A Symmetric Patch-Based Correspondence Model for Occlusion Handling

Yi Deng[†] dengyi00@mails.tsinghua.edu.cn
Qiong Yang[‡] qyang@microsoft.com
Xueyin Lin [†] lxy-dcs@mail.tsinghua.edu.cn
Xiaoou Tang [‡] xitang@microsoft.com

†Key Lab of Pervasive Computing(MOE)
Dept. of Computer Science, Tsinghua University,
Beijing 100084, P.R. China

‡Microsoft Research Asia
Beijing Sigma Center, Zhichun Road, Hai Dian District,
Beijing 100080, P.R. China

Abstract

Occlusion is one of the challenging problems in stereo. In this paper, we solve the problem in a segment-based style. Both images are segmented, and we propose a novel patch-based stereo algorithm that cuts the segments of one image using the segments of the other, and handles occlusion areas in a proper way. A symmetric graph-cuts optimization framework is used to find correspondence and occlusions simultaneously. The experimental results show superior performance of the proposed algorithm, especially on occlusions, untextured areas and discontinuities.

1. Introduction

Occlusion is one of the major challenges in stereo. For a two-frame stereo system, a point in an image is *occluded* if its corresponding point is invisible in the other image. Computing of occlusion is ambiguous, so prior constraints need to be imposed. Ordering and uniqueness are two constraints typically used. Ordering constraint inhibits the change of order between images. This constraint is often exploited in a dynamic program framework [6], because it can greatly reduce the search space and lead to an efficient matching. But it is incorrect when there are thin objects in the scene [6]. Uniqueness is effective constraint, which enforces at most one correspondence for each element of both images. Zinick and Kanade imposed it as the inhibitions in a cooperative frame [17]. Kolmogorov and Zabih [8] proposed an graph-cuts based algorithm that searches a global optimal *unique* configuration of assignments between pixels of images. Jian *et al*[13] modified the uniqueness constraint to a weaker constraint, visibility constraint, so that the problem of uniqueness caused by sampling [11] can be avoided when the scene contains horizontally slanted planes. More early techniques for occlusion handling can be found in the survey by Egnal and Wildes [6] and that by Brown *et al*[4].

Our new idea of handling occlusion originates from the observation that the occlusion border in one image corresponds to a discontinuity in the other images [6], and the discontinuity often makes strong texture on the other image which can be achieved by color segmentation. Therefore, we provide a framework that can use the segmentation of one image to help compute the occlusion in the other.

Color segmentation information is used in several recent stereo approaches [7, 16, 15, 14, 2]. They are called segment-based methods. They use the assumption that discontinuity only happens at the boundaries of segmented regions (we call it *discontinuity assumption* in this paper). The performance of discontinuity and untextured area is improved if the scene contains generally slanted plane. But occlusion is not explicitly moded in Hong and cheng's [7] and Zhang and Kambhamettu's [16] algorithms. So it is hard to identify occlusions and they use a robust error criteria before global matching or region growing. Bleyer and Gelautz's [2] formulate the problem in a two-level framework. Uniqueness is used in the pixel level to infer occlusion, and discontinuity assumption is imposed in segment-level. However, their method only use segmentation information of one image while the boundary information of occlusion is actually in the other image from our analysis above.

In this paper, we use segmentation of both images, and propose a patch-based framework to explicitly handle occlusions. Based on the observation that the observation that the shared edge of a visible area and an occluded area corresponds to a discontinuity in the other image, we introduce the concept of *patch*. We first segment both images, and warp the segment of one image to the other by disparity. Then the warped segment is divided into several small patches by the segment boundaries in the other image. We constrain the boundary of occlusions to be the boundary of patches. A symmetric global framework using graph cuts is constructed to find the disparity and occlusions embodied by the patch segmentation. The new correspondence approach gives a proper constraint for occlusions, which leads to better results near occlusions, and inherits the advantage of segment-based approaches on slanted plane, as well as untextured and discontinuity areas.

The main contributions in this paper include: 1) By using segmentation of both images, a novel matching unit *patch* is introduced, which preserves the visibility consistency and lead to a proper handling of occlusions; 2) A symmetric graph-cuts framework on patches is proposed to find disparities and occlusions simultaneously.

