Occlusion detection and filling in disparity map for multiple view synthesis

Conference Paper · January 2012

CITATIONS
4

READS
442

4 authors, including:

Seongyun Cho
Anyang University
40 PUBLICATIONS **889** CITATIONS
SEE PROFILE

Jeongmok Ha
Pohang University of Science and Technology
20 PUBLICATIONS **28** CITATIONS
SEE PROFILE

Hong Jeong
POSTECH POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY
138 PUBLICATIONS **565** CITATIONS
SEE PROFILE

Occlusion Detection and Filling in Disparity Map for Multiple View Synthesis

Seongyun Cho
Dept. of electrical eng.
Postech
Pohang, Korea
Email: cho932@postech.ac.kr

Jeongmok Ha
Dept. of electrical eng.
Postech
Pohang, Korea
Email: jmokha@postech.ac.kr

Insun Sun
Dept. of electrical eng.
Postech
Pohang, Korea
Email: sllinsun@postech.ac.kr

Hong Jeong
Dept. electrical eng.
Postech
Pohang, Korea
Email: hjeong@postech.ac.kr

***Abstract*—In synthesizing intermediate views from two stereo images with the help of stereo matching, we suggest an efficient method for removing occlusion. The proposed method can deal with the problems: noise correction from stereo matching, occlusion removal for high fidelity rendering, and parallel computing for real-time synthesis. The proposed algorithm is compared with the state-of-the-art algorithms to show the better performance in rendering and computational speed.**

***Index Terms*—Multiple view; Stereo matching; Occlusion.**

I. Introduction

Humans can unconsciously obtain three-dimensional (3D) information because they have binocular vision. Using this principle, 3D displays show two different images to two eyes separately. Among the many 3D displays, this paper is dedicated to multi-view display.

The multi-view 3D TV broadcasting suffers some problems. First, multi-view camera array is high price and bulky size. Second, it requires large memory size of multi-view images data. Third, there is bandwidth limit for large data transmission in 3D TV broadcasting system [1]. To solve these problems, the number of the multi-view images should be reduced and intermediate views should be synthesized from the reduced multi-view images. According to the geometric information, intermediate view synthesis field has three main categories: non-geometry-based methods, explicit geometry-based methods, and implicit geometry-based methods [2], [3].

Non-geometry-based method can obtain intermediate views without estimated or given geometric information. Kobota *et al.* reconstructed the novel view directly from the stereo images through space-invariant filtering without geometry estimation [4], [5], [6] . However, the methods show significant error in the occluded regions because this method neglects effect of the occluded regions. Lightfield [7] or Lumigraph [8] rendering needs no or very little geometry information. These methods need many image for rendering.

Explicit geometry-based methods use given geometric information [9], [10]. To get complete geometric information, it needs not only cameras but also additional equipment. These methods can obtain high quality intermediate views. However, these methods have limitation on applications because of the difficulties of obtaining complete geometric information.

Implicit geometry methods need disparity estimation from images [11], [12], [13]. Since the quality of the synthesized intermediate views depends on the accuracy of disparity estimation, selecting matching method is important for multi-view synthesis. Recently, a number of methods have been suggested in this direction.

To infer intermediate view images from stereo images, first of all, we estimated geometric structure (disparity map) using stereo matching with the stereo images. Then, we synthesized intermediate view images using stereo images. Among many existing stereo matching algorithms, we used the fast belief propagation (FBP) [14], [15] stereo matching algorithm that is characterized with the reliable accuracy and computation.

Jin and Jeong [16] and Wang and Zhao [12] used the above approach; however, they showed unnatural results in the occlusion regions and object boundaries because the disparities obtained by BP algorithm have noisy result in the occlusion regions, object boundaries and sharp objects. All the methods based upon disparity suffer from the deficiency in disparity such as noise and occlusion. To remedy these problems, we suggest better methods for detecting occlusion, removal and disparity correction as well as other artifacts.

This paper is organized as follows. In § II, we will define the environments and problems that will be used throughout in this paper. In § III, we propose methods for rectifying occlusion in the given disparity map. In § IV, we propose an intermediate view synthesis method. The experimental results are shown in § V. Finally, we draw conclusions in § VI.

II. Problem Statement

This section describes the optical environment for intermediate view synthesis and explain how to estimate disparity map using BP algorithm.

Let's consider two $M \times N$ color images, I^l and I^r, which are the left and right images, respectively. Our goal is to generate intermediate view I_α from I^l and I^r, where $\alpha(0 \leq \alpha \leq 1)$ denotes the relative intermediate view position between I^L and I^R. The objective is

- Given the two rectified images, I^l and I^r, and the disparity map, D,
- Compute I_α, $0 < \alpha < 1$.

