Stereo Matching Using Cost Volume Watershed and Region Merging

Xiao Tan[a,b], Changming Sun[b], Xavier Sirault[c], Robert Furbank[c], Tuan D. Pham[d]

[a] *The University of New South Wales, Canberra, ACT 2600, Australia.*
[b] *CSIRO Computational Informatics, Locked Bag 17, North Ryde, NSW 1670, Australia.*
[c] *CSIRO Plant Industry, Clunies Ross Street, Canberra, ACT 2601, Australia.*
[d] *Aizu Research Cluster for Medical Engineering and Informatics, The University of Aizu, Fukushima 965-8580, Japan.*

Abstract

Segment based disparity estimation methods have been proposed in many different ways. Most of these studies are built upon the hypothesis that no large disparity jump exists within a segment. When this hypothesis does not hold true, it is difficult for these methods to estimate disparities correctly. Therefore, these methods work well only when the images are initially over segmented but do not work well for under segmented cases. To solve this problem, we present a new segment based stereo matching method which consists of two algorithms: a cost volume watershed algorithm (CVW) and a region merging (RM) algorithm. For incorrect under segmented regions where pixels on different objects are grouped into one segment, CVW algorithm regroups the pixels on different objects into different segments and provides disparity estimation to the pixels in different segments accordingly. For unreliable estimated regions, such as occluded regions, we merge them into neighboring reliable segments for robust disparity estimation. The comparison between our method and the current state-of-the-art methods shows that our method is very competitive and is robust particularly when the images are initially under segmented.

Keywords: 3D/stereo scene analysis, Stereo matching, Watershed segmentation, Region merging.

1. Introduction

Dense stereo matching is one of the key issues in computer vision. Many factors such as noise, distortion, reflection of light, lack of texture and occlusion will affect the matching process and make the matching results unreliable. A survey of studies on stereo matching and a broad range of stereo matching methods can be found in [1]. Stereo matching methods can be roughly categorized into two classes, local and global methods. Methods in the first class build on the hypothesis that neighboring pixels have similar disparities. These methods determine disparities by measuring the similarity between pixels in two local windows in stereo images. To perform well at boundaries, many techniques such as multiple windows [2], shiftable window [3], adaptive window [4], and weighted window [5] have been proposed. Global methods usually formulate the stereo matching problem using a Markov random field (MRF) model for integrating constraints such as the ordering constraint [6, 7, 8, 9], the uniqueness constraint [10], the visibility constraint [11], the piecewise smoothness constraint [3, 12, 13, 14], and the ground control point constraint [15, 16, 17].

In recent studies, most of the advanced methods employ the segmentation constraint [11, 18, 19, 20, 21, 22, 23, 24, 25]. The segment based methods preserve the boundary of disparity jumps and perform well in textureless regions where direct matching may suffer from the lack of intensity variation. The segment based methods typically have two main steps, image segmentation and disparity assignment [18, 26, 27, 23, 28]. In the first step, pixels are grouped into segments based on their color or texture. The disparities of pixels in these segments are then regularized according to some models such as a single disparity value [19, 28], a disparity plane [21, 22, 24, 29], or more sophisticated functions [25, 30].

In [18], a segment based method is described based on the hypothesis that no large disparity jump exists within an image segment. Under this hypothesis, the disparity can be calculated by warping the reference image to the source image. Thus, the stereo matching problem is solved by minimizing a global image warping energy. Li and Chen [21] proposed another segment based stereo matching method using graph cuts to assign the plane parameters by minimizing an energy function which considers both the observed data and the discontinuity between the segments. As the smoothness between segments is taken into account, this method can handle over segmentation; but it is still built upon the hypothesis that disparities in each

segment change smoothly. Sun *et al.* [11] and Yang *et al.* [24] used a plane to approximate each segment obtained from a color based segmentation method and then plane approximation is used as a soft constraint to control the labeling of pixel disparities. In [19] and [28], stereo matching is carried out on the segments obtained from over segmentation results. These two methods do not perform well for slant or curved surfaces because disparities of pixels in a segment are forced to be the same.

The common disadvantage of all segment based methods mentioned above is that they are all built upon the hypothesis that no large disparity jump exists within an image segment. In other words, the 3D surface boundaries are exactly aligned with the segment boundaries. Therefore, they all prefer an over segmentation input to prevent missing any boundaries of 3D surfaces. By exploiting a soft segmentation, the method described in [25] does not require 3D surface boundaries to coincide with segment boundaries. However, it also suffers from under segmentation, as the 3D models are calculated directly from the initial segmentations without using any method to identify the disparity jump within a segment. Unfortunately, not all segments can be well represented by a plane model (see Fig. 1) and not all 3D surface boundaries coincide with segment boundaries. Therefore, over segmentation can not be guaranteed without some prior knowledge about the image. Furthermore, as stated in [21], it is difficult to correctly estimate the 3D model for small segments which are typically obtained from an over segmentation. The computational cost will also be high if images are highly over segmented. Therefore, such over segmentation may not be a good choice. In this paper, we propose a new segment based stereo matching method. Comparing with the prior arts, the proposed method is very robust to different initial segmentations, especially to the under segmentation where it is usually difficult for competing methods.

2. Outline of the Proposed Method

A data cost volume which contains local similarity measurements between two pixels lying on the same scanline in the two stereo images is served as part of the input to our method. This data cost volume can be obtained from any local stereo matching algorithms. In this paper, we adopt the cross region based method as described in [31] with default parameters. An initial segmentation is required as the other input to our method. The initial segmentation can be obtained from any color or texture segmentation methods.

(a)

(b)

Figure 1: (a) Left image and the initial segmentation of the "Cones" dataset. Regions of two objects within the black contour are incorrectly grouped into one segment. (b) Left image and the initial segmentation of the "Reindeer" dataset. The segment within the white contour is a 3D curve on the cushion.

The planar model in the disparity space is used to represent the segments in this paper. We do not use any sophisticated model for the sake of simplicity and computational efficiency. Moreover, as will be demonstrated later, our method approximates image segments which correspond to 3D curved surfaces by multiple planes. Therefore, using a sophisticated model is generally not necessary. Given the initial segmentation, we detect under segmented regions and split them into subsegments using a cost volume watershed (CVW). After splitting, a plane is assigned to each subsegment. It is usually difficult to provide reliable plane estimation particularly when they are occluded. Therefore, we group these segments into neighboring corresponding segments whose parameters have been reliably estimated in the region merging (RM) step. Finally, we incorporate the segment based disparity estimation as a segment constraint with spital smoothness prior into the hierarchical belief propagation (BP) [32]. The framework of our method is shown in Fig. 2.