2. Problem formulation

2.1. Stereo problem

Let L and R be the set of pixels in the left and right images respectively, and let $P = L \cup R$. The pixel in the left image will have coordinate (p_x, p_y), and the pixel in the right image will have coordinate (p'_x, p'_y). The stereo problem can be formulated as a labeling problem, in which each pixel $p \in P$ must be assigned a label f_p within some label set $\mathcal{L}$. That is, the goal is to find a labeling configuration f that assigns each pixel $p \in P$ a label $f_p \in \mathcal{L}$.

To describe the generally slanted plane, we use a 3-parameter linear transform in this paper, and take the parameters of the linear transform as the definition of labels, i.e.

$$f_p = f_{p'} = \langle c_1, c_2, c_3 \rangle \Leftrightarrow$$
$$p \overset{\langle c_1,c_2,c_3 \rangle}{\leftrightarrow} p', \text{where } p'_x = c_1 p_x + c_2 p_y + c_3, p'_y = p_y$$

where $p \overset{\langle c_1,c_2,c_3 \rangle}{\leftrightarrow} p'$ means p and p' are corresponding points if assigned a label $\langle c_1, c_2, c_3 \rangle$ to either of them. If a point is occluded in the other image, its label is ϕ.

2.2. Patch and visibility consistency

In order to find the labels for all the points that are mostly accordant to the input stereo image pair, prior assumptions are generally used, such as smoothness assumption and uniqueness constraints. In segment-based algorithms, discontinuity assumption is used. However, the border of the segments in one image is not always the border of occlusion in that image, and the shared edge of a visible area and an occluded area is corresponding to a discontinuity of the other image. Therefore, we firstly separate the segment of one image into patches by using the discontinuity of the other, and impose a new constraint, which enforce the same visibility for all the pixels within a *patch*. In the following paragraphs, we will give definition of the patch and show why this constraint is reasonable.

Suppose that a segment r is a region in the left image, and its label is denoted as f_r. If $f_r = \phi$, r is fully occluded, we consider it as a whole. Otherwise, we warp all the points in r into the right image by it. The warped segment r' may cross a number of segments in the right image, e.g. two segments s' and t' in the right image in Figure 1. Suppose that the shared edge between s' and t' is e', there should be a shared edge e correspondent to e' in r. As a result, the points in r are separated into two sets, q_1 and q_2 by e. We call them the *patches* of the region r. For a clear description, we define $q_r^{f_r}(i)$ as the i-th patch of segment under label f_r or simply $q_r(i)$. By assuming the boundary of segment in the right image to be the potential discontinuity, the corresponding shared edge in the left image is the potential occlusion boundary. So we enforce the same visibility for all the points within a patch, and call it the *patch-consistency* constraints. Different patches can have different visibilities, but one patch cannot partly occluded. In this way, we use the segmentation information in one image to give a hard constraint to occlusion in the other image. The partial visibility within a segment is allowed and guide by segmentation information, which is advantageous over previous segment-based approaches. Experimental results will also show its advantage in later sections.

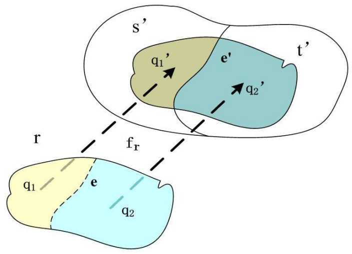


Figure 1. Definition of patches. Region r in the left image are warped to the right image and the warped image is separated by shared edge e' of s' and t' into q'_1 and q'_2. Accordingly r is separated into two patches q_1 and q_2.

The definition of patch is symmetric, i.e. the patches in the right image can be similarly defined. For example in Figure 1, q'_1 and q'_2 in the right image are patches of segment s' and t' respectively if they are assigned with the same label with r. In this situation, we call $q_1 \sim q'_1$ (and $q_2 \sim q'_2$) a patch pair, because if one of them is visible, the other should be visible too. If $f_{t'} \neq f_r$, for each visible patch in t', a corresponding patch within one segment of the left image with the label $f_{t'}$ can be found. So all the visible patches in the left and right images are paired.

