# STEREO MATCHING WITH ASYMMETRIC OCCLUSION HANDLING IN WEIGHTED LEAST SQUARE FRAMEWORK

*Dongbo Min and Kwanghoon Sohn*

Dept. of Electrical and Electronic Eng., Yonsei University, Seoul, Korea
khsohn@yonsei.ac.kr

**ABSTRACT**

This paper presents a novel method for stereo matching with occlusion handling. In order to estimate optimal cost, we define an energy function and solve the iterative equation with the numerical method. We improve performance and convergence rate by using several acceleration techniques. The proposed method is computationally efficient since it does not use color segmentation or any global optimization techniques. For occlusion handling, which has not been performed effectively by any conventional cost aggregation approaches, we combine the occlusion problem with the proposed minimization scheme. Asymmetric information is used so that few additional computational loads are necessary. Experimental results show that performance is comparable to that of many state-of-the-art methods.

***Index Terms***— Cost aggregation, multiscale approach, stereo vision, occlusion handling, weighted least square

## 1. INTRODUCTION

For decades, the correspondence problem has been an important issue in the field of computer vision, and many methods have been proposed to solve this problem. An extensive review of stereo matching algorithms can be found in [1]. Local approaches use correlations between color or intensity patterns in neighboring windows. Performance depends on how the optimal window is selected in each pixel, but finding an optimal window with an arbitrary shape and size is very difficult. To solve this problem, a number of methods have been proposed.

In general, adaptive window algorithms try to find optimal windows for each pixel by adaptively changing the window size and shape. Kanade and Okutomi [2] proposed a way of selecting an appropriate window by evaluating the local intensity and disparity variations. Yoon et al [3] used boundary information to compute accurate windows for each pixel. Multiple window algorithms [4] used a small number of different windows, whose reference points lie in several positions. Yoon and Kweon [5] proposed a general method that computes an optimal local support window. However, it is very computationally expensive to perform pixel-wise support weight computation.

Another issue discussed in this paper is occlusion handling. Several constraints have been used in stereo matching for occlusion handling. Bobick [6] exploited the ordering constraint by using dynamic programming. This approach is very efficient but the ordering constraint is invalid when an image has a thin object. Most approaches have used global optimization schemes to detect the occlusion regions by using uniqueness constraint, and assign pre-defined values to the occluded pixels [7].

This research was supported by the MIC, Korea, under the ITRC support program supervised by the IITA (IITA-2005-(C1090-0502-0027))

**Fig. 1**. Per-pixel and estimated costs in the 'Tsukuba' image, when disparity is 0.

In this paper, we propose a novel approach of performing efficient cost aggregation and handling occlusion for stereo matching. To estimate the optimal cost, we define an energy function and solve a corresponding iterative equation with several acceleration techniques. We combine the occlusion problem, which has not been solved by any existing cost aggregation approaches, into the iterative scheme. It is not necessary to define a pre-defined value for the occluded pixel and it is possible to use asymmetric information with trivial additional computational loads, that is, only the left disparity field needs to be used.

## 2. PROPOSED COST AGGREGATION

### 2.1. Problem statement

When estimating the disparity field, only the left and right image pairs are used. We obtain the difference image by shifting the right image further to the right, and then subtracting the left and the shifted right images. This is done for all disparities. A set of difference images is called 3D cost volume $e(p, d)$, where $p$ and $d$ represent the 2D locations of pixels and disparity, respectively. We call the 3D cost volume a per-pixel cost. In order to estimate the optimal cost, we define the per-pixel cost $e$ as follows:

$$e(p, d) = E(p, d) + n \tag{1}$$

, where $n$ represents noise. We simplify $E(p, d)$ to $E(p)$, since the same process is performed for each disparity. Fig. 1 shows the per-pixel and estimated costs for the 'Tsukuba' image. Given the observation data, we use the prior knowledge that costs should vary smoothly, except at object boundaries. From this observation, we are able to estimate the cost function by minimizing the following energy model with anisotropic diffusion term:

$$\varepsilon(E) = \int_{\Omega} (E(p) - e(p))^2 \, dp + \lambda_d \int_{\Omega} g(|\nabla I|) |\nabla E|^2 dp \tag{2}$$