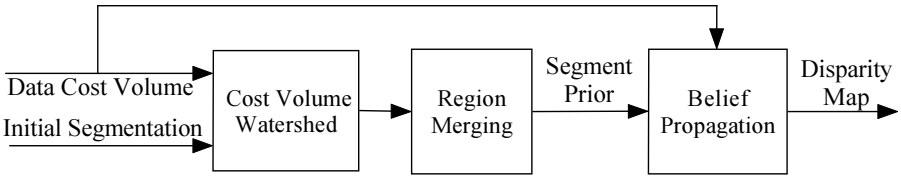


Figure 2: The framework of the proposed method.

3. Cost Volume Watershed and Segment Splitting

A disparity plane assigned to a segment is defined by parameters (a, b, c):

$$d = ax + by + c \tag{1}$$

where d is the disparity value, x and y are image coordinates of a pixel. The CVW algorithm splits a segment into two subsegments if multiple objects exist. Recursively performing the CVW, we will split a segment containing multiple objects into multiple subsegments. Then a plane will be assigned to each of these subsegments.

3.1. Seed Regions Extraction

A seed region is extracted from a two-step random sample consensus (RANSAC) algorithm [33] which consists of a possible plane (PP) finding

step and a seed region extraction step. The pixels are classified into occluded, reliable, and unreliable pixels. We do not use occluded or unreliable pixels in the plane finding step. The occluded pixels are detected by using cross checking. The reliable and unreliable pixels are identified using a confidence measurement [24]. To obtain a PP, we randomly select reliable pixels in the segments and calculate the plane parameters by using least squares fitting and then count the number of inliers. Inlier or outlier identification is carried out by checking the disparity difference between the disparity obtained from local matching and the estimated disparity given by the plane. The pixels whose disparity difference is smaller than the maximum rounding error, 0.5, are regarded as inliers. The plane with the maximum number of inliers is chosen as a PP. After a PP is chosen, we extract seed region by selecting pixels in the segment according to the chosen PP. In the extraction step, pixels in the segment are classified into inlier or outlier according to their disparity distance to this PP (inliers for the pixel whose distance difference is smaller than 0.5; outliers otherwise). Then, the maximum connected component of inlier pixels is selected as the seed region of the PP. The pixels declared as inliers are removed from the set of reliable pixels which are used to find another PP and the corresponding seed region. If no pixels are declared as outliers, the process will stop. If the segment contains multiple planes or objects, these outliers will typically be pixels on other planes or objects.

The process of seed region extraction is shown in Fig. 3. The position of pixels in Fig. 3 is illustrated in 1D for illustration purpose, while the idea is the same when pixels are in 2D image domain.

3.2. Cost Volume Watershed

After obtaining two PPs and their corresponding seed regions, the watershed algorithm [34, 35] is used to split a segment s into two subsegments. Our method is different from the studies in [34, 35] where the watershed is used on the grey level of a pixel. In our method, the watershed is carried out on the data costs of pixels at the disparity estimated from a PP. Let the data cost of pixel p at disparity d_{s_i} be $D_p\big(d_{s_i}\big)$, where d_{s_i} is the disparity estimated from the corresponding PP of s_1 or s_2. As d_{s_i} is a real number, we estimate $D_p\big(d_{s_i}\big)$ using a linear interpolation on the data cost values of two nearest integer numbers: $d_-\big(d_- \leq d_{s_i}\big)$ and $d_+\big(d_{s_i} \leq d_+\big)$:

$$D_p\big(d_{s_i}\big) = \big(d_+ - d_{s_i}\big)D_p\big(d_-\big) + \big(d_{s_i} - d_-\big)D_p\big(d_+\big) \tag{2}$$

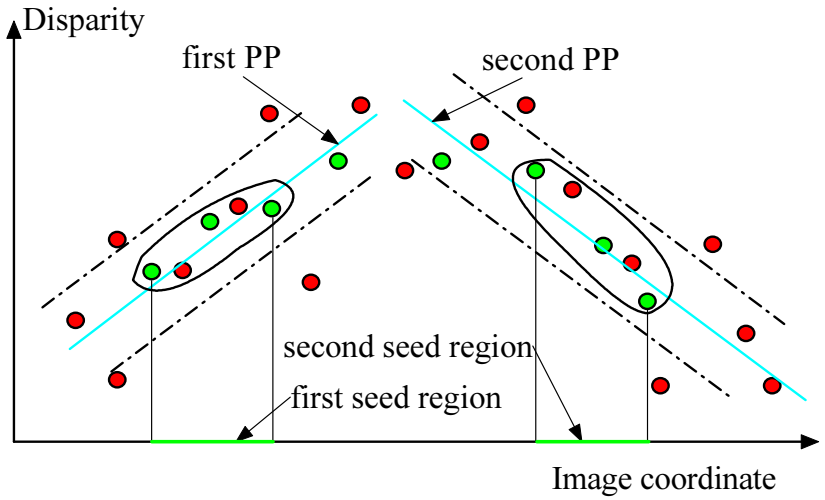


Figure 3: The illustration of seed region extraction. Red points are occluded or unreliable pixels. The disparities of pixels in the figure are obtained from local matching by using a winner-take-all(WTA) scheme. Two blue lines are two PPs. Dash lines on the two sides of the blue lines define the regions within which pixels are regarded as inliers. Two seed regions of the two PPs correspond to the green line segments on the image coordinates.

When a segment is from a curved surface in 3D, the CVW will split it into subsegments and use multiple planes to approximate the 3D curved surface. When using planes to approximate the curved surfaces, we expect the boundaries of these subsegments to be smooth. Therefore, in such a case, we place the boundaries between two subsegments on the projection of the intersection of their corresponding 3D planes (see Fig. 4). The projected intersection line of the two planes parameterized by $\varsigma_1(a_1, b_1, c_1)$ and $\varsigma_2(a_2, b_2, c_2)$ is given by:

$$(a_1 - a_2)\, x + (b_1 - b_2)\, y + (c_1 - c_2) = 0 \tag{3}$$

However, placing the boundary of subsegments on the projection is not always correct. For example, when two subsegments are on two parallel planes of different depths, there is even no intersection line. We notice that, when using multiple planes to approximate curved surfaces, the mass centers of two subsegments lie on different sides of the projected intersection line. Therefore, we set the boundaries of the two subsegments to be their projected intersection line only when two mass centers lie on different sides of the