Using the *patch-consistency constraint*, the label configuration can be reformulated in a *segment-patch* level. That is, for each segment r in either image, we assign it a label $f_r \in \mathcal{L}$, and if $f_r \neq \phi$, we assigned equal number of visibilities $v_r(i)$ (also denoted as $v_r(q_r(i))$ for each patch of r. The i-th patch of r is visible if $v_r(i) = 1$ and otherwise occluded. Besides, we constrain the configuration to be *regular*, i.e. the visible patches in the configuration should be in pair. For example in Figure 1, if $f_r \neq \phi$ and $v_r(q_1) = 1$, we constrains that $f'_s = f_r$ and $v_{s'}(q'_1) = 1$.

The label of each point can be computed as

$$f_p = \begin{cases} f_r & f_r \neq \phi \wedge v_r(i) = 1 \\ \phi & otherwise \end{cases}, \forall p \in q_r(i).$$

By abuse of notations, we use f to denote the configuration in a *segment-patch* level in the rest of the paper.

2.3. Energy function

We compute the optimal configuration under an energy minimization framework:

$$\begin{aligned} f_{opt} &= \arg\min\nolimits_f E(f) \\ &= \arg\min\nolimits_f E_{data}(f) + E_{smooth}(f) + E_{occl}(f). \end{aligned}$$

$E_{data}(f)$ is the energy of matching errors for each visible patches. It is defined as

$$E_{data}(f) = \sum_r T(f_r \neq \phi) \sum_i \epsilon_{patch}(q_i, f_r),$$

$$\epsilon_{patch}(q_i, f_r) = \sum_{p \in q_i} (\epsilon_{point}(p, p'), p \overset{f_r}{\leftrightarrow} p')$$

where $T(\cdot)$ equals 1 if the argument holds and otherwise 0, and $\epsilon_{point}(p, p')$ is the intensity difference between point p in the one image and point p' in the other image.

$E_{smooth}(f)$ exploits smoothness assumptions. If two connected patch with same label contains different visibility, we impose a penalty. The selection of this smoothness term affects whether the energy can be minimized efficiently by graph-cuts, so we put its definition in next section.

$E_{occl}(f)$ gives penalties to occluded pixels (otherwise a trivial configuration with all pixels occluded will take the least energy). It is defined as:

$$E_{occl}(f) = C_o \sum_r E_{occl}(r),$$

$$E_{occl}(r) = \begin{cases} S_a(r) & f_r = \phi \\ \sum_i (1 - v_r(i)) & otherwise \end{cases}$$

where C_o is a occlusion constant controlling the weight of occlusion energy in the whole energy.

3. Energy minimization

The patches is generated by warping the segment according to its label, but we do not know the label of a segment before matching. So a global framework is proposed to compute labels of segments and the visibility of each patch simultaneously.

3.1. Alpha expansion framework

We know that a segment can have $|\mathcal{L}|$ possible labels and the separation of the segment into patches is generally different under each label. So the whole searching space is huge, so it is impractical to directly search the optimal result. We use the α expansion framework proposed by Boykov *et al*[3] to solve the problem. By using it, the problem is solved in an iterative style, and a strong local minimum is obtained in each iteration. After convergence, the global minimum is achieved.

In our situation, we start from a configuration with all segments occluded. Then in each iteration, a label α is chosen, and a local minimum within one α expansion is computed using graph-cuts. If no label can further decrease the energy, we get the final minimized configuration. If a configuration is within an α expansion of f, a segment can only have one of the following 3 choices: keeping its current label in f, becoming occluded, or changing its label to α, and the configuration should keep regular.

3.2. Binary-variable energy term

Now we convert the minimization of $E(f)$ in each iteration (α expansion move) into a minimization of a binary-variable energy, so that the latter minimization can be performed by graph-cuts.

We classify the the segments into two classes according its labels before expansion:

1. For each segment r in either image, $f_r \notin \{\phi, \alpha\}$, we allocate a *labeling variable* l_r to decide the label of r after expansion, denoted as $\tilde{f}_r$. The relation between l_r and $\tilde{f}_r$ is

   $$\tilde{f}_r = \begin{cases} f_r/\phi & l_r = 0 \\ \alpha/\phi & l_r = 1 \end{cases}.$$

   Whether f_r equals ϕ is determined by the visibility of the patches. Suppose the number of r's patches under label f_r and α are N_r^0 and N_r^α respectively. The visibility of the patches are determined as follow:

   (a) If r is in the left image, we allocate N_r^0 *visibility variables*, $b_r^0(i)$, indicating visibilities of patches under the label f_r when $l_r = 0$, and define $\tilde{v}_r(i) = 1 - b_r^0(i)$. we also allocate N_r^α *visibility* variables, $b_r^\alpha(i)$, indicating visibilities of patches under the label α when $l_r = 1$, and define $\tilde{v}_r(i) = b_r^\alpha(i)$.

