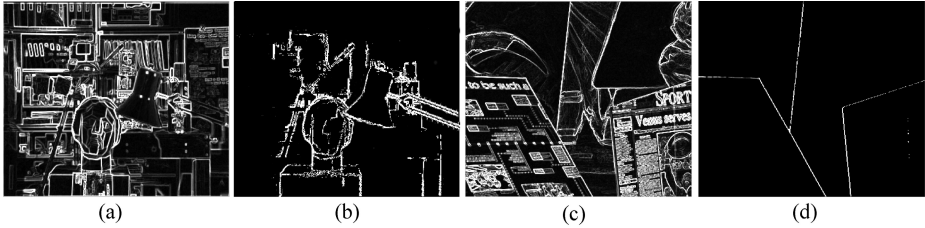
The boundary information is updated in hierarchical framework. According to the information of edge direction and boundary, adaptive window size/ shape and the reference region are computed to execute accurate and fast disparity estimation. In boundary regions, to reduce 'overfitting' problem in the object boundary, we use adaptive shape windows. In homogeneous region, the different size of window is defined to overcome the problems in the textureless regions.

$$\begin{aligned} & \text{the pixel is boundary if } \frac{Max(diff_1, diff_2)}{Min(diff_1, diff_2)} > B_{th} \\ & diff_1 = |d' - d|, diff_2 = |d - d'| \begin{cases} \text{horizontal: } d'(i,j) = d(i, j \pm k), \\ \text{vertical: } d'(i,j) = d(i \pm k, j) \end{cases} \end{aligned} \quad (4)$$

where  $diff_1$  and  $diff_2$  are the difference between disparity of neighboring pixels and  $k$  is a window size in each level. The region of block is decided as object boundary when the rate of difference is more than the threshold ( $B_{th}$ ). The process is repeated in each level, and we can obtain the boundary of a reference image as shown in Fig. 9.



**Fig. 8.** 4x4 block directions at boundary regions: (a) Tsukuba reference image (b) Four direction



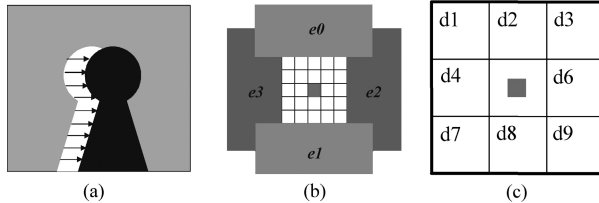
**Fig. 9.** Edge map and boudnary map: (a)(b) Tsukuba image, (c)(d) Venus image

### 3.3 Post Processing

According to the selection of adequate window in each region, problems of local method are reduced considerably. Nevertheless, there are still some problems in occluded and textureless regions such as 'foreground-fatting' problem in boundary region. In this section, we propose a post-processing scheme for improving the performance.

#### Occlusion handling

The occluded region exist in depth discontinuous regions nearby an object and disparity of this region is filled with the false values. Therefore, we should assign exact disparity values in occlusion region using boundary map. Disparity discontinuity exists at object boundary and occlusion regions are the left side of the object. Thus, disparities of the occluded regions is filled with the value of left side disparity at occluded regions as shown in Fig. 10 (a).



**Fig. 10.** Post processing : (a) Occlusion handling (b) Definition of boundary regions in a block (c) Definition of reference region

### Disparity regularization

Although we use adaptive window, some outliers still exist in homogeneous regions. Thus, we propose a filter that calculate the optimal disparity value in the neighborhood disparities at the outlier while preserving the disparity discontinuity at object boundary. The proposed filter decides the outlier with mean ( $win_m$ ) and variance ( $win_v$ ) inside fixed window by equalization as shown in Eq(5), and correct value is computed as shown in Eq(6).