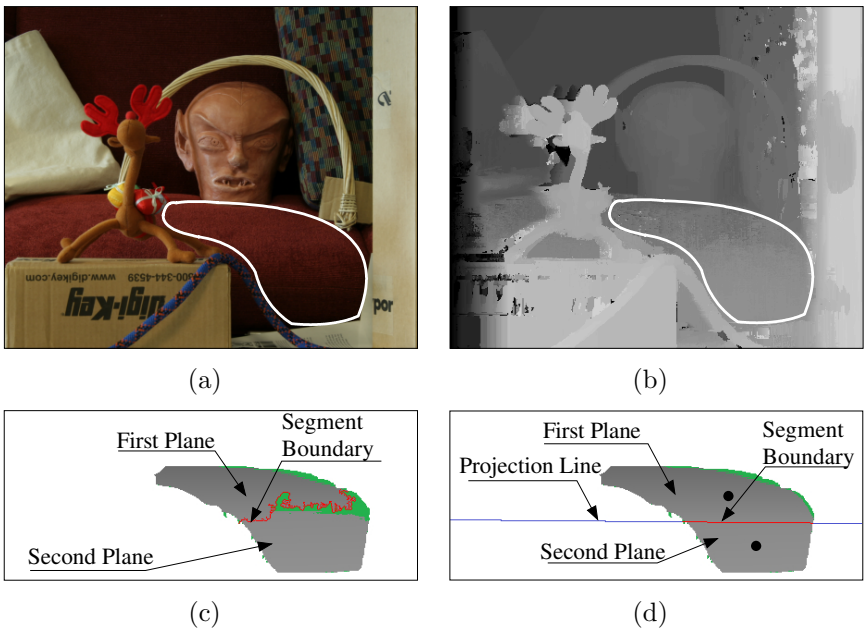
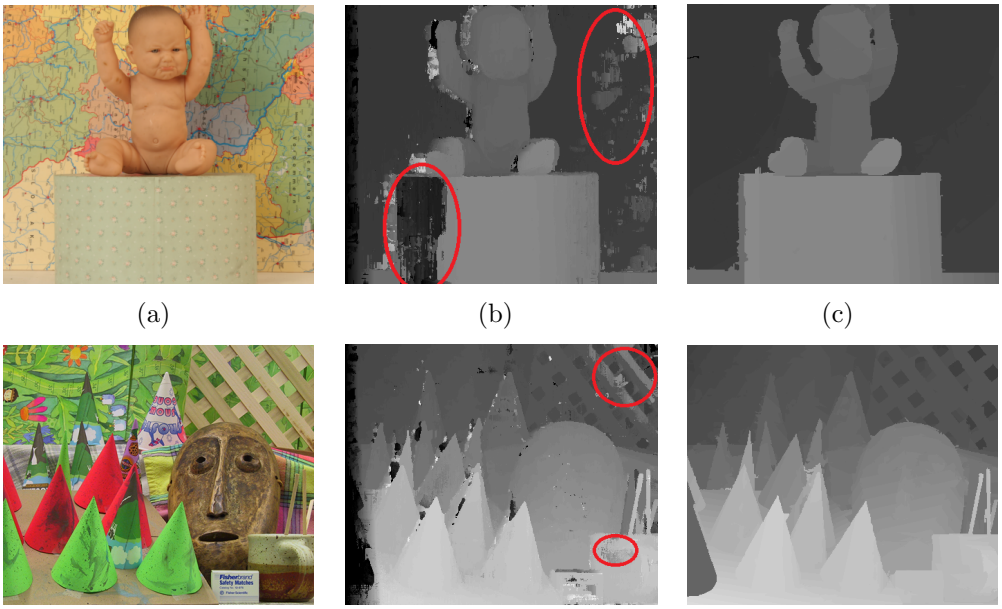


Figure 4: Illustration of approximating curved surface using planes. (a) Left image of the stereo image pair. A segment of curved surface is denoted. (b) Disparity results obtained from the data cost using the WTA scheme. (c) Results before placing subsegment boundary to the projection. (d) Results after placing subsegment boundary to the projected intersection line. The boundary between two subsegments are denoted in red. The blue line is the projected intersection line. The two dark points are the centers of masses of the two subsegments. Green pixels are the bad pixels.

projected intersection line.

3.3. Decision of Splitting

We have presented how to obtain PPs and how to split a segment into two subsegments in Sections 3.1 and 3.2. For segments where bad pixels dominate, especially those pixels in textureless regions (see Fig. 5), splitting them from its original segment is incorrect. To solve this problem, after splitting



(a) (b) (c)

(d) (e) (f)

Figure 5: (a)-(c) Left images of the "Baby" datasets, disparity estimation obtained from the data cost by using WTA (pixels with incorrect disparity are denoted in red ovals). (d)-(f): Results obtained by CVW.

the segment into two subsegments, we assign a plane to each subsegment by minimizing the following energy function.

$$E\left(\varsigma_1, \varsigma_2\right) = E_{\text{data}}\left(\varsigma_1, \varsigma_2\right) + E_{\text{smooth}}\left(\varsigma_1, \varsigma_2\right) + E_{\text{mdl}}\left(\varsigma_1, \varsigma_2\right) \tag{4}$$

where $\left(\varsigma_1, \varsigma_2\right)$ are planes assigned to subsegments s_1 and s_2 respectively. E_{data} is a data cost term:

$$E_{\text{data}}\left(\varsigma_1, \varsigma_2\right) = \sum_{p \in s_1} D_p\left(d_{s_1}\right) + \sum_{p \in s_2} D_p\left(d_{s_2}\right) \tag{5}$$

where d_{s_i} is the disparity obtained from the fitted plane for subsegment s_i. The second or the smoothness term penalizes disparity jump at subsegment boundaries:

$$E_{\text{smooth}}(\varsigma_1, \varsigma_2) = \sum_{p \in B(s_1, s_2)} |d_{s_1} - d_{s_2}| \tag{6}$$

where $B(s_1, s_2)$ is the boundary pixels between s_1 and s_2. The third term, the minimum description length (MDL) term [25], is used to prevent over fitting small regions:

$$E_{\text{mdl}}(\varsigma_1, \varsigma_2) = \begin{cases} \lambda_m & \varsigma_1 \neq \varsigma_2 \\ 0 & \varsigma_1 = \varsigma_2 \end{cases} \tag{7}$$

Let ς_1^p and ς_2^p be the corresponding PPs of the two subsegments s_1 and s_2. The plane assignment result has three possible combinations: $(\varsigma_1^p, \varsigma_1^p)$, $(\varsigma_2^p, \varsigma_2^p)$, and $(\varsigma_1^p, \varsigma_2^p)$. When the plane assignment of two subsegments are equal, i.e., $(\varsigma_1, \varsigma_2) = (\varsigma_1^p, \varsigma_1^p)$ or $(\varsigma_1, \varsigma_2) = (\varsigma_2^p, \varsigma_2^p)$, it means that segment s is not split and is fitted by one plane. If the plane assignment of two subsegments are equal, we will stop the CVW process on this segment; otherwise we continue carrying out the CVW on its subsegments until no subsegment is split.