The relationship between the images are illustrated in Fig. 1. A point P in 3-space is projected on the left and right images,

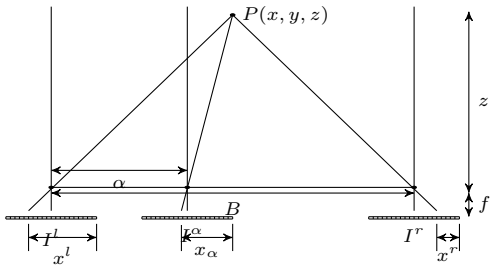


Fig. 1. The geometry of the two rectified images and the Intermediate view

respectively, possibly on different positions. The variable, z, is the depth of P, f is the focal length, x^l and x^r are the horizontal coordinates of the corresponding points on the left and right image, respectively, and B is the horizontal distance between the two cameras. The two cameras are rectified so that the corresponding points appear on the epipolar lines, as specified by the disparity D.

This task is not easy because

- the disparity map is not perfect and contains noisy,
- the occlusion area must be detected and filled naturally, and
- the intermediate image has uncertain areas to be filled correctly.

Our goal is to remedy these problems.

To go further, let's define some of the variables. From Fig. 1, we can obtain the relationship between disparity and depth.

$$d^l = x^l - x^r = \frac{Bf}{z}, \quad d^r = x^l - x^r = \frac{Bf}{z}. \tag{1}$$

We assume that these quantities are given by a preprocessing called stereo matching. The problem is that the two quantities may not be the same since a point in 3-space may be seen by one camera and not be seen by the other camera. These areas of images, called occlusion, are the undefined areas that must be detected and filled appropriately for high quality rendering. Further, due to the limitation of matching, the disparity map is generally noisy. When we use the disparity map without any correction, the resulting synthesized images are very unreliable.

To estimate disparity map, we used belief propagation (BP) algorithm for stereo matching. BP algorithm solves energy minimization problems on Markov random fields (MRFs). The used MRF energy function be represented as follows [17].

$$E(d) = \sum_{p \in P} D_p(d_p) + \sum_{(p,q) \in N} V(d_p, d_q), \qquad d_p \in L, \tag{2}$$

where P denotes a set of image pixels, L denotes a set of disparity labels, d_p denotes disparity of pixel p, N denotes the set of four adjacent pixels around pixel p, $D(d_p)$ denotes data term and $V(d_p, d_q)$ denotes smoothness term. Using BP algorithm, we find disparity d which minimizes the energy function $E(d)$.

$$\hat{d} = \arg\min_d E(d) \tag{3}$$

The energy function (2) is composed of and data term $D(d_p)$ and smoothness term $V(d_p, d_q)$. $D(d_p)$ is the matching cost of assigning label p_d to pixel p, and $V(d_p, d_q)$ is the discontinuity (or smoothness) cost of assigning label p_d to pixel p and label q_d to pixel q.

We defined data term $D_p(d_q)$ and smoothness term $V(d_p, d_q)$ as follows.

$$D_p(d_q) = \lambda \min(\left|I^r(x,y) - I^t(x - d_p, y)\right|, K_D), \tag{4}$$

where I^r denotes intensity value of reference image and I^t denotes intensity value of taget image, K_D denotes truncation value for data term and λ denote a scaling factor.

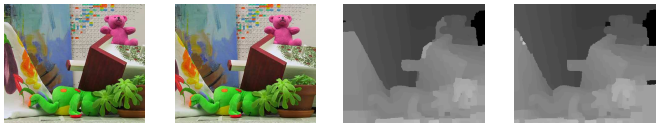
$$V(d_p, d_q) = \min(|d_p - d_q|, K_V), \tag{5}$$

where K_V denotes truncation value for smoothness term.

In (4) and (5), we used a truncated linear cost function for data term and the smoothness term to make energy function $E(d)$ robust to occlusion and artifacts that violate the brightness constancy assumption.

III. Disparity Refinement

We assume the disparity map is given by some stereo matching such as BP [17],[15]. In Fig. 2, the top views are



(a) I^l (b) I^r (c) d^l (d) d^r

Fig. 2. The *Teddy* images and those disparity maps obtained by BP algorithm. (a)-(b): the left and right images, (c)-(d): the left and right disparities.

the left and right images, I^l and I^r; the bottom views are the corresponding disparity maps, d^l and d^r.

The occlusion is where there is inconsistency between left and right disparity maps [18]. To represent this concept quantitatively, we first consider the disparity as mapping, $d^l : x_l \to x^l + d(x_l)$ and $d^r : x^r \to x^r + d^r(x_r)$. For the consistent region, the twice mapping must return to the original site: $d^r(x^l + d^l(x^l)) + x_l + d^l(x^l) = x^l$ or $x^r + d^r(x^r) + d^l(x^r + d^r(x^r)) = x^r$. Let's define $C^l(x_l) \triangleq d^l(x_l) + d^r(x_l + d^l(x_l))$ and $C^r(x^r) \triangleq d^r(x^r) + d^l(x^r + d^r(x^r))$. Then, if $C(x) = 0$, the point x is consistent, otherwise occlusion [18].