$$win_m + win_v < d \quad or \quad win_m - win_v > d \quad (5)$$

$$d_c = \arg \min_{d_k} \{ (sum_e + sum_{ei}) \times |m_d - m_i| \} \quad i = e_0, e_1, e_2, e_3, \quad (6)$$

where  $m_d$  is mean of disparities in a window and  $m_i$ s' are means of disparities (0,1,2 and 3 indicate top, bottom, left and right direction, respectively) as shown in Fig. 10(b).  $sum_{ei}$  denote the amount of edges at same place in boundary region of window and  $sum_e$  represent the amount of edge in a window.  $d_k$ s' are the candidate disparity vectors in neighborhood regions.  $d$  is exchanged by optical disparity,  $d_c$ . In boundary regions, due to disparity discontinuity, we use intensity values for the discrimination of best disparity at boundary, instead of window mean and variance. According to the boundary direction, we decide candidate disparities and check the optimal disparity value using Eq (7)

$$d_c = \arg \min_{d_k} (|I_l(i, j) - I_r(i, j + d_k)|), \quad (7)$$

where  $d_k$ s' are candidates of disparities in referred regions as shown in Fig (10)(c) ([ horizontal: $d_2, d_8$  ], [ vertical: $d_4, d_6$  ], [ diagonal: $d_1, d_9$  ], [ anti-diagonal: $d_3, d_7$  ] ) .

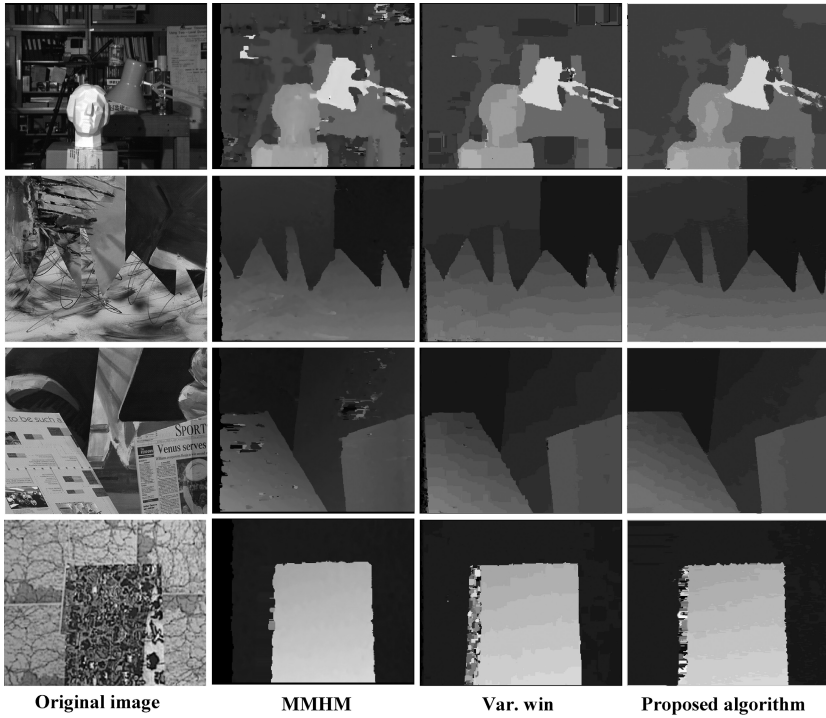
## 4 Experimental Results

To evaluate the performance of our approach, we used a test bed proposed by Scharstein and Szeliski [1]. We evaluated the proposed algorithm on these test data sets, Tsukuba, Venus, Sawtooth and Map. For all the experiments, we set  $c=0.5$ ,  $B_{th}=10$ , and minimum and maximum window sizes to 4 by 4 and 32 by 32 squares.

The performance of the proposed algorithm is measured by the percentages of bad matching (where the absolute disparity error is greater than 1 pixel). This measure is calculated in three different parts of an input image including the entire image(all), untextured (untex), and discontinuities (disc) regions.

Fig. 11 shows the results of stereo matching for the 4 standard stereo images. For comparison, we include the results of window based methods. The results show that the proposed algorithm achieves good performance in conventionally challenging areas such as object boundaries and untextured regions. Especially, the results of ambiguous regions and the preserving of disparity discontinuity are better than the conventional methods. This is due to the efficient selection of adaptive window and post-processing according to boundary map. The running times for the Tsukuba, Sawtooth, Venus, and Map scenes are 1, 2, 2, 1 seconds,