, where $g(|\nabla I|)$ decreases monotonically with respect to $|\nabla I|$. This is known as the diffusivity function, which plays the role of a discontinuity marker. The minimization of Eq. (2) yields the following Euler-Lagrange equation. We obtain the solution to the equation by calculating the asymptotic state ($t \rightarrow \infty$) of the parabolic system, as shown in Eq. (3).

$$\frac{\partial E}{\partial t} = -E(p) + e(p) + \lambda_d \nabla \cdot (g(|\nabla I|) \nabla E) \tag{3}$$

**Fig. 2**. Per-pixel and estimated costs when disparity is 0, and estimated disparity map.

The final solution can be found in a recursive manner with numerical method. Fig. 2 shows the results of the estimated cost with nonlinear diffusion filtering, and the disparity field computed with these cost functions. To evaluate the performance of the cost aggregation scheme, we use WTA (Winner-Takes-All) method. We find that there are some problems in the textureless and occluded regions, although edge-preserving smoothing is performed. In Eq. (1), the optimal cost model considers sensor noise only. Thus, another strategy to solve these problems is necessary.

### 2.2. Cost Aggregation with Weighted Least Square

To solve the problem in textureless regions, we consider using the smoothness constraint with more neighborhoods. As opposed to image restoration/denoising, it is necessary to gather sufficient texture in the neighborhoods for reliable matching. To include more neighborhoods, we propose a new energy function with the weighted least square method:

$$\begin{aligned}&\varepsilon(E) = \textstyle\int_\Omega (E(p) - e(p))^2\, dp \\ &+\lambda \textstyle\int_\Omega \sum_{n\in N_1} \left\{ \begin{array}{c} w_{p,p+n}(E(p) - E(p+n))^2 \\ +w_{p,p+n^\perp}(E(p) - E(p+n^\perp))^2 \end{array} \right\} dp \\ &N_1 = \{(x_n, y_n) | 0 < x_n \le M,\ \ 0 \le y_n \le M\}\end{aligned} \tag{4}$$

, where $w$ represents the weighting function between corresponding neighbor pixels. $n$ and $n^\perp$ represent the 2D vectors, which are perpendicular to each other. $M$ represents the size of a set of neighbor pixels. When the element of the set $N_1$ is $(1, 0)$ only, Eq. (4) is similar to anisotropic diffusion. In other words, Eq. (4) can be considered to be the generalized function of Eq. (2). Taking the first derivative of Eq. (4) with respect to $E$, we yield the following equation:

$$\begin{aligned}&E(p) - e(p) \\ &+\lambda \sum_{n\in N_1} \left\{ \begin{array}{l} w_{p,p+n}(E(p) - E(p+n)) \\ -w_{p-n,p}(E(p-n) - E(p)) \\ +w_{p,p+n^\perp}(E(p) - E(p+n^\perp)) \\ -w_{p-n^\perp,p}(E(p-n^\perp) - E(p)) \end{array} \right\} = 0\end{aligned} \tag{5}$$

To simplify the above equation, we redefine the set of neighbor pixels. When $p$ is $(x, y)$, the set can be expressed as:

$$N(p) = \{p + p_n | -M \le x_n, y_n \le M,\ \ x_n + y_n \ne 0\}$$

By using the above notation, Eq. (5) is expressed as:

$$E(p) - e(p) + \lambda \sum_{m\in N(p)} w_{p,m}(E(p) - E(m)) = 0 \tag{6}$$

The solution of the $(k+1)^{th}$ iteration is obtained by the following equation:

$$\begin{aligned}E^{k+1}(p) &= \bar{e}(p) + \bar{E}^k(p) \\ &= \frac{e(p) + \lambda \sum_{m\in N(p)} w_{p,m} E^k(m)}{1 + \lambda \sum_{m\in N(p)} w_{p,m}}\end{aligned} \tag{7}$$

Eq. (7) consists of two parts: normalized per-pixel matching cost and weighted neighboring pixel cost. By running the iteration scheme, the cost function $E$ is regularized with the weighted neighboring pixel cost. In the proposed method, we use the symmetric Gaussian weighting function with the CIE-Lab color space in Eq. (8). $r_c$ and $r_s$ are weighting constants for the color and geometric distances, respectively. As opposed to $g(|\nabla I|)$ in Eq. (2), it is necessary to use the term for geometric distance in the weighting function, since the smoothness constraints with more neighborhoods are considered.