We relax the strict definition by introducing a threshold $T_h \approx 2$ so that varying degrees of occlusion could be obtained. The larger T_h, the sparser the occlusion will be. Using this notation, we can obtain the *occlusion map*, $O = \{o(x,y) | o(x,y)\delta(C(x,y) \le T_h)\}$, where $\delta(x)$ is an indication function having 1 if x is true, 0 otherwise. This process can be realized by scanning the epipolar line from left to right in raster manner.

For the Teddy example, the occlusion region thus found are shown in Fig. 3. The occlusion map, O, is sparse in general.

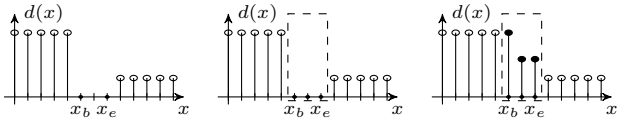
(a) The occlusion map O^l for d^l (b) The occlusion map O^r for d^r

Fig. 3. The Teddy example: the red regions are occluding areas of d^l and d^r, respectively.

In the following, let's examine the occlusion map and find a way to refine any anomaly in it.

A. Rectifying the Thin Occlusion

It is observed that if the region of occlusion is very thin, the boundaries are etched out by the smoothness term. The true boundary must be recovered. Fig. 4 illustrates such case. In the first figure, occlusion between x_b and x_e is very thin

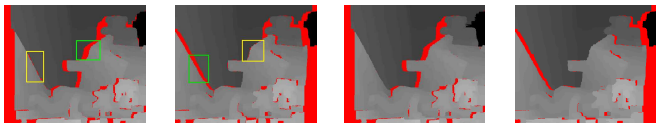


(a) Thin occlusion (b) uncertain region (c) our goal

Fig. 4. Dealing with the thin occlusion

and must be restored to the original sharp boundary as shown in the third. The true boundary is in-between the starting and end points and thus must be found by some optimal decision.

For example, Fig. 5 contains this type of occlusion. The red



(a) d^l (b) d^r (c) d^l after correction (d) d^r after correction

Fig. 5. Rectifying the smoothing errors. (a)-(b): the smoothing errors in yellow boxes. (c)-(d): after the rectification.

pixels in the yellow box are the types of thin occlusion. The red pixels in the green box are the ordinary occlusion, which can be readjusted at the last stage.

Let's consider a thin occlusion region, $[x_b^l, x_e^l]$ in the occlusion map O^l, where $x_e^l - x_b^l < \eta_t$ for some small $\eta_t \approx 2$. Then, the corresponding region in the other image must be wider than $x_e^l - x_b^l$. This can be represented by the disparity: $d^l(x_b^l - 1) > d^l(x_e^l + 1)$ for d^l and $d^r(x_b^r - 1) < d^r(x_e^r + 1)$ for d^r.

Inside such a region, $[x_b^l, x_e^l]$, we have to find a true boundary so that different disparity values can be assigned before and after that boundary. An optimal boundary must satisfy the photometric constraint:

$$x^* = \arg\min_{a \in [x_b, x_e]} \sum_{x=x_b}^{a} \left| I^l(x) - I^r(x - d^l(x_b - 1)) \right| + \sum_{x=a+1}^{x_e} \left| I^l(x) - I^r(x - d^l(x_e + 1)) \right|. \quad (6)$$

Note that no smoothness term is involved here so that the boundary, originally sharp but smoothed by the stereo matching, can be recovered.

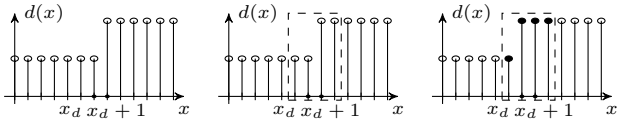
Once the optimal boundary is found, the pixels in $[x_b^l, x_e^l]$, can be filled as

$$d(x) = \begin{cases} d(x_b - 1), & x \in [x_b, x^*], \\ d(x_e + 1), & x \in [x^* + 1, x_e]. \end{cases} \quad (7)$$

Fig. 5(c) and (d) are the results of this method. One can notice that once smoothed boundaries, marked with yellow boxes, are all sharpened.

B. Rectifying Discontinuous Disparity Area

Regardless of the occlusion, disparity errors might occur where the disparity changes abruptly, again due to the nature of smoothness as shown in Fig. 6 [19]. Let's detect such regions and adjust them accordingly. In the first figure, the



(a) discontinuous disparity (b) uncertain region (c) our goal

Fig. 6. Dealing with discontinuous disparity

disparity change rapidly from x_d to $x_d + 1$. However, the true boundary must be detected without any intervention of the smoothness role of matching.