4. Region Merging

After the CVW, we obtain a disparity estimation for each pixel from the fitted plane for this segment. We also have another disparity estimation obtained from the data cost. If these two disparity estimations are consistent, i.e., their difference is smaller than the rounding error 0.5, the pixel will be regarded as a support pixel to the plane estimation. The occluded pixels are not regarded as support pixels even though the two disparity estimations are consistent. If the ratio of the support pixels to all pixels in a segment is smaller than a threshold, the plane estimation to this segment will be regarded as unreliable and the segment will be merged into a neighboring segment. As large segments are more likely to represent independent objects and should not be grouped into other segments, the threshold T for deciding whether to merge a segment reduces with the size of the segment. We have explored the threshold T with respect to the segment size using a variety of functions: linear function, quadratic function, and logarithmic function and found that the best formulation is the logarithmic function with $T = \alpha - \beta \ln(S_s)$, where S_s is the number of pixels in s. Values of α and β are empirically set to $\alpha = 1.2$ and $\beta = 0.15$.

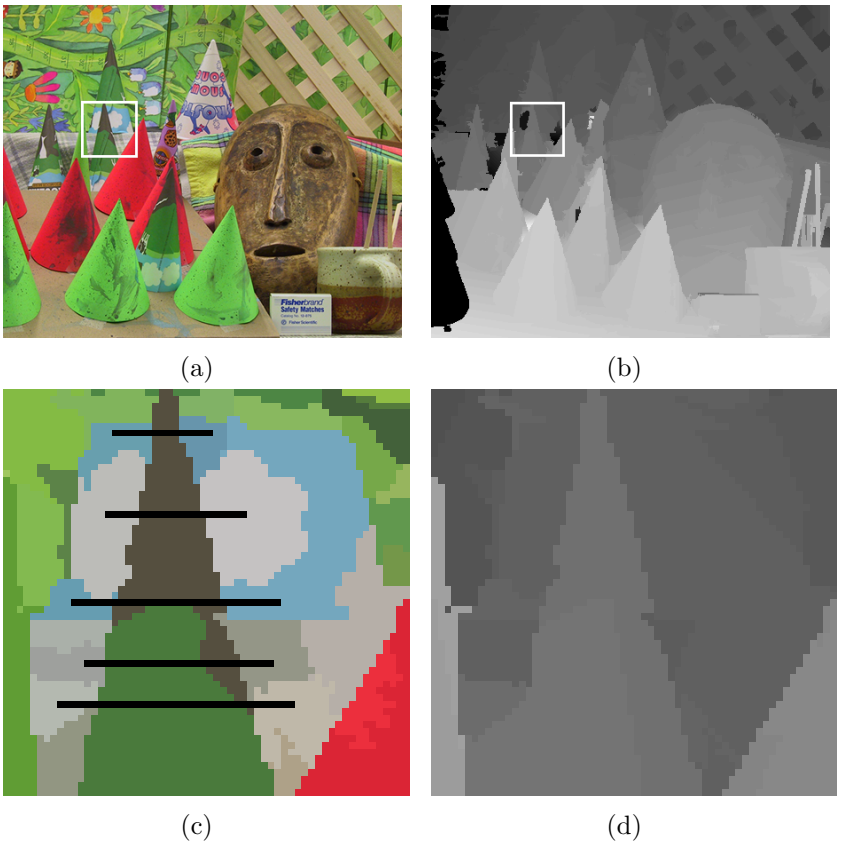


Figure 6: Illustration of merging occluded segments. (a) The left image of the "Cones" dataset (a cone in the white rectangle is broken into two parts by a cone in the foreground); (b) Result obtained from the CVW; (c) The occluded segments (the segments on the left hand side of the black lines) are merged into segments on the right hand side of the black lines; (d) Close-up view of the disparity map after region merging.

Merging a segment s into another segment s_m means assigning the plane parameter of s_m to s. Typically, s_m is a neighboring segment of s. But when s belongs to an object which is broken into several parts due to occlusion (see Fig. 6), merging s into any neighboring segment is not right. Therefore, we check the segments within the range of N_s pixels to the boundaries of s to see if these segments are connected with s in the 3D space using the method proposed in [36]. The segments which are connected with s in the 3D space are treated as candidates of s_m. Let $C(s)$ be the set of candidate of s_m for s. s_m is found by

$$s_m = \arg \min_{s' \in C(s)} \left(w_\mathrm{data}(s') + w_\mathrm{p}(s') + w_\mathrm{c}(s') \right) \tag{8}$$

where $w_\mathrm{data}(s')$, a data dependent cost of assigning the plane of s' to s, is defined as the averaged data cost of all unoccluded pixels of s calculated by using Eq. (2). If pixels in s are all occluded, we do not use the data term for s. The last two terms are the grouping costs by proximity based on the Gestalt grouping principle [5][37]: when a segment is close to or has a similar color to s, it is likely that the segment belongs to the same object with s. The grouping costs by proximity are defined by a spatial term $w_\mathrm{p}(s') = \frac{\Delta_\mathrm{p}}{\gamma_p}$ and a color term $w_\mathrm{c}(s') = \frac{\Delta_\mathrm{c}}{\gamma_c}$ respectively. γ_p and γ_c control the weights of the spatial and color terms. The distance between two segments, Δ_p, is defined as the minimum distance in the image space between them. The color difference, Δ_c, is defined as the Euclidean distance between the average color of the two segments in the RGB space. As an unreliable segment may be in the candidate set of another segment, $C(s)$, the minimum solution for Eq. (8) over all unreliable segments can not be found directly. Therefore we use a greedy search method to find an approximate solution: we record the plane assignment and the minimum cost of Eq. (8) for a segment in each iteration. In the next iteration, if the new assignment of s_m gives a smaller cost than the recorded minimum cost, we will take the new assignment and record this assignment and the new cost value. We check an unreliable segment once within each iteration. The iteration stops when plane assignments of all unreliable segments stay unchanged during an iteration.