   (b) If r is in the right image, we need not allocate new binary variables, and choose the correct binary variables allocated for segments in the left image to indicate the visibility of the patches. We use the same notation for those chosen visibility variables.

2. For each segment $r \subset P, f_r \in \{\phi, \alpha\}$, labeling variable is not necessary, only visibility variables $b_r^\alpha(i)$ are allocated.

The set of all binary variables is denoted as $V = \{l_r, b_r^0(i), b_r^\alpha(i)\}$.

There are some other constraints for values of V. If $l_r \neq 0$, we require $\forall b_r^0(i) \neq 0$, and if $l_r \neq 1$, $\forall b_r^\alpha \neq 1$. If this requirement is satisfied, we say V is *regular*, and otherwise *irregular*. When V is regular, we denote the corresponding configuration as $f(V)$.

The α expansion move can be performed by minimizing the following energy function of binary variables:

$$\tilde{f} = \arg\min\nolimits_V E^b(V),$$

$$E^b(V) = \begin{cases} E\left(f(V)\right) & V \text{ is regular} \\ \infty & \text{otherwise} \end{cases}$$

$E^b(V)$ can be rewritten as the sum of the following terms:

$$\begin{aligned} E^b(V) &= E^b_{reg}(V) \\ &\quad + E^b_{data}(V) + E^b_{smooth}(V) + E^b_{occl}(V) \end{aligned}$$

$E_{reg}(V)$ takes infinity value if V is not regular, and 0 otherwise. It can be written from the definition of regular V:

$$E_{reg}(V) = \sum_r \sum_i E^0_{reg}(l_r, b_r^0(i)) + \sum_i E^\alpha_{reg}(l_r, b_r^\alpha(i)),$$

$$E^0_{reg}(l_r, b_r^0(i)) = \begin{cases} \infty & l_r = 1 \wedge b_r^0(i) = 0 \\ 0 & \text{otherwise} \end{cases},$$

$$E^\alpha_{reg}(l_r, b_r^\alpha(i)) = \begin{cases} \infty & l_r = 0 \wedge b_r^\alpha(i) = 1 \\ 0 & \text{otherwise} \end{cases}$$

E^b_{data} and E^b_{occl} can be trivially derived from the definition of E_{data} and E_{occl}. We know give the definition for E^b_{smooth} and equivalent E_{smooth}. Our visibility variables are assignment-like variables as in the approach by Kolmogorov and Zabih [8]. So we take the similar smoothness energy function as:

$$E^b_{smooth}(V) = C_s \sum_q \sum_{q_n \in \mathcal{N}_q} S_c(q, q_n) \cdot T(b_q = b_{q_n})$$

where $\mathcal{N}_q$ is the set of neighboring patches of q with the same label as q, $S_c(q, q_n)$ is the length of shared border of q and q_n, b_q is the visibility variable corresponding to patch q and C_s is a smoothness constant controlling the balance of smoothness with other energy. The equivalent E_{smooth} is

$$E_{smooth} = \sum_q \sum_{q_n \in \mathcal{N}_q^0} E_{smooth}(q, q_n),$$

$$E_{smooth}(q, q_n) = S_c(q, q_n) \begin{cases} 0 & f_q = f_{q_n} \\ C_s & f_q \neq \phi \vee f_{q_n} \neq \phi \\ 2C_s & \text{otherwise} \end{cases}$$

where $\mathcal{N}_q^0$ is the set of neighboring patches of q.

3.3. Regularity of Energy Function

$E^b(V)$ can be trivially rewritten as the sum of energy items up to 2 variables at a time, i.e:

$$E^b(V) = \sum_i E^i(v_i) + \sum_{i<j} E^{i,j}(v_i, v_j)$$

And for all $E^{i,j}(v_i, v_j)$,

$$E^{i,j}(0,0) = E^{i,j}(1,1) = 0, E^{i,j}(0,1), E^{i,j}(1,0) \geq 0$$

So it obeys the regularity inequality introduced by Kolmogorov and Zabih [9]. We then use the results of [9] to compute the minimization of $E^b(V)$.