A discontinuous site, x_d, can be detected by observing $|d^l(x_d, y) - d^l(x_d + 1, y)| > \eta_d$ for some small $\eta_d \approx 2$. Around a small neighbor $\alpha \approx 2$ around x_d, we can determine an optimal boundary:

$$x^* = \arg\min_{x_d - \alpha \le a \le x_d + \alpha} \sum_{x=x_d-\alpha}^{a} \left| I^l(x) - I^r(x - d^l(x_d)) \right| + \sum_{x=a+1}^{x_d+\alpha+1} \left| I^l(x) - I^r(x - d^l(x_d + 1)) \right|. \quad (8)$$

Note also that no smoothness term is involved here.

Once the optimal boundary is found, pixels in $[x_d - \alpha, x_d + \alpha]$ can be assigned with the disparity values:

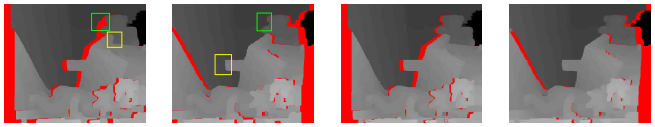
$$d(x) = \begin{cases} d(x_d), & x \in [x_d - \alpha, x^*], \\ d(x_d + 1), & x \in [x^* + 1, x_d + \alpha + 1]. \end{cases} \quad (9)$$

In Fig. 7, the inconsistent occlusion regions are denoted by green boxes. The inconsistent occlusion regions occur though

disparity and occlusion refinement. For occlusion consistency, The pixels in this region are filled by

$$d^l(x,y) = d^r(x + d^r(x,y)) \quad \text{if} \quad d^l(x,y) = \phi \quad \text{and} \quad d^r(x + d^r(x,y)) \neq \phi. \tag{10}$$

Fig. 7 shows the results for the discontinued disparities and the inconsistent occlusion region.

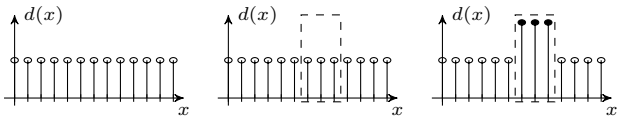


(a) d^l (b) d^r (c) d^l after refinement (d) d^r after refinement

Fig. 7. The area of the discontinued disparity is refined and the inconsistent occlusion regions is filled. (a)-(b): the smoothing errors in yellow boxes and inconsistent occlusion region in green boxes. (c)-(d): after correction.

C. Rectifying the Narrow Object

For small pointed or thin objects, the disparities tend to be influenced overwhelmingly by the bigger neighbor values again due to the nature of smoothness [19]. This is shown in Fig. 8. In the first figure, a narrow object is missing in the

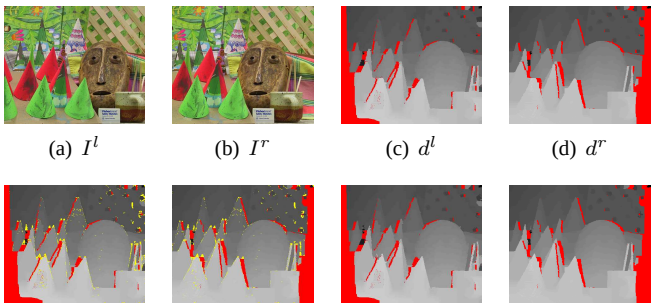


(a) narrow object (b) uncertain region (c) our goal

Fig. 8. Dealing with narrow objects

disparity map. Avoiding smoothness operations, the objects must be detected by the stereo matching around this place.

In Fig. 9, the *Cones* images show such effects. In Figs. 9(c)



(a) I^l (b) I^r (c) d^l (d) d^r

(e) d^l, PED error marked (f) d^r, PED error marked (g) d^l after refinement (h) d^r after refinement

Fig. 9. Small objects: (a)-(b) original images, (c)-(d) disparity map and occlusion map, (e)-(f) small objects in yellow marks, and (g)-(h) after refinement.

and (d), one can observe the disparity errors around the cone tips and chapsticks.

Let's define the *photometric disparity error detection* (PED) function to detect errorsome disparities by photometric difference [20]. For the thresholds, $t_1 < t_2$, we define

$$S^l(x,y,d) = 1 - U(t_1 - \min_{c\in\{R,G,B\}} \{|I^{l,c}(x,y) - I^{r,c}(x-d,y)|\}) \cdot U(t_2 - \max_{c\in\{R,G,B\}} \{|I^{l,c}(x,y) - I^{r,c}(x-d,y)|\}), \tag{11}$$

where $U(t)$ is unit step function which returns 1 when $t \geq 0$, otherwise 0. If $S^l(x,y,d)$ or $S^r(x,y,d)$ is one, the disparity d is photometric erroneous; otherwise, the disparity d is photometric correct. The regions detected by PED are marked in Figs. 9(e) and (f).