4.1. Incorporating with Belief Propagation

After region merging, a segment based disparity estimation is obtained for each pixel. It is then used for regularizing the data cost as follows:

When p is not an occluded pixel:

$$\tilde{D}_p(d) = \begin{cases} D_p(d) & \text{if} \quad d = d_- \ \text{ or } \ d = d_+ \\ D_p(d) + P_d & \text{otherwise} \end{cases} \tag{9}$$

When p is an occluded pixel:

$$\tilde{D}_p(d) = \begin{cases} 0 & \text{if} \quad d = d_- \ \text{ or } \ d = d_+ \\ P_d & \text{otherwise} \end{cases} \tag{10}$$

where d_- ($d_- < d$) and d_+ ($d_+ > d$) are two nearest integer labels of d, the segment based disparity estimation. The term P_d penalizing the disagreement of d is set to 10 empirically. Then the hieratical BP is used on the regularized data cost $\tilde{D}$ with truncated linear function as the smoothness term penalizing the disparity jump between neighboring pixels: $\sum_{(p_1,p_2)\in\mathcal{N}} \min(|d_{p_1} - d_{p_2}|, T_s)$, where $\mathcal{N}$ is the set of 4 connected neighboring pixels and T_s is the truncation of smoothness term.

5. Experimental Results

The initial segmentation used in our method can be obtained from any color or texture based segmentation methods. In our experiments, the mean-shift algorithm [38] with different parameters is used to generate initial segmentations. Experiments are carried out with quantitative evaluation on the Middlebury datasets [39].

5.1. Parameter Selection

Our method involves four parameters: λ_m for preventing the CVW from over fitting small regions, γ_{p} and γ_{c} for controlling the grouping weights when merging regions, and N_{s} for defining the search range in the region merging step. Parameters λ_m, γ_{p} and γ_{c} are set emperically. According to our experiments, our result is fairly constant when λ_m changes from 1000 to 1400. The performance of CVW with respect to λ_m on the "Teddy" and "Cones" datasets is given in Fig. 7. We then study the performance of our algorithm with respect to parameters γ_{p}, γ_{c}, and N_{s}. Fig. 8 shows the performance of region merging according to γ_{p} and γ_{c} by keeping λ_m constant. Fig. 9 shows the performance of region merging according to N_{s} by keeping other parameters fixed. As shown in this figure, the accuracy improvement

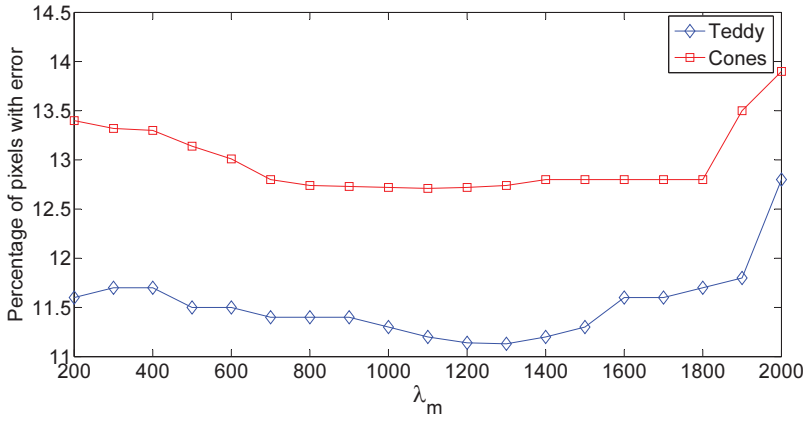


Figure 7: The performance of the CVW with respect to parameter λ_m on the "Teddy" and "Cones" datasets .

with the increase of N_{s} stops at some value. Because a segment is more likely to be merged into a neighboring segment rather than some segments far from it, setting a large value N_{s} will increase the computation cost. Based on the experiment, we set $\lambda_m = 1200$, $\gamma_{\mathrm{p}} = 80$, $\gamma_{\mathrm{c}} = 30$, and $N_{\mathrm{s}} = 0.25N_{\mathrm{d}}$ where N_{d} is the disparity range of the image.

5.2. Results Evaluation

We run our method on the Middlebury stereo dataset with constant parameters, i.e., $\lambda_m = 1200$, $\gamma_{\mathrm{p}} = 80$, $\gamma_{\mathrm{c}} = 30$, $N_{\mathrm{s}} = 0.25N_{\mathrm{d}}$, and $T_s = 5$. The running time on the Tsukuba dataset is 5 seconds on a standard PC with a 3.0 GHz CPU and 4GB RAM using Visual C++ 2008. The CVW takes about 10 seconds, the region merging process takes about 4 seconds; and the BP takes about another 4 seconds. The intermediate results and the final disparity maps are shown in Fig. 10. The quantitative evaluation is listed in Table 1 where the error threshold is one pixel of disparity. The results for the four datasets "Tsukuba", "Venus", "Teddy", and "Cones" are ranked based on the accuracy in the pixels in all regions (all), visable regions (vis), and depth discontinuous regions (disc). Our method is ranked at the 16th among all 148 submitted methods.

Experiments for testing the generality of our method are carried out on more stereo datasets. The results are shown in Fig. 11. The first four datasets are chosen from the Middlebury website [40]. The last two are a close-up view

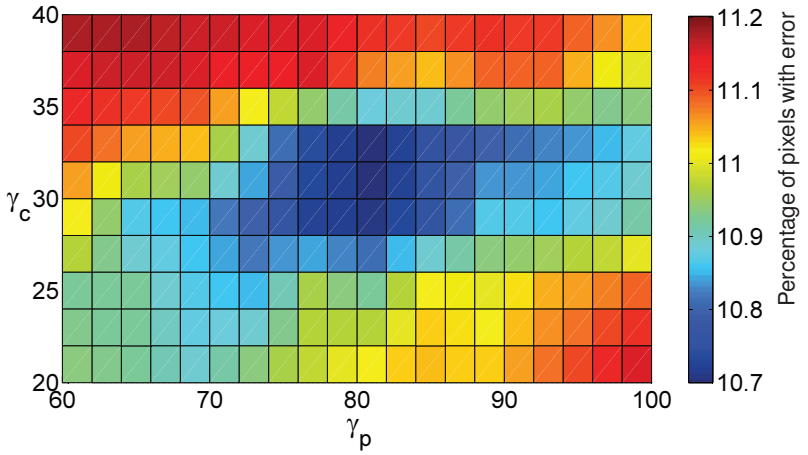


Figure 8: The performance of region merging according to γ_{p} and γ_{c} on the "Teddy" dataset.