4. Algorithm implementation

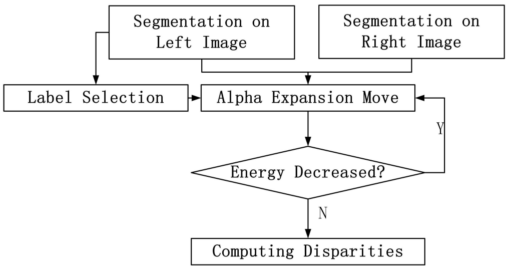
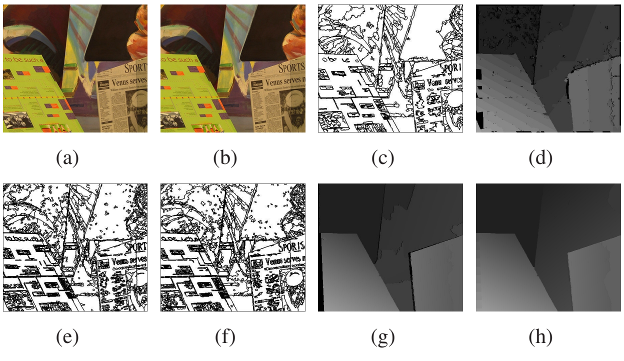


Figure 2. Flowchart of our algorithm.

The flowchart of the whole algorithm is shown in Figure 2. The left image (Figure 3.a) is firstly segmented into relatively large segments using mean-shift segmentation algorithm [5] (Figure 3.c). A Sum-of-Absolute-Difference(SAD) algorithm with Birthfield and Tomas's dissimilarity algorithm [1] plus cross-checking algorithm is used to find disparities of reliable points. A plane fitting similar to [7] is exploited to select the label set $\mathcal{L}$ (Figure 3.d). Input images are then over-segmented into smaller segments (Figure 3.e and Figure 3.f). The symmetric algorithm proposed in Section 3 is exploited to compute labels of each segment and visibilities of each patch. Disparities and occlusion results are then obtained (Figure 3.g). An occlusion filling operation is used to compute the disparity map without occlusions (Figure 3.h).


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3. Intermediate results for "venus". (a) and (b) are input left and right images. (c) is segmentation result for disparity selection. (d) is the result of label selection. (e) and (f) are segmentation results for matching. (g) is the result with occlusions (marked in black), and (h) is the result after occlusion filling.

4.1. Parameter selection

There are two parameters in our algorithm the smoothness constant C_s and occlusion constant C_o. We find that C_s is somehow sensitive to input images and propose a method to select the value automatically so that our algorithm can be more adaptive.

Our selecting strategy is designed according to the following analysis. The data error energy makes the correctly matched patch pair, which contains least SAD error in noise free situation, to be selected. Noise may cause a wrong patch to have smaller errors than the correct one, but the wrong patch is often inconsistent with neighbors. Smoothness energy is used to punish the inconsistency and reject the wrong match. Therefor we choose a larger constant for greater noise level.

The noise level is estimated using the disparity map of reliable points in the label selection step. For each reliable point, we compute a matching error ϵ, and take the average of all matching errors $\bar{\epsilon}$ as the average noise level. C_s is set by a value proportional to $\bar{\epsilon}$.

5. Experiments and discussion

5.1. Experimental results

In order to get the performance of detecting occlusion and the effectiveness of modeling occlusion, we firstly compare our occlusion result with several recent approaches: "GC+occl" algorithm by Kolmogorov and Zabih [8] which is a pixel-based approach using a symmetric graph-cut framework to handle occlusion, "Seg+GC" algorithm by Hong and Chen [7] which is a segment-based asymmetric graph-cut approach that does not explicitly detect occlusion, and "Layer" algorithm by Lin and Tomasi [10] which is a combination of pixel-based and segment-based approaches. Two image pairs are used, which are the "tsukuba" and "venus" data sets from [12]. Same parameters are selected for both data sets. We use the source code from Kolmogorov's homepage to compute results of "GC+occl". The non-occlusion results of "Seg.+GC" are downloaded from [12]. The occlusion result is computed by checking the visibility of each point in the non-occlusion result. Result of "Layer" is from the authors' website. The results are shown and compared in Figure 4. Table 1 gives the error statistics for "tsukuba" and "venus" respectively. They are quantitatively evaluated by 3 criteria, which are the percentages of: false positive, false negative, bad points near occlusion. A bad point is a point whose absolute disparity error is greater than one [12]. We make a near occlusion model by dilating the occlusion area to 10 pixels and excluding the occlusion area.