For each of such points, (x,y), we propagate the disparities of the 8 neighbors to it. As the iteration proceeds, the disparities of the neighbor, $N(x,y)$, are stacked up as the disparity set $\{D_i(x,y) | D_i(x,y) \subset [0, L-1]\}$ for L disparity levels. In the ith iteration,

$$D_i(x,y) = \begin{cases} \bigcup_{s\in\{N(x,y)\}} \{d(s)\}, & i = 1, \\ \bigcup_{s\in\{N(x,y)\}} \{D_{i-1}(s)\}, & i \geq 2. \end{cases} \tag{12}$$

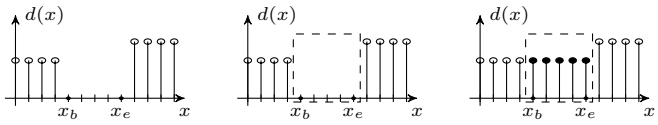
After T iterations, $i = T$, we select the minimum disparity which are photometric correct ($S(x,y,d) = 0$). If there is no d which is in $D(x,y)$ and photometric correct ($S(x,y,d) = 0$), we select the minimum disparity.

$$d(x,y) = \begin{cases} \min\{d \mid S(x,y,d) = 0, \text{ if } d \in D(x,y)\}, \\ \min\{D(x,y)\}, \\ \text{ if } \{d \mid S(x^s,y,d) = 0, d \in D(x,y)\} = \emptyset. \end{cases}$$

Fig. 9(g) and (f) are the results. Some of the pointed and thin objects are enhanced even if there still remain many errornous places.

D. General Occlusion Filling

After the special cases of occlusion are remedied, the remnants are the ordinary occlusion regions. This is shown in Fig. 10. The first figure shows a region of occlusion, specified



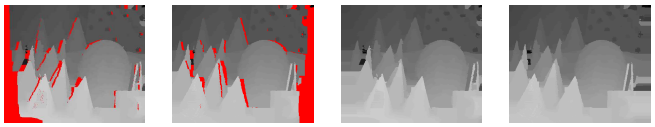
(a) general occlusion (b) uncertain region (c) our goal

Fig. 10. Dealing with general occlusion

by $[x_b, x_e]$. The original disparities are a smoothed version due to the stereo matching.

For a region of occlusion, $[x_b, x_e]$, each the pixels are filled by with the disparities on both sides. That is, we assign them with the smaller disparity of the both side disparities, $d(x_e - 1, y)$ and $d(x_b + 1, y)$. For each $x_b \leq x \leq x_e$,

$$d^l(x) = \begin{cases} d^l(x_b - 1), & d(x_b - 1) < d(x_e + 1), \\ d^l(x_e + 1), & d(x_b - 1) > d(x_e + 1). \end{cases} \tag{13}$$



(a) d^l (b) d^r (c) d^l after refinement (d) d^r after refinement

Fig. 11. Occlusion filling. (a)-(b): before, (c)-(d): after occlusion filling.

Fig. 11 shows the result. The red region in the images are filled according to the proposed method. The bottom images contain much lesser uncertain areas.

IV. Intermediate View Synthesis

Even if the disparity map is rather accurate, there arise new problems when we try to generate intermediate images from it. One problem arises when the colors at the object boundary differ greatly. Figs. 12 illustrates such example. An object B is

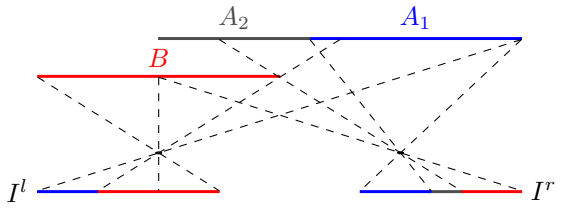


Fig. 12. Projection of two objects. A: background object, B: foreground object, I^l: left image, and I^r: right image.

layered above a background object A. This scene is projected to the two images, I^l and I^r.