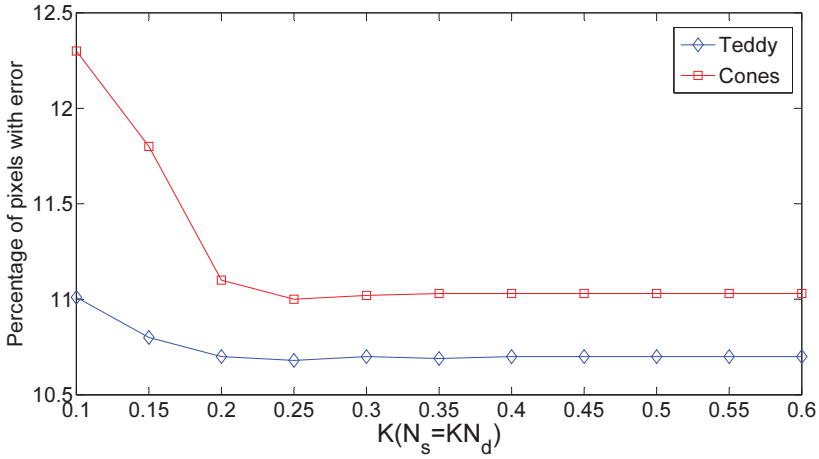


Figure 9: The performance of region merging according to N_{s} on the "Teddy" and "Cones" datasets.

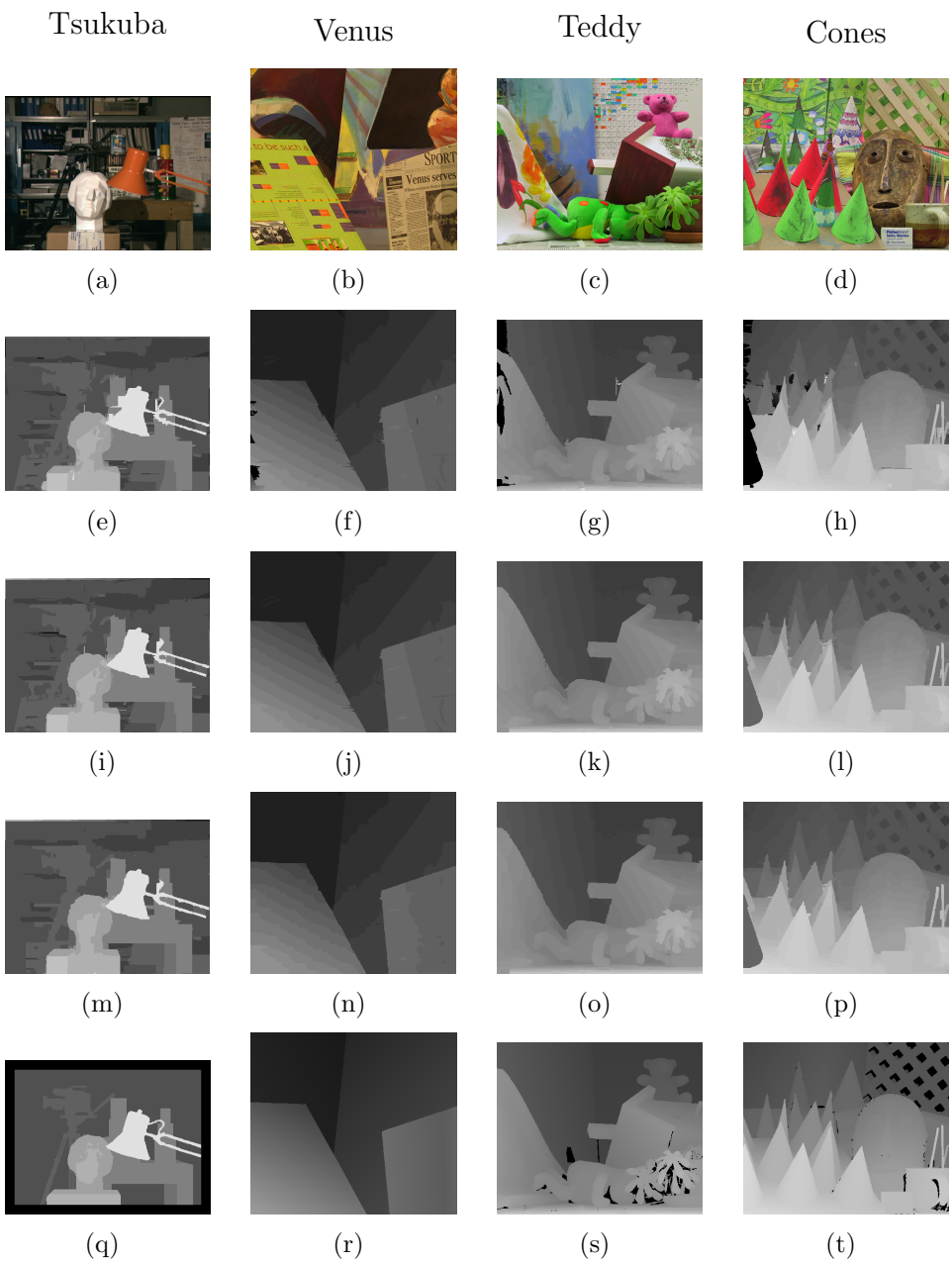


Figure 10: (a)-(d): Left images of the datasets; (e)-(h): Disparity maps after the CVW algorithm; (i)-(l): Disparity maps after the RM algorithm; (m)-(p): Disparity maps after BP optimization; (q)-(t): Ground truth of each dataset.

Table 1: The result of our method and other methods on the Middlebury stereo datasets.

Methods	Tsukuba			Venus			Teddy			Cones		
	vis	all	disc	vis	all	disc	vis	all	disc	vis	all	disc
ADCensus	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.9	2.42	7.25	6.95
SurfaceStereo	1.28	1.65	6.78	0.19	0.28	2.61	3.12	5.10	8.65	2.89	7.95	8.26
OurMethod	1.08	1.55	5.57	0.19	0.39	1.83	4.18	5.96	10.7	3.42	8.80	9.20
PlaneFitBP	0.97	1.83	5.26	0.17	0.51	1.71	6.65	12.1	14.7	4.17	10.7	10.6
OverSegmBP	1.69	1.97	8.47	0.50	0.68	4.69	6.74	11.9	15.8	3.19	8.81	8.89

of a rock and a bottom view of a tower. These experiments show that our method is applicable for various types of stereo images.

