Occlusion Removal and Filling in Disparity Map for Real Time Multiple View Synthesis

Seongyun Cho, Insun Sun, Jeongmok Ha, Hong Jeong

Pohang University of Science and Technology (Postech)

slinkshot@naver.com
sllinsun@postech.ac.kr
jmokha@postech.ac.kr
hjeong@postech.ac.kr

Abstract

In synthesizing multiple intermediate views from two stereo using stereo-matching, we suggest an efficient method for removing occlusion. We dealt with the noise correction from stereo matching, occlusion removal for high fidelity rendering, and parallel computing for real-time synthesis. The proposed algorithm shows the better performance in rendering quality and computational speed.

Keywords: Multiple View, Stereo Matching, Occlusion

1. 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 images 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.

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 Section 2, we will define the environments and problems that will be used throughout in this paper. In Section 3, we propose methods for rectifying occlusion in the given disparity map. In Section 4, we propose an intermediate view synthesis method. The experimental results are shown in Section 5. Finally, we draw conclusions in Section 6.

2. Problem statement

This section describes the optical environment for intermediate view synthesis and explains 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 $I_\alpha$ 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 $I_\alpha$, $0 \leq \alpha \leq 1$.

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

![Figure 1. The geometry of the two rectified images and the Intermediate view](image)

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

$$
\begin{align*}
    d^l &= x^l - x^r = \frac{bf}{z}, \\
    d^r &= x^l - x^r = \frac{bf}{z}.
\end{align*}
$$

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].

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

(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 $p$, $P$ denotes data term and $V(d_p, d_q)$ denotes smoothness term. Using BP algorithm, we find disparity $d$ which minimizes the energy function $E(d)$.

$$\hat{d} = \arg\min_{d \in L} E(d) ,$$

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 $d_q$ to pixel $p$ and label $d_q$ to pixel $q$.

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

$$D_p(d_p) = \lambda \min(I^r(x, y) - I^t(x - d_p, y), K_p),$$

where $I^r$ denotes intensity value of reference image and $I^t$ denotes intensity value of target image, $K_p$ denotes truncation value for data term and denote a scaling factor.

$$D_p(d_p) = \lambda \min(I^r(x, y) - I^t(x - d_p, y), K_p),$$

where $K_p$ 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.

### 3. Disparity Refinement

We assume the disparity map is given by some stereo matching such as BP [17],[15]. Figure 2 shows the left and right images, $I^t$ and $I^r$ and corresponding disparity maps, $d^t$ and $d^r$.

![Figure 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.](image)

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, $d^l: x^l \rightarrow x^l + d(x^l)$ and $d^r: x^r \rightarrow x^r + d^r(x^r)$. For the consistent region, the twice mapping must return to the original site: $x^l + d^l(x^l + d^l(x^l)) + d^l(x^l) = x^l$ or $x^r + d^r(x^r) + d^l(x^r + d^r(x^r)) = x^r$. Let’s define $C^l(x^l) \equiv d^l(x^l) + d^r(x^l + d^l(x^l))$ and $C^r(x^r) \equiv d^l(x^r + d^r(x^r)) + d^l(x^r + d^r(x^r))$. 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. Using this notation, we can obtain the occlusion map, $(x, y) = \{o(x, y)\}$ where $o(x, y) = \delta(C(x, y) > T_h)$, where $\delta(x)$ is an indicator 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 found occlusion are shown in Figure 3. The occlusion map, $O$, is sparse in general. In the following, let’s examine the occlusion map and find a way to refine any anomaly in it.
3.1 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. Figure 4 illustrates such case. In the first figure, occlusion between \( x_b \) and \( x_e \) is very thin 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, Figure 5 contains this type of occlusion. The red 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, \([x_b^t, x_e^t]\) in the occlusion map \( O^t \), where \( x_b^t - x_e^t < \eta_t \) for some small \( \eta_t \). Then, the corresponding region in the other image must be wider than \( x_b^t - x_e^t < \eta_t \). This can be represented by the disparity \( d^t(x_b^t - 1) > d^t(x_e^t + 1) \) for \( d^t \) and \( d^r(x_b^r - 1) < d^r(x_e^r + 1) \) for \( d^r \). Inside such a region, \([x_b^t, x_e^t]\), 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:

\[
x^* = \arg\min_{x \in [x_b^t, x_e^t]} \sum_{x=x_b^t}^{x_e^t} |l^t(x) - l^r(x - d(x_b^t - 1))| + \sum_{x=x_e^t+1}^{x_e^r} |l^t(x) - l^r(x - d(x_e^t + 1))|,
\]

(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 \([x_b^t, x_e^t]\) can be filled as

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

(7)
Figure 5(c) and (d) are the results of this method. One can notice that once smoothed boundaries, marked with yellow boxes, are all sharpened.

3.2 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 Figure 6. [19]. Let’s detect such regions and adjust them accordingly. In the first figure, the disparity changes rapidly from $x_d$ to $x_d + 1$. However, the true boundary must be detected without any intervention of the smoothness role of matching.