$$\begin{aligned}&w_{p,m} = \exp\left(-\left(\tfrac{C^L_{p,m}}{2r_c^2} + \tfrac{C^R_{p,m}}{2r_c^2} + \tfrac{S_{p,m}}{2r_s^2}\right)\right) \\ &C_{p,m} = (L_p - L_m)^2 + (a_p - a_m)^2 + (b_p - b_m)^2 \\ &S_{p,m} = (p - m)^2\end{aligned} \tag{8}$$

### 2.3. Acceleration Scheme

#### *2.3.1. Gauss-Seidel Acceleration*

One reason for slowing down the convergence in Eq. (7) is that the updated components in each pixel are used only after one iteration is complete. We compensate for this problem by using the updated components in each pixel intermediately. We divide a set of neighbor pixels $N(p)$ into two parts: the causal part $N_c(p)$ and the non-causal part $N_n(p)$. Eq. (7) is expressed as follows, based on this relationship:

$$\begin{aligned}&E^{k+1}(p) = \bar{e}(p) + \bar{E}^k(p) \\ &= \frac{e(p) + \lambda \sum_{m\in N_c(p)} w_{p,m} E^{k+1}(m) + \lambda \sum_{m\in N_n(p)} w_{p,m} E^k(m)}{1 + \lambda \sum_{m\in N(p)} w_{p,m}}\end{aligned} \tag{9}$$

#### *2.3.2. Multiscale Approach*

As previously mentioned, it is necessary to gather pixel information at a large distance to ensure reliable matching. This implies that a number of iterations are required to estimate the correct cost function. We use a multiscale approach to solve this problem. Our method is different from the conventional approaches in the sense that it is applied in the cost domain. In Eq. (9), the cost function $E(p)$ can generally be initialized to $e(p)$. We can initialize the value close to the optimal cost in each level by using the final value in the coarser level.

Using Eq. (9), the proposed method performs cost aggregation independently in each section with the same disparity of the 3D cost volume. Conventional multiscale approaches reduce image resolution at first, and then the estimation process continues. The reduction of the resolution also reduces the search range of the disparity. For instance, if we use the multiscale approach over three levels, the search range will have been reduced to a quarter of the original search range on the coarsest level. Thus, two cost functions in the finer level $E_f(p, 2d)$ and $E_f(p, 2d+1)$ are initialized by using the cost function in the coarser level $E_c(p, d)$. To avoid this problem, we use an alternative multiscale scheme for cost aggregation. We first compute the 3D cost volume and then perform the proposed multiscale scheme in each 2D cost function. The proposed multiscale method runs the iterative scheme at the coarsest level by initializing the cost function to $e(p, d)$. After $K$ iterations, the resulting cost function is used to initialize the cost function in the finer level, and this process is repeated until the finest level is reached. The proposed multiscale scheme is shown in Fig. 5, which includes adaptive interpolation and occlusion handling.

When the cost function on the $(l+1)^{th}$ level is defined as $E_{l+1}(p)$, we can refine the resolution of the cost function $E_l(p)$ on the finer level by using bilinear interpolation. However, if bilinear interpolation is used, the error can be propagated into the neighborhood regions, especially on the boundary region. To avoid this problem, we propose an adaptive interpolation method based on the weighted least square:

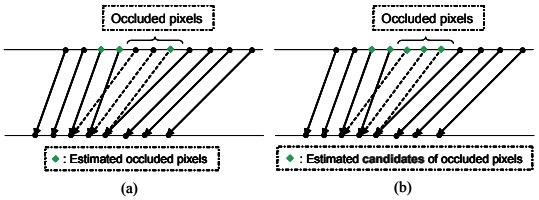


**Fig. 3**. Several cases of asymmetric occlusion detection: (a) using the geometric constraints only, (b) using both the geometric and photometric constraints.

$$E_l(p) = \frac{e_l(p) + \lambda_a \sum_{p_m \in N(p_i)} w_{p,p_m} E_{l+1}(p_m)}{1 + \lambda_a \sum_{p_m \in N(p_i)} w_{p,p_m}} \quad (10)$$