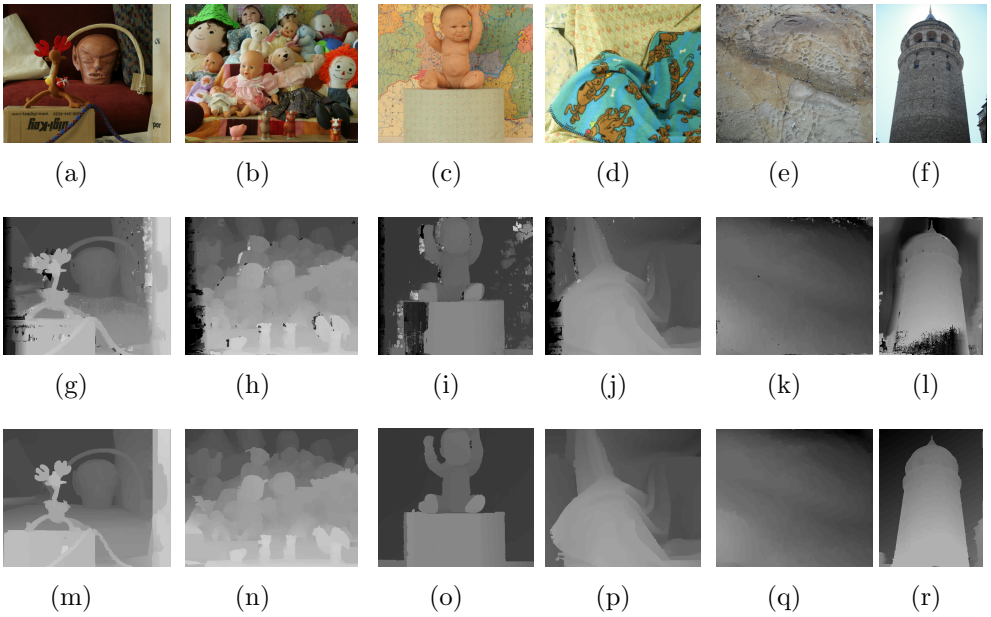


Figure 11: (a)-(f): Left images of the datasets; (g)-(l): Initial disparity maps obtained by applying WTA to the data costs; (m)-(r): Results obtained from our method.

6. Discussion

Our method is related to some other segmentation based methods where the scene is also represented by a collection of planes [21, 22, 11, 24]. The main advantage of our method over those methods is that our method employs CVW which splits under segmented regions into subregions; while other

methods enforce a single plane for representing these regions. This makes our method robust to under segmented regions where objects in different depths are usually grouped into these regions. Representing them by a single plane as used in those methods is prone to incorrectly fitting results. Some examples are given in Fig. 12 where different initial segmentations are used. Our CVW splits objects in different depths and fit them independently. Regularization parameter λ_m ranging from 0 to ∞ controls the splitting process. $\lambda_m = 0$ means that no penalty is imposed on fitting small segments. Conversely, setting $\lambda_m = \infty$ enforces all pixels in a segment being on a plane which is the model used in previous studies. The choice of λ_m is a tradeoff between splitting different objects in a segment at the expense of over fitting small regions. We found λ_m ranging between 1000 and 1400 to be a reasonable tradeoff.

In [21], planes are estimated via minimizing a global energy function which prevents the algorithm from incorrectly fitting the bad pixels in textureless regions. For the same purpose, our splitting process is carried out under the guidance of an energy function. The reason that our CVW does not over fit bad pixels in textureless regions while handles under segmented region is the following. In textureless regions, the term E_{data} is comparable for the obtained PPs, so the term E_{smooth} makes the single plane assignment favorable. In under segmented regions or regions with a curved surface, the term E_{data} dominates, so assigning different planes to different subsegments is favorable.

Our method handles under segmented regions which contain more than two objects or contain curved surfaces by recursively splitting segments into subsegments until all subsegments are well represented by planes. However, the proposed method has a common limitation as other plane model based methods–the plane model is not always suitable for reserving the curvature on the curve especially where the curve contains many bad pixels. For example, in Fig. 5(b) where the surface of the cylinder is approximated by two planes and most pixels on the left part of the cylinder are incorrectly matched. The fitting results would assign these pixels on to the plane which is close to them. This is because when failing to extract the correct plane for bad pixels, the smoothness term in Eq. (4) favors to assign these bad pixels on to a neighboring plane.

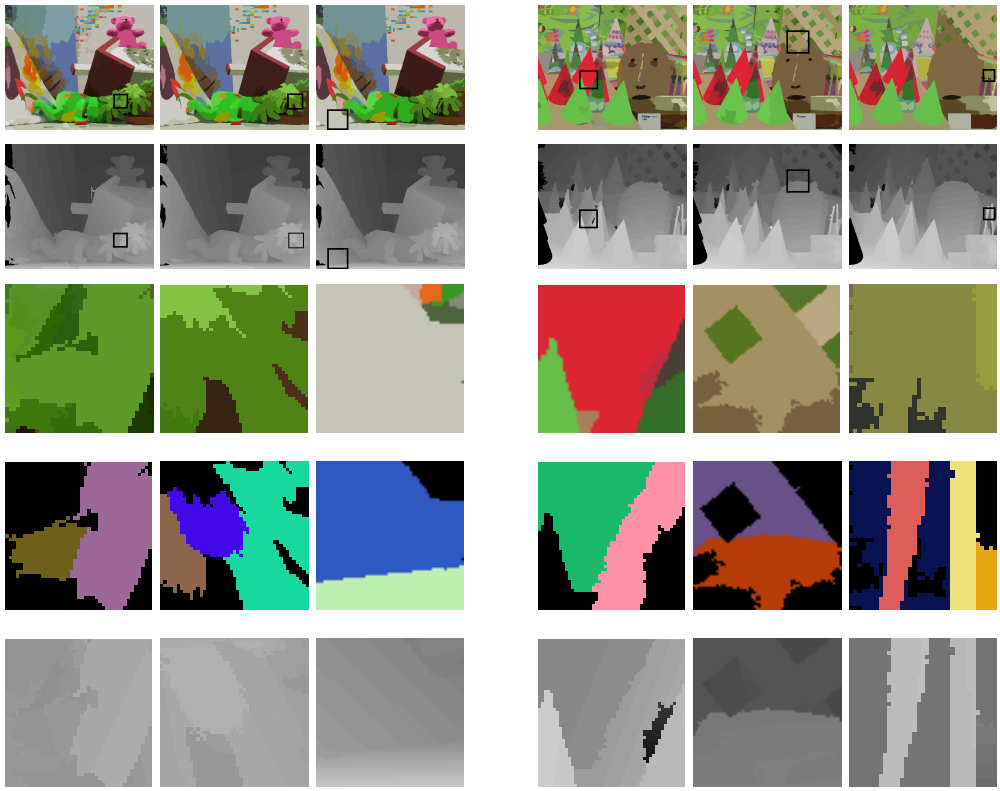


Figure 12: Results of the CVW on the "Teddy" and "Cones" datasets on different initial segmentation results; First row: initial segmentation results; Second row: disparity maps obtained from the CVW; Third row: close-up view of a part of initial segmentation results as indicated by the small square shown in the first and second rows; Fourth row: subsegments obtained from the CVW; Last row: close-up view of the disparity estimation obtained from the CVW.