**Fig. 11.** Results for (from top to bottom) Tsukuba, Sawtooth, Venus, and Map image pairs

**Table 1.** Comparative performance of algorithms

	Tsukuba			Sawtooth			Venus			Map	
	all	untex	disc	all	untex	disc	all	untex	disc	all	disc
Proposed method	2.25	1.58	12.19	0.77	0.23	6.76	0.86	0.28	6.49	0.40	5.18
Adapt. weights[10]	1.51	0.65	7.24	1.14	0.27	5.48	1.14	0.61	4.49	1.47	13.58
Var. win[4]	2.35	1.65	12.17	1.28	0.23	7.09	1.23	1.16	13.35	0.24	2.98
Graph cut[11]	1.94	1.09	9.49	1.30	0.06	6.34	1.79	2.61	6.91	0.31	3.88
Tree DP[14]	1.77	0.38	9.48	1.44	0.84	6.87	1.21	1.41	5.04	1.45	13.00
Comp. win[8]	3.36	3.54	12.91	1.61	0.45	7.87	1.67	2.18	13.24	0.33	3.94
MMHM[15]	9.76	13.85	24.39	4.76	1.87	22.49	6.48	10.36	31.29	8.42	12.68

respectively on Pentium IV 3.0 GHz. Table 1. shows quantitative results for the stereo images using true maps. The results show that the proposed method has comparable performance with state-of-the-arts.

## 5 Conclusion

In this paper, we presented a new stereo matching algorithm using adaptive window and boundary map. By efficiently using the advantage of boundary

information with adaptive window, we could acquire comparable performance with state-of-the-arts.

The boundary map improves the performance of the proposed algorithm. We plan to investigate the more accurate boundary extraction method using disparity information with intensity value. We are also examining the real time computational issues related to the proposed technique.

## Acknowledgement

This research was partially supported by the MIC, Korea, under the ITRC support program supervised by the IITA (IITA-2005-(C1090-0502-0027)) and was partially supported by the MOE, MOCIE and the MOLAB through the fostering project of the Lab of Excellency.

## References

1. D. Scharstein and R. Szeliski: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *IJCV*, Vol. 47 (2002) 7-42.
2. T. Kanade and M. Okutomi: A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiments, *PAMI*, vol. 16, no. 9 (1994) 920-932.
3. Y. Boykov., O. Veksler, and R. Zabih: A Variable Window Approach to Early Vision, *PAMI*, vol. 20, no. 12, (1998) 1283-1294.
4. O. Veksler.: Fast Variable Window for Stereo Correspondence using Integral Images, *CVPR*, vol. 1 (2003) 556-561.
5. A. Fusiello, V. Roberto and E. Trucco.: Efficient Stereo with Multiple Windowing, *CVPR*, (1997) 858-863.
6. A.F. Bobick and S.S. Intille.: Large Occlusion Stereo, *IJCV*, vol. 33, no. 3 (1999) 181-200.
7. S. B. Kang, R. Szeliski, and C. Jinxjang.: Handling Occlusions in Dense Multi-View Stereo, *CVPR*, vol. 1 (2001) 103-110.
8. O. Veksler.: Stereo matching by compact windows via minimum ratio cycle, *ICCV01*, (2001) 540-547.
9. H. Kim, Y. Choe and K. Sohn.: Disparity estimation using region-dividing technique with energy-based regularization, *Optical Engineering*, vol. 43, no. 8, pp 1882-1890, Aug. 2004.
10. K.-J. Yoon and I.-S. Kweon.: Locally Adaptive Support- Weight Approach for Visual Correspondence Search, *CVPR*, Vol.2 (2005) 924-931.
11. Boykov.Y., Veksler.O., Zabih.R: Fast approximate energy minimization via graph cut, *PAMI23*, (2001) 1222-1239.
12. Kolmogorov, V., Zabih, R.: Computing visual correspondence with occlusions using graph cuts, *ICCV01*, (2001) 508-515.
13. Y. Kim, J. H. Lee, C. Park and K. Sohn.: MPEG-4 compatible stereoscopic sequence CODEC for stereo broadcasting, *IEEE Trans. on Consumer Electronics*, vol. 51, no. 4, pp. 1227-1236 Nov. 2005.
14. O. Veksler.: Stereo correspondence by dynamic programming on a tree, *CVPR*, Vol. 2 (2005) 20-25.
15. K. Muhlmann, D. Maier, J. Hesser, and R. Manner.: Calculating dense disparity maps from color stereo images, an efficient implementation, *SMBV*, (2001) 30-36.