, where $p_i = (x_i, y_i)$ represents a pixel on the coarser level, and $N(p_i)$ on the $(l+1)^{th}$ level is a set of 4-neighboring pixels. In Eq. (10), $w$ represents the weighting function, equivalent to that in Eq. (7). We set the weighting factor to $\lambda_a = 15$. Another advantage of adaptive interpolation is to increase the resolution of the cost function so that no blocking artifact exists. The adaptive interpolation by the intensity values on two successive levels leads to the up-sampling scheme, which preserves the discontinuities on the boundary region. Thus, it is not necessary to perform the cost aggregation scheme on the finest level, and this makes the proposed method faster. In the experimental results, we will show that adaptive interpolation increases the resolution of the cost function without requiring any blocking artifact.

## 3. OCCLUSION HANDLING

Most approaches have used an iterative scheme which combines the uniqueness constraint into a global optimization method for occlusion handling. In this section, we introduce a new approach for dealing with the occlusion problem in the proposed cost aggregation scheme. Only the left disparity field, and not a pre-defined value for the occluded pixels, is used in the occlusion handling.

Our main goal is not to detect the occluded pixels in an image correctly but to determine a candidate set of occluded pixels. Then, reasonable cost functions are assigned in the candidate set. Although some visible pixels may be contained in the candidate set, this problem is solved by using the proposed cost aggregation. For asymmetric occlusion detection, we use geometric and photometric constraints. To determine whether a pixel is visible or not, we have to evaluate the disparity values of the neighboring pixels. The disparity of the occluding pixels is larger than that of the occluded pixels. Before defining the visibility function, we describe the function $S_r(j)$ as a set of pixels in the right image:

$$S_r(j) = \{i | i - d(i) = j, \ \ all\ i\ with\ \ 0 \le i \le W-1\}$$

, where $i$ and $j$ represent the $x$ coordinates of the left and right images, respectively. $W$ represents the width of the image and $d$ represents the disparity of the pixel. When there are multiple matching points at pixels in the other image, that is, $\#(S_r(j)) \ge 1$, the pixel with the largest disparity among $S_r(j)$ is considered as visible and the remaining pixels are considered occluded. This is valid only if the occluding pixels have reliable disparities. Fig. 3 shows several cases of asymmetric occlusion detection. If the disparities in the occluded pixels are larger than those of the visible pixels, the occluded pixels block the other visible pixels, as shown in Fig. 3(a). Thus, we use the photometric constraint to evaluate the reliability of the occluding pixels. The costs at the occluded pixels are generally larger than those of the visible pixels. If the cost at the pixel, which is determined as occluding pixels by geometric constraints, is not smaller

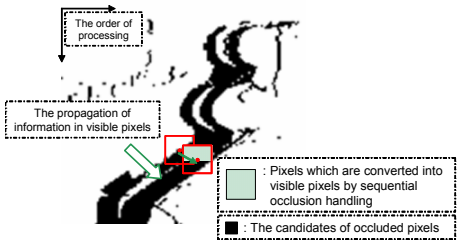


**Fig. 4**. Propagation of information at the visible pixels in sequential occlusion handling.

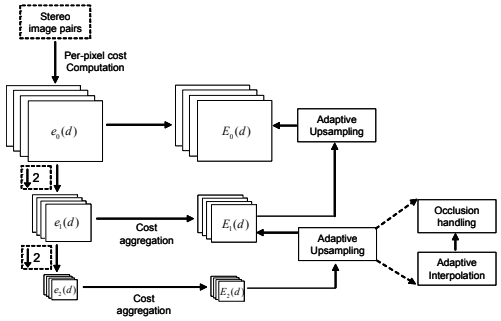


**Fig. 5**. Overall framework of the proposed method.

than that of the remaining occluded pixels, we can not guarantee the reliability of the occluding pixels. Therefore, all the pixels in $S_r(j)$ are used as occlusion candidates, as shown in Fig. 3(b). We determine a set of occlusion candidates instead of a set of occlusions by using this constraint. We define visibility function $O_l$ which takes the value 1 (or 0) when the pixel is visible (or occluded). By using $O_l$, we redefine the iterative scheme in Eq. (9) as follows:

$$E^{k+1}(p) = \frac{O_l(p)e(p) + \lambda \sum_{m \in N_c(p)} O_l(m) w_{p,m} E^{k+1}(m) + \lambda \sum_{m \in N_n(p)} O_l(m) w_{p,m} E^{k}(m)}{O_l(p) + \lambda \sum_{m \in N(p)} O_l(m) w_{p,m}} \quad (11)$$