7. Conclusion

In this paper, we propose a new dense stereo matching method which is built on the CVW and the RM algorithms. The proposed CVW algorithm can differentiate pixels on different segments from under segmentation which usually contain multiple objects and can also use multiple planes to approximate segments with curved surfaces. The proposed RM algorithm makes an accurate disparity estimation for pixels in occluded or unreliable regions by merging such regions into their neighboring segments. The experimental results show the effectiveness of our method. The comparison between our method and the current state-of-the-art algorithms has been carried out and good results are obtained.

Acknowledgments

We thank Chao Zhang and Ran Su of the University of New South Wales for their comments. Tan is partially supported by the China Scholarship Council. Sun is partially supported by the CSIRO's Transformational Biology Capability Platform. A preliminary conference version of this paper appeared in [41].

References

[1] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International Journal of Computer Vision 47 (1) (2002) 7–42.

[2] A. Fusiello, V. Roberto, E. Trucco, Efficient stereo with multiple windowing, in: IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 858–863.

[3] M. Okutomi, Y. Katayama, S. Oka, A simple stereo algorithm to recover precise object boundaries and smooth surfaces, International Journal of Computer Vision 47 (1) (2002) 261–273.

[4] T. Kanade, M. Okutomi, A stereo matching algorithm with an adaptive window: Theory and experiment, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (9) (1994) 920–932.

[5] K. J. Yoon, I. S. Kweon, Adaptive support-weight approach for correspondence search, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 650–656.

[6] P. N. Belhumeur, A Bayesian approach to binocular steropsis, International Journal of Computer Vision 19 (3) (1996) 237–260.

[7] A. F. Bobick, S. S. Intille, Large occlusion stereo, International Journal of Computer Vision 33 (3) (1999) 181–200.

[8] I. J. Cox, A maximum likelihood N-camera stereo algorithm, in: IEEE Conference on Computer Vision and Pattern Recognition, 1994, pp. 733–739.

[9] H. Ishikawa, D. Geiger, Occlusions, discontinuities, and epipolar lines in stereo, in: European Conference on Computer Vision, 1998, pp. 232–248.

[10] D. Marr, T. Poggio, Cooperative computation of stereo disparity, Science 194 (4262) (1976) 283–287.

[11] J. Sun, Y. Li, S. B. Kang, H. Y. Shum, Symmetric stereo matching for occlusion handling, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, 2005, pp. 399–407.

[12] J. Sun, N. N. Zheng, H. Y. Shum, Stereo matching using belief propagation, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (7) (2003) 787–800.

[13] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (11) (2001) 1222–1239.

[14] O. Woodford, P. Torr, I. Reid, A. Fitzgibbon, Global stereo reconstruction under second-order smoothness priors, IEEE Transactions on Pattern Analysis and Machine Intelligence (2009) 2115–2128.

[15] J. C. Kim, K. M. Lee, B. T. Choi, S. U. Lee, A dense stereo matching using two-pass dynamic programming with generalized ground control points, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, 2005, pp. 1075–1082.

[16] L. Wang, R. Yang, Global stereo matching leveraged by sparse ground control points, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3033–3040.

[17] A. Geiger, M. Roser, R. Urtasun, Efficient large-scale stereo matching, Asian Conference on Computer Vision (2011) 25–38.

[18] H. Tao, H. S. Sawhney, R. Kumar, A global matching framework for stereo computation, in: International Conference on Computer Vision, Vol. 1, 2001, pp. 532–539.

[19] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, High-quality video view interpolation using a layered representation, ACM Transactions on Graphics 23 (3) (2004) 600–608.

[20] Y. Taguchi, B. Wilburn, C. L. Zitnick, Stereo reconstruction with mixed pixels using adaptive over-segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[21] L. Hong, G. Chen, Segment-based stereo matching using graph cuts, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, 2004, pp. 74–81.

[22] Z. F. Wang, Z. G. Zheng, A region based stereo matching algorithm using cooperative optimization, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[23] A. Klaus, M. Sormann, K. Karner, Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure, in: International Conference on Pattern Recognition, Vol. 3, 2006, pp. 15–18.

[24] Q. Yang, L. Wang, R. Yang, H. Stewénius, D. Nistér, Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (3) (2008) 492–504.

[25] M. Bleyer, C. Rother, P. Kohli, Surface stereo with soft segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 1570–1577.

[26] J. Y. A. Wang, E. H. Adelson, Representing moving images with layers, IEEE Transactions on Image Processing 3 (5) (1994) 625–638.

[27] S. Baker, R. Szeliski, P. Anandan, A layered approach to stereo reconstruction, in: IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 434–441.

[28] C. L. Zitnick, S. B. Kang, Stereo for image-based rendering using image over-segmentation, International Journal of Computer Vision 75 (1) (2007) 49–65.

[29] M. Bleyer, M. Gelautz, A layered stereo algorithm using image segmentation and global visibility constraints, in: International Conference on Image Processing, Vol. 5, 2004, pp. 2997–3000.

[30] M. H. Lin, C. Tomasi, Surfaces with occlusions from layered stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (8) (2004) 1073–1078.

[31] K. Zhang, J. Lu, G. Lafruit, Cross-based local stereo matching using orthogonal integral images, IEEE Transactions on Circuits and Systems for Video Technolog 19 (7) (2009) 1073–1079.

[32] P. F. Felzenszwalb, D. P. Huttenlocher, Efficient belief propagation for early vision, International Journal of Computer Vision 70 (1) (2006) 41–54.

[33] M. A. Fischler, R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM 24 (6) (1981) 381–395.

[34] S. Beucher, F. Meyer, The morphological approach to segmentation: the watershed transformation, Mathematical Morphology in Image Processing (1992) 433–481.

[35] L. Vincent, P. Soille, Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6) (1991) 583–598.

[36] M. Bleyer, C. Rother, P. Kohli, D. Scharstein, S. Sinha, Object stereo - joint stereo matching and object segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3081–3088.

[37] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.

[38] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 603–619.

[39] D. Scharstein, R. Szeliski, http://www.vision.middlebury.edu/stereo/ (2011).

[40] H. Hirschmuller, D. Scharstein, Evaluation of cost functions for stereo matching, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.

[41] X. Tan, C. Sun, X. Sirault, R. Furbank, T. D. Pham, Tree structural watershed for stereo matching, in: Image and Vision Computing New Zealand, 2012, pp. 340–345.