This example can be redrawn as in Fig. 13. The problem

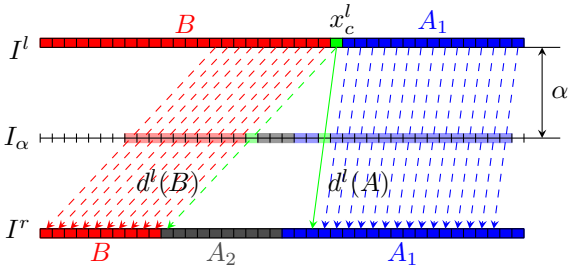
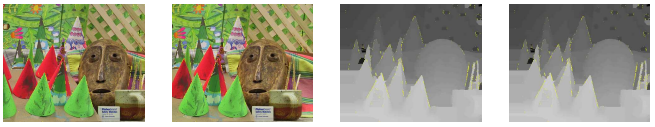


Fig. 13. The trouble at object boundary. x_c^l: object boundary between object A and B, I_α: intermediate view at position α, $d^l(A)$: disparity of object A, $d^l(B)$: disparity of object B.

occurs at the occlusion x_c', where the colors change. Due to the nature of optics, the color at this site is usually a mixture of the colors from both sides of the boundary. There is a clear confusion which colors we have to choose for the boundary pixel, x_c^l, which may result in different assignment of disparities for the intermediate view. In Fig. 13, the two cases are marked with green solid and dashed lines, respectively. If $d^l(A)$ is chosen, the mixed color $I^l(x_c^l)$ must be placed on I_α between A_1 and A_2, otherwise $I^l(x_c^l)$ must be placed on I_α between B and A_2. $I^l(x_c^l)$ is clearly problematic because, whichever we choose, both cases make a striking difference on I_α. Therefore, we'd better exclude such pixels and disparities before synthesis.

In general, $I^l(x_c^l)$ at the object boundary occurs where the disparity is discontinuous and their PED value is one. Accordingly, we can obtain the pixel set $X_c = \{x_{c,k} | k \in [0, K]\}$, where K is the number of such regions. We exclude the detected pixel set X_c and their disparities from input images and disparity maps.

As an example, the yellow marked pixels in Fig. 14 represent pixel set X_c, which were detected, eliminated. Note that most of the yellow marked pixels are concentrated on the object boundaries.



(a) I^l (b) I^r (c) d^l (d) d^r

Fig. 14. Dealing with the problematic pixels at the object boundaries. (a)-(b): the modified input images. (c)-(d): the modified disparity maps. yellow-marked pixels: the problem pixels.

Once X_c removed, the intermediate views can be built by linear interpolation. Two candidate intermediate views are available at position α, I_α^{l*}, corresponding to the left images, I^l.

$$I_\alpha^{l*}(x^l - \alpha d^l(x^l, y), y) = I^l(x^l, y). \tag{14}$$

Similarly, the same process can be done for the two candidate disparity maps.

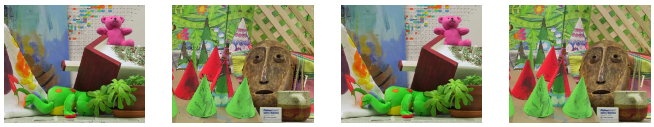
$$d_\alpha^{l*}(x^l - \alpha d^l(x^l, y), y) = d^l(x^l, y). \tag{15}$$

When I_α^{l*} and I_α^{r*} are combined into I_α by comparing d_α^{l*} and d_α^{r*}, the three cases may arise: 1) only one pixel in I_α^{l*} or I_α^{r*} exist at particular location (x_α, y) 2) two pixels in I_α^{l*} and I_α^{r*} exist at a particular location (x_α, y), and their disparities ($d_\alpha^{l*}(x_\alpha, y)$ and $d_\alpha^{r*}(x_\alpha, y)$) are unequal and 3) two pixels in I_α^{l*} and I_α^{r*} exist at a particular location (x_α, y), and their disparities ($d_\alpha^{l*}(x_\alpha, y)$ and $d_\alpha^{r*}(x_\alpha, y)$) are equal.

For the first case, we simply choose the existing pixel for $I_\alpha(x_\alpha, y)$. In the second case, choose the pixel with the greater disparity for $I_\alpha(x_\alpha, y)$ because the greater is closer to the camera and lesser is behind greater [10]. In the third case, combine $I_\alpha^{l*}(x_\alpha, y)$ and $I_\alpha^{r*}(x_\alpha, y)$ into $I_\alpha(x_\alpha, y)$.

$$I_\alpha(x_\alpha, y) = (1 - \alpha) I_\alpha^{l*}(x_\alpha, y) + \alpha I_\alpha^{r*}(x_\alpha, y). \tag{16}$$

As an example, Fig. 15 shows the results of the images at position $\alpha = 0.5$. The holes (Figs. 15(a) and (b) represent the eliminated pixels. Since the number of the holes are small in Figs. 15(c) and (d), the simple linear interpolation can naturally fill the gaps as shown.