The overall process of the proposed occlusion handling method is as follows. When the 3D cost volume is given, we are able to estimate the disparity by using an optimization method, and perform proposed occlusion handling with the estimated disparity. In this paper, we only use the WTA method to evaluate performance of proposed cost aggregation. However, other techniques can be used in the proposed method. If we use belief propagation for disparity estimation at each level, the message computed at each specific level can be used to initialize the message of the finer level. This scheme is very similar to hierarchical belief propagation [10].

Occlusion handling is sequentially performed. After the cost aggregation scheme is performed at the pixels of the set of occlusion candidates, the pixels become visible, in other words, $O_l(p) = 1$. This is very reasonable for occlusion handling since the occluded pixels aggregated with the visible pixels are used as visible pixels again in Eq. (11). Fig. 4 shows the process of sequential occlusion handling. The information of the visible pixels is propagated to estimate the cost function at the occluded pixels.

## 4. EXPERIMENTAL RESULTS

### 4.1. Overall Framework & Experimental Environments

The basic framework of the proposed method is to perform cost aggregation in a coarse-to-fine manner. Fig. 5 shows the overall process of the proposed method. In order to initialize the cost function in the finer level, adaptive interpolation is performed with Eq. (10), and then occlusion handling is performed (once at each level). This process is repeated until the finest level is reached.

**Table 1**. Objective evaluation for the proposed method with the Middlebury test bed

| Algorithm | *Tsukuba* | | | *Venus* | | | *Teddy* | | | *Cone* | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | nonocc | all | disc | nonocc | all | disc | nonocc | all | disc | nonocc | all | disc |
| AdaptingBP [8] | 1.11 | 1.37 | 5.79 | 0.10 | 0.21 | 1.44 | 4.22 | 7.06 | 11.8 | 2.48 | 7.92 | 7.32 |
| SymBP+occ [7] | 0.97 | 1.75 | 5.09 | 0.16 | 0.33 | 2.19 | 6.47 | 10.7 | 17.0 | 4.79 | 10.7 | 10.9 |
| **Our method** | **1.36** | **1.93** | **7.19** | **0.55** | **1.26** | **5.84** | **6.90** | **12.1** | **17.5** | **3.60** | **8.57** | **9.36** |
| OverSegmBP [9] | 1.69 | 1.97 | 8.47 | 0.50 | 0.68 | 4.69 | 6.74 | 11.9 | 15.8 | 3.19 | 8.81 | 8.89 |
| AdaptWeight [5] | 1.38 | 1.85 | 6.90 | 0.71 | 1.19 | 6.13 | 7.88 | 13.3 | 18.6 | 3.97 | 9.79 | 8.26 |

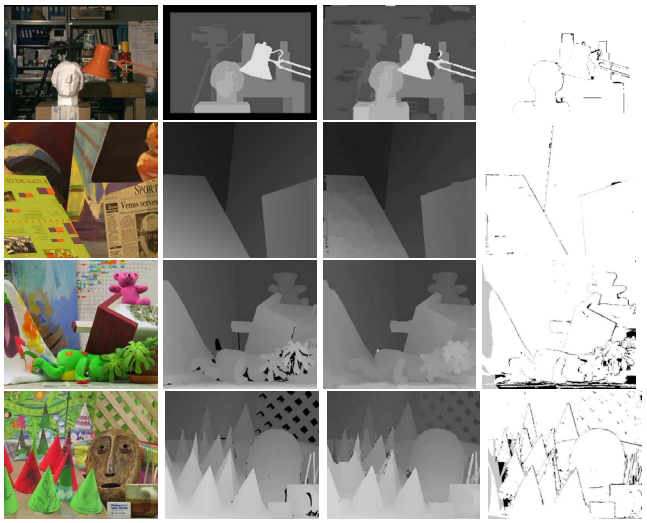

**Fig. 6**. Results for (from top to bottom) 'Tsukuba', 'Venus', 'Teddy' and 'Cone' image pairs: (from left to right) original images, ground truth maps, our results, error maps.