Table 1. Occlusion evaluation for "tsukuba" and "venus". (Best of each group is in italic and bold face)

tsukuba	False pos	False neg	Near occl.
Our results	***1.05%***	30.16%	***4.10%***
GC+occl [8]	1.51%	32.91%	6.44%
Seg+GC [7]	1.19%	32.51%	7.72 %
Layered [10]	2.28%	***25.42%***	8.87%
venus			
Our results	***0.19%***	***16.61%***	***0.54%***
GC+occl [8]	1.88%	32.97%	12.24%
Seg+GC [7]	0.55%	17.73%	0.67%
Layered [10]	0.37%	50.63%	0.90%

From Table 1 we can see that, using the uniqueness and segmentation makes the ratio of false positive than all the others for both images. The ratio of false negative is comparable with others. The boundary of our occlusion results in Figure 4 is cleaner than others', because they are bounded by segmented regions. Our error near occlusion is also obviously better the others. That proves that the performance of visible areas can also benefit correctly detecting occlusion. This experiment shows the power of our patch-based approaches on occlusion handling.

We also summit our results to the standard test bed [12] for dense two-frame stereo algorithms in order to compare

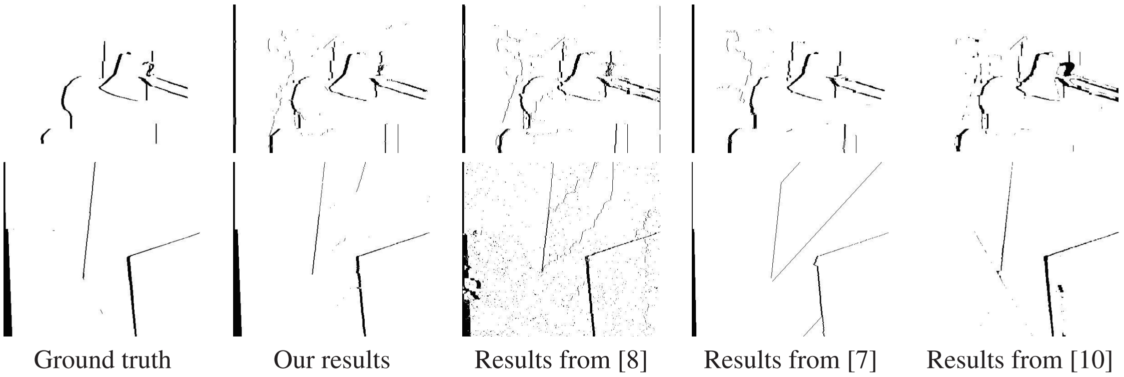


Figure 4. Occlusion results for "tsukuba" (the upper row) and "venus" (the lower row) datasets.

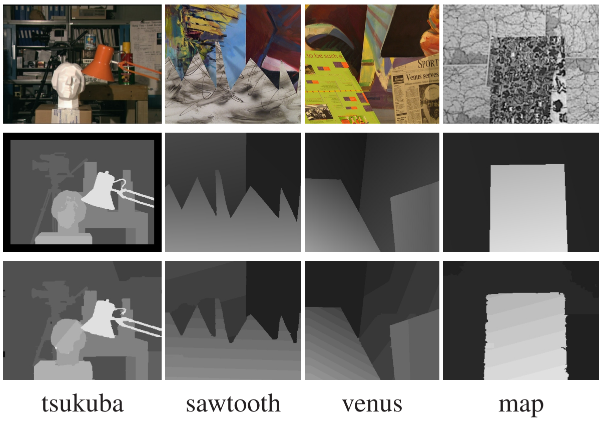


Figure 5. Results of middlebury datasets with automatically chosen parameters. The first row is the left images, the second row is the ground truth, and the third row is our results.

our over-all performance . Disparity maps are shown in Figure 5, and quantitative evaluations are shown in Table 2. Result of "tsukuba" is superior than others' in the list, results of "sawtooth" and "venus" are also on the top level. The patch-based formulation inherits the advantage of segment-based approaches on slanted plane and untextured area. The over-all performance for scenery with only fronto planes (”tsukuba” dataset) is also better than others, because of the well occlusion handling. But the result for "map" is not enough good. We found that artifacts happen at the place where the color foreground object there is very similar to the color of background. The color segmentation failed to segment them into two segment. Discontinuity assumption is thus violated. Similar statistical data can be found in other segment-based approaches. This reveals the limitation of all current segment-based approaches that when discontinuity assumption is not satisfied, disparities in those areas will fail to be correctly computed.