(a) I_α before linear interpolation (b) I_α before linear interpolation (c) I_α after linear interpolation (d) I_α after linear interpolation

Fig. 15. The results of linear interpolation ($\alpha = 0.5$). (a)-(b): before linear interpolation. (c)-(d): after linear interpolation

TABLE I
PARAMETERS IN BP AND DISPARITY CORRECTION

λ	K_D	K_V	T_h	t_1	t_2	T
0.07	15	1.7	2	20	25	60

V. EXPERIMENTAL RESULTS

Let's compare the methods explained in § III and § IV with others. Parameters in the implementation are denoted in Table I. Here, λ, K_D and K_V are for BP algorithm that is described in §II. We referred to Felzenszwalb's paper [17] for λ, K_D and K_V. Also, T_h is the thresholds for checking disparity consistency. The parameters, t_1 and t_2, are the low and upper thresholds of PED function. The remaining parameter, T, is the number of the iteration to find candidate disparities.

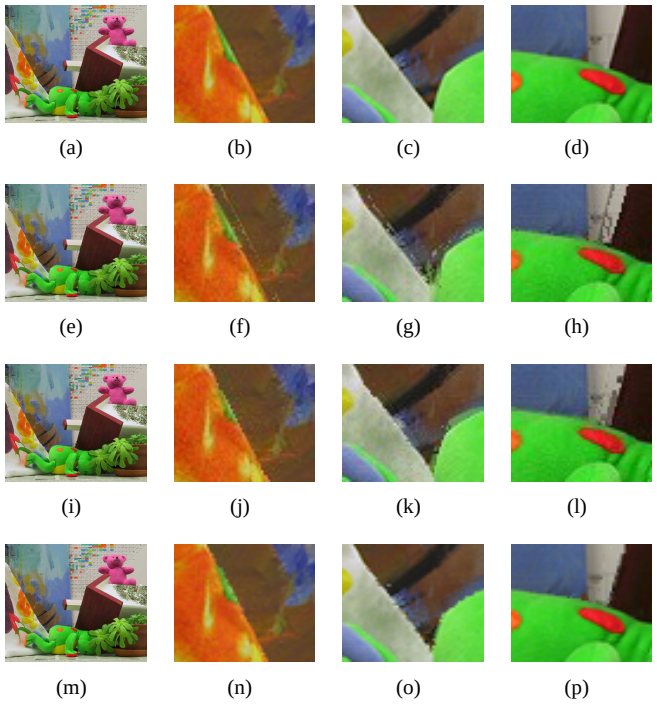
Let's consider the algorithm complexity. First, $O(MN)$ time complexity is required to check the disparity consistency. The correction of narrow occlusion, discontinuous disparity, narrow objects, and general occlusion need, respectively, $O(MN)$, $O(5MN)$, $O(MNT)$, and $O(MN)$. The image synthesis needs $O(MN)$ time complexity. As we can see, all the operations are within $O(MN)$ which means real time computations.

For the experiment, we based on Middlebury sample images (http://vision.middlebury.edu/stereo/) which are rectified stereo images. We compared our method with Jin [16] and Wang [12]. Jin's method intactly uses disparity maps obtained by classical BP algorithm for intermediate view synthesis. Wang's method uses adaptive BP algorithm to obtain disparity map. Disparity maps obtained by adaptive BP algorithm has better result than classical BP algorithm. In Wang's method, however, error rate at discontinued areas, which are crucial for high-quality view synthesis, is still high. So, we can observe unnatural results in the Wang's method.

According to the algorithms, the intermediate images have been generated for each *Venus*, *Teddy*, and *Cones*. In each set of experiments, the second and sixth images of the image sequence are the left and right images, respectively ($\alpha = 0, 1$), and the remaining images are the intermediate images with $\alpha = 0.25, 0.5, 0.75$.

A. Teddy Experiments

For the qualitative evaluations, we used the left and right images of *Teddy* for input image, and generated an intermediate image viewed at center, $\alpha = 0.5$). Figs. 16 shows the results. The first row of images are the ground truth images.



(a) (b) (c) (d)
(e) (f) (g) (h)
(i) (j) (k) (l)
(m) (n) (o) (p)

Fig. 16. Comparison of the intermediate image at $\alpha = 0.5$ for *Teddy*. (a)-(d): ground true images with magnified parts. (e)-(h): Jin's result. (i)-(l): Wang's result. (m)-(p): our result.

The left most is the left image and the others are the magnified versions. The first column is the same images for the left image. The second, third, and the fourth rows are the results from Jin, Wang, and Ours. Notice that in the second and third column, the phantom edges which are apparent in the other results are invisible in our result. In the fourth column, the spurious edges in other results are also not clearly removed in our result.

We can observe that our method outperformed at the object boundaries and disparity discontinuity areas than other methods. Because our method preserves the accuracy in disparity discontinuity areas which are crucial for high-quality view synthesis and eliminates linearly interpolated colors at the object boundaries which make a problem for high-quality view synthesis, our method can outperformed at the object boundaries and disparity discontinuity areas than other methods.