We evaluate the performance of the proposed method and compared it with state-of-the-art methods in the Middlebury test bed [11]. The results for each test dataset are evaluated by measuring the percentage of bad matching pixels (where the absolute disparity error is greater than 1 pixel). The measurement is computed for three subsets of an image: nonocc (the pixels in the non-occluded regions), all (the pixels in both the non-occluded and half-occluded regions), and disc (the visible pixels near the occluded regions).

The proposed method is tested using the same parameters for all the test images. The two parameters in the weighting function are $r_c = 8.0$, $r_s = 8.0$, and the weighting factor is $\lambda = 1.0$. We use the multiscale approach at four levels, and the number of iterations is $(3, 2, 2, \times)$, on a coarse to fine scale. The iteration number of the finest level is not defined since we use the adaptive interpolation technique in the up-sampling step, as mentioned in section 2.4.2. The sizes of the sets of neighbor pixels are $5 \times 5$, $7 \times 7$, $9 \times 9$, and $9 \times 9$. In the finest level, only occlusion handling is performed.

### 4.2. Performance Analysis

Fig. 6 shows the results of the proposed method for the test bed images. The proposed method yielded accurate results for the discontinuity, occluded, and textureless regions. Table 1 shows that the proposed method obtained comparable performance with state-of-the-art methods. Fig. 7 shows the results obtained by the proposed occlusion handling method. The occlusion candidate set contained as many occluded pixels as possible in order to perform occlusion handling well. The proposed occlusion handling method is different from the extrapolation technique widely used for occlusion handling. While the extrapolation technique is just filling by using the disparities of the visible pixels, the proposed method propagates the information of the visible pixels into that of the occluded pixels. This is very similar to the concept of edge-preserving nonlinear diffusion. Fig. 8 shows the intermediate results of the multiscale approach. Since the cost function in each level was obtained after performing adaptive interpolation, the cost function was considered as that in the finer level. We found that the estimated disparity map in level 1 had the finest resolution as shown in Fig. 8(c).

**Fig. 7**. Disparity map before handling, occlusion candidate, disparity map after handling (from left to right).

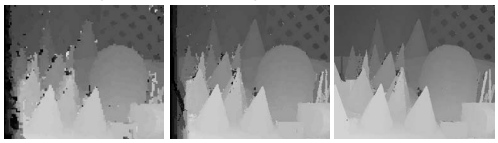

**Fig. 8**. Disparity maps in each level in a multiscale approach: level 3, 2, 1 (from left to right).

## 5. CONCLUSION

In this paper, we have proposed the cost aggregation and occlusion handling method for stereo matching with the weighted least square. By solving the iterative scheme with acceleration techniques such as the Gauss-Seidel method and multiscale approach, we efficiently estimated an accurate disparity map. The information at the visible pixels was propagated into the occluded pixels by sequential occlusion handling. The experimental results show that the performance of the proposed method is comparable to state-of-the-art methods in the Middlebury stereo datasets.

## 6. 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, Apr. 2002.

[2] T. Kanade and M. Okutomi, A stereo matching algorithm with an adaptive window: theory and experiment, *IEEE Trans. PAMI*, vol. 16, no. 9, pp. 920-932, Sep. 1994.

[3] S. Yoon, D. Min, and K. Sohn, "Fast dense stereo matching using adaptive window in hierarchical framework," *Proc. ISVC*, pp. 316-325, 2006.

[4] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," *Proc. IEEE CVPR*, pp. 858-863, 1997.

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

[6] A. Bobick and S. Intille, "Large occlusion stereo," *IJCV*, vol. 33, no. 3, pp. 1-20, Sep. 1999.

[7] J. Sun, Y. Li, S. Kang, and H. Shum, "Symmetric stereo matching for occlusion handling," *Proc. IEEE CVPR*, pp. 399-406, 2005.

[8] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," *Proc. IEEE ICPR*, pp. 15-18, 2006.

[9] L. Zitnick and S.B. Kang, "Stereo for image-based rendering using image oversegmentation," *IJCV*, vol. 75, no. 1, pp. 49-65, Oct. 2007.

[10] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *Proc. IEEE CVPR*, pp. 261-268, 2004.

[11] http://vision.middlebury.edu/stereo