5.2. Discussion

In this section, we give a comparison between our algorithm and other correspondence approaches.

We firstly consider the labeling space. In most pixel-based algorithms, the labeling space is equivalent to the disparity space. Pairwise smoothness assumption (explicitly or implicitly imposed) gives a bias of same disparities for neighboring pixels. This affects the performance of those algorithms when there are greatly slanted planes in the scene. So Ogale and Aloimonos used a 2D linear parameter space for horizontally slanted planes, and many segment-based algorithms used a 3D linear parameter space which can model generally slanted plane. In Table 2, we can find that pixel-based algorithm can easily achieve sound results in "tsukuba" and "map", which only contains fronto or near-fronto planes, while segment-based approaches performs better in "sawtooth" and "venus" which contains horizontally (in "venus") and vertically (in both of them) slanted planes. But the 3D linear space is much larger than 1D disparity space, so a label selection algorithm is often used to select all possible linear parameters before matching.

Another comparison is between the different levels of using segmentation or texture information. In most previous pixel-based algorithms, texture information is used to control the smoothness intention of neighboring points, e.g. in [8], intensity difference between neighboring pixels is used to adjust the smoothness constant. The segment-based algorithms use the color segmentation results as a hard constraint for labels. The points in a segment are considered as a single matching unit, and the number of matching units greatly decreases. This is why they can use the larger 3D linear parameter labeling space than pixel-based approaches. But the segmentation error is also transmitted to the disparity results, which causes the imperfect results for "map". Sun *et al*[13] used the segmentation information in a soft style. The fitted plane information gives a bias to the pixel-based matching, and the results on "sawtooth" and

Table 2. Evaluation results on Middlebury stereo test-bed

Algorithm	Tsukuba			Sawtooth			Venus			Map	
	all	untex.	disc.	all	untex.	disc.	all	untex.	disc.	all	disc.
Sym.BP+occl	0.97_{2}	0.28_{3}	5.45_{2}	$\mathbf{0.19}_{1}$	$\mathbf{0.00}_{1}$	$\mathbf{2.09}_{1}$	0.16_{4}	0.02_{3}	2.77_{6}	$\mathbf{0.16}_{1}$	$\mathbf{2.20}_{1}$
OUR METHOD	$\boldsymbol{\mathit{0.88}}_{1}$	$\boldsymbol{\mathit{0.19}}_{1}$	$\boldsymbol{\mathit{4.95}}_{1}$	$\mathit{0.29}_{5}$	$\boldsymbol{\mathit{0.00}}_{1}$	$\mathit{3.23}_{5}$	$\mathit{0.09}_{2}$	$\mathit{0.02}_{3}$	$\mathit{1.50}_{2}$	$\mathit{0.30}_{7}$	$\mathit{4.08}_{10}$
Segm.-based GC	1.23_{5}	0.29_{5}	6.94_{6}	0.30_{6}	$\mathbf{0.00}_{1}$	3.24_{6}	$\mathbf{0.08}_{1}$	$\mathbf{0.01}_{1}$	$\mathbf{1.39}_{1}$	1.49_{24}	15.46_{29}
Graph+segm.	1.39_{9}	0.28_{3}	7.17_{8}	0.25_{4}	$\mathbf{0.00}_{1}$	2.56_{3}	0.11_{3}	0.02_{2}	2.04_{3}	2.35_{29}	20.87_{33}
Segm.+glob.vis.	1.30_{7}	0.48_{8}	7.50_{10}	0.20_{2}	$\mathbf{0.00}_{1}$	2.30_{2}	0.79_{7}	0.81_{8}	6.37_{11}	1.63_{26}	16.07_{31}
Layered	1.58_{12}	1.06_{14}	8.82_{13}	0.34_{7}	$\mathbf{0.00}_{1}$	3.35_{7}	1.52_{15}	2.96_{25}	2.62_{5}	0.37_{12}	5.24_{12}
Belief prop.	1.15_{3}	0.42_{6}	6.31_{3}	0.98_{14}	0.30_{19}	4.83_{12}	1.00_{10}	0.76_{7}	9.13_{17}	0.84_{21}	5.27_{13}
MultiCam GC	1.85_{15}	1.94_{20}	6.99_{7}	0.62_{12}	$\mathbf{0.00}_{1}$	6.86_{17}	1.21_{12}	1.96_{15}	5.71_{9}	0.31_{9}	4.34_{11}
Region-Progress.	1.44_{10}	0.55_{9}	8.18_{11}	0.24_{3}	$\mathbf{0.00}_{1}$	2.64_{4}	0.99_{9}	1.37_{13}	6.40_{12}	1.49_{25}	17.11_{32}
2-pass DP	1.53_{11}	0.66_{10}	8.25_{12}	0.61_{10}	0.02_{10}	5.25_{13}	0.94_{8}	0.95_{9}	5.72_{10}	0.70_{19}	9.32_{20}
GC+occl.	1.19_{4}	0.23_{2}	6.71_{4}	0.73_{13}	0.11_{13}	5.71_{15}	1.64_{18}	2.75_{23}	5.41_{8}	0.61_{17}	6.05_{15}