B. Qualitative Assessment

For the quantitative evaluations, we used the left and right images from *Cones*, *Teddy*, and *Venus* images, and generated intermediate views at $\alpha = 0.25$, $\alpha = 0.5$ and $\alpha = 0.75$, where the real camera images are available. Then we evaluate peak signal-to-noise ratios (PSNR) between our synthesized images and real camera images, and calculate average PSNR of view synthesis at $\alpha = 0.25$, $\alpha = 0.5$ and $\alpha = 0.75$. Table II presents the quantitative evaluations of our proposed views and other results. It is clear that our result is better than the other methods in PSNR evaluations. The small difference in PSNR means a lot since the number of pixels in problem area

TABLE II
PSNR OF THE SYNTHETIC IMAGES BY JIN, WANG, AND OURS

	Cones	Teddy	Venus
Jin	27.38 dB	29.33 dB	31.36 dB
Wang	27.69 dB	30.37 dB	31.50 dB
Ours	28.01 dB	30.44 dB	32.88 dB

is very small compared to the large image size. That is, if we are confined to the problem area, the difference in PSNR will be great. That's the reason we have to observe also the actual images for quality assessment where the differences are apparent.

VI. CONCLUSION

In this paper, we proposed two classes of algorithms: disparity enhancement and reliable synthesis of multi-view images. As for the disparity, thin, discontinuous, narrow object, and general occlusions are rectified for better rendering. For the synthesis, the ambiguity around color change is resolved for intermediate images. Our results have been compared with others in terms with qualitative and quantitative assessments.

With the binocular images and stereo matching, we can synthesize all the intermediate views fulfilling the original goal of the multi-view synthesis.

ACKNOWLEDGMENT

This work has been supported by the following funds: the Brain Korea 21 Project, the Ministry of Knowledge Economy, Korea, under the Core Technology Development for Breakthrough of Robot Vision Research support program supervised by the National IT Industry Promotion Agency. This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the IT Consilience Creative Program support program supervised by the NIPA (National IT Industry Promotion Agency)

REFERENCES

[1] C. Lu, H. Wang, H. Ren, and Y. Shen, "Virtual view synthesis for multi-view 3D display," in *2010 Third International Joint Conference on Computational Science and Optimization*, 2010.

[2] S. Chan, H. Y. Shum, and K. T. Ng, "Image-based rendering and synthesis," *IEEE SIGNAL PROCESSING MAGAZINE*, vol. November 2007, pp. 22–33, 2007.

[3] H.-Y. Shum and S. B. Kang, "A review of image-based rendering techniques," in *in Proc. Visual Communications and Image Processing 2000*, 2000.

[4] A. Kubota, K. Aizawa, and T. Chen, "Reconstructing dense light field from array of multifocus images for novel view synthesis," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 16(1), pp. 269–279, 2007.

[5] A. Kubota, T. Hamanaka, and Y. Hatori, "View interpolation using defocused stereo images: A space-invariant filtering approach," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009.

[6] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, "Unstructured lumigraph rendering," in *SIGGRAPH '01 Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001.

[7] M. Levoy and P. Hanrahan, "Light field rendering," in *in Proc. SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

[8] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen, "The lumigraph," in *in Proc. SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

[9] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach," in *in Proc. SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

[10] A. K. Jain, L. C. Tran, R. Khoshabeh, and T. Q. Nguyen, "Efficient stereo to multiview synthesis," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[11] L. Zhang, D, Wang, D., and A. Vincent, "An adaptive object-based reconstruction of intermediate views from stereoscopic images," in *2001. Proceedings. 2001 International Conference on Image Processing*, 2001.

[12] C. Wang and L. Zhao, "Intermediate view synthesis based on adaptive BP algorithm and view interpolation," *Journal of Convergence Information Technology*, vol. 5(10), pp. 72–81, 2010.

[13] I.-L. Jung, T. Chung, K. Song, and C.-S. Kim, "Virtual view synthesis using multi-view video sequences," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 2009.

[14] S. Park and H. Jeong, "Real-time stereo vision FPGA chip with low error rate," in *International Conference on Multimedai and Ubiquitous Engineering*, 2007.

[15] ——, "VLSI architecture for MRF based stereo matching," *International Journal of Computer Vision*, vol. 4599, pp. 55–64, 2007.

[16] C. Jin and H. Jeong, "Intermediate view synthesis for multi-view 3D displays using belief propagation based stereo matching," in *Third 2008 International Conference on Convergence and Hybrid Information Technology*, 2008.

[17] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *International Journal of Computer Vision*, vol. 70(1), p. 41 54, 2006.

[18] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *Conference on Computer Vision and Pattern Recognition*, 1997.

[19] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, p. 7 42, 2002.

[20] J. Lu, S. Rogmans, G. Lafruit, and F. Catthoor, "Stream-centric stereo matching and view synthesis: A high-speed approach on gpus," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, vol. 19(11), pp. 1598–1611, 2009.