“venus” are much better than other pixel-based algorithms. Their approach does not suffer from segmentation error directly, but the plane fitting is in a local style and the fronto bias still exists. Our approach belongs to the segment-based category, but we use more segmentation information for occlusion handling.

6. Conclusions

A patch-based corresponding algorithm using graph-cuts handling occlusions is proposed. Unlike other segment-based approaches, both images are segmented and segments are further separated into patches during matching. More information from segmentation is used. Occlusions are handled in a proper way. The experimental results show performance improvement on occlusion for scenes with slanted planes.

Segmentation error is still a drawback of our approach as other segment-based ones. So in our future work we consider several new technique that can make the algorithm less suffer from the segmentation error. Better results have been obtained and will be reported in recent publications.

Acknowledgement

Thanks Zhiqiang Wang for helping to test the codes. We are grateful to Jian Sun and Xin Zheng for constructive discussion and comments.

References

[1] S. Birthfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *PAMI*, 20(4):401–406, April 1998.

[2] M. Bleyer and M. Gelautz. Graph-based surface reconstruction from stereo pairs using image segmentation. In *Proc. SPIE*, volume 5656, Jan. 2005.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, Nov. 2001.

[4] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *PAMI*, 25(8):993–1008, August 2003.

[5] C. Christoudias, B. Georgescu, and P. Meer. Synergism in low-level vision. In *Proc. ICPR*, volume 4, pages 150–155, 2002.

[6] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *PAMI*, 24(8):1127–1133, August 2002.

[7] L. Hong and G. Chen. Segment-based stereo matching using graph cuts. In *Proc. CVPR*, volume 1, 2004.

[8] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *Proc. ICCV*, 2001.

[9] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph-cuts? In *Proc. ECCV*, pages 232–249, 2002.

[10] M. Lin and C. Tomasi. Surfaces with occlusions from layered stereo. *PAMI*, 26(8):1073–1078, Aug. 2004. http://robotics.stanford.edu/m̃ichelin/layered_stereo.

[11] A. S. Ogale and Y. Aloimonos. Stereo correspondence with slanted surfacecritical implication of horizontal slant. In *Proc. CVPR*, volume 1, pages 568–573, 2004.

[12] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1):7–42, May 2002. http://cat.middlebury.edu/stereo/.

[13] J. Sun, Y. Li, S.-B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. In *Proc. CVPR*, volume 1, 2005.

[14] H. Tao, H. S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *Proc. ICCV*, volume 1, pages 532–539, 2001.

[15] Y. Wei and L. Quan. Region-based progressive stereo matching. In *Proc. CVPR*, volume 1, pages 106–113, 2004.

[16] Y. Zhang and C. Kambhamettu. Stereo matching with segmentation-based cooperation. In *Proc. ECCV*, pages 556–571, 2002.

[17] C. L. Zitnick and T. Kanade. A cooperative algorithm for stereo matching and occlusion detection. *PAMI*, 22(7):675–694, July 2000.