![Figure 6. Dealing with discontinuous disparity.](image)

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

$$x^* = \arg \min_{x_d - \alpha \leq x \leq x_d + \alpha} \sum_{x=x_d-\alpha}^{x_d+\alpha+1} |l^l(x) - l^r(x - d^l(x_d))| + \sum_{x=x_d+1}^{x_d+\alpha+1} |l^l(x) - l^r(x - d^l(x_d + 1))|$$

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:

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

For occlusion consistency, The pixels in this region are filled by

$$d^l(x, y) = d^r(x + d^l(x, y)), \text{ if } d^l(x, y) \neq \phi \text{ and } d^r(x + d^l(x, y)) \neq \phi.$$  \hspace{2cm} (10)

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

![Figure 7.](image)

3.3 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 Figure 8. In the first figure, a narrow object is missing in the disparity map. Avoiding smoothness operations, the objects must be detected by the stereo matching around this place.
In Figure 9, the Cones images show such effects. In Figs. 9(c) and (d), one can observe the disparity errors around the cone tips and chopsticks. Let’s define the photometric disparity error detection (PED) function to detect erroneous disparities by photometric difference [20]. For the thresholds, \( t_1 < t_2 \), we define

\[
S^t(x,y,d) = 1 - U\left(t_1 - \min_{e \in \{x, y, d\}} \{l^t(x,y) - l^e(x-d,y)\}\right) \cdot U\left(t_2 - \max_{e \in \{x, y, d\}} \{l^t(x,y) - l^e(x-d,y)\}\right)
\]

where \( U(t) \) is unit step function which returns 1 when \( t \geq 0 \), otherwise 0. If \( S^t(x,y,d) \) or \( S^e(x,y,d) \) is one, the disparity \( d \) is photometrically erroneous; otherwise, the disparity \( d \) is photometrically correct. The regions detected by PED are marked in Figure 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 \( D_i(x,y) \in [0, L-1] \) for \( L \) disparity levels. In the \( i \)th iteration,

\[
D_i(x,y) = \begin{cases} 
\bigcup_{s \in N(x,y)} \{d(s)\}, & i = 1, \\
\bigcup_{s \in N(x,y)} \{d_{i-1}(s)\}, & i \geq 2.
\end{cases}
\]

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 photometrically correct (\( S(x,y,d) = 0 \)), we select the minimum disparity.

\[
d(x,y) = \begin{cases} 
\min\{d|S(x,y,d) = 0, \ d \in D(x,y)\}, & \text{if } (d|S(x,y,d) = 0, \ d \in D(x,y)) = \phi,
\end{cases}
\]

Figure 9(g) and (f) are the results. Some of the pointed and thin objects are enhanced even if there still remain many erroneous places.
3.4 General Occlusion Filling

After the special cases of occlusion are remedied, the remnants are the ordinary occlusion regions. This is shown in Figure 10. The first figure shows a region of occlusion, specified 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 \(d(x_e + 1, y)\). For each \(x_b \leq x \leq x_e\),

\[
    d^l(x) = \begin{cases} 
        d(x_e - 1), & d(x_e - 1) < d(x_e + 1), \\
        d(x_e + 1), & d(x_e - 1) > d(x_e + 1). 
    \end{cases}
\]

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

4. 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. Figure 12 illustrates such example. An object B is 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 Figure 13. The problem occurs at the occlusion \(x^l_0\), 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, \(x^l_1\), which may result in different assignment of disparities for the intermediate view. In Figure 13, the two cases are marked with green solid and dashed lines, respectively. If \(d^l(A)\) is chosen, the
mixed color $I'(x_f')$ must be placed on $I_a$ between $A_1$ and $A_2$, otherwise $I'(x_f')$ must be placed on $I_b$ between $B$ and $A_2$. $I'(x_f')$ is clearly problematic because, whichever we choose, both cases make a striking difference on $I_a$. Therefore, we’d better exclude such pixels and disparities before synthesis.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{image13.png}
\caption{The trouble at object boundary. $x'_c$: object boundary between object A and B, $I_c$: intermediate view at position $\alpha$. $d'(A)$: disparity of object A, $d'(B)$: disparity of object B.}
\end{figure}

In general, $I'(x_f')$ at the object boundary occurs where the disparity is discontinuous and its PED value is one.

Accordingly, we can obtain the pixel set $X_e = \{ x_{e,k} | k \in [0, K] \}$, where $K$ is the number of such regions. We exclude the detected pixel set $X_e$ and their disparities from input images and disparity maps.

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

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{image14.png}
\caption{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.}
\end{figure}

Once $X_e$ removed, the intermediate views can be built by linear interpolation. Two candidate intermediate views are available at position $\alpha$, $I^{\alpha}_a$, corresponding to the left images, $I'$. 

$$I^{\alpha}_a(x^l - \alpha d'(x^l, y), y) = I'(x^l, y). \quad (15)$$

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

$$d^\alpha_a(x^l - \alpha d'(x^l, y), y) = d'(x^l, y) \quad (16)$$

When $I^\alpha_a$ and $I^\alpha_b$ are combined into $I_a$ by comparing $d^\alpha_a$ and $d^\alpha_b$, the three cases may arise: 1) only one pixel in $I^\alpha_a$ or $I^\alpha_b$ exist at particular location $(x_a, y)$, 2) two pixels in $I^\alpha_a$ and $I^\alpha_b$ exist at a particular location $(x_a, y)$, and their disparities $d^\alpha_a(x_a, y)$ and $d^\alpha_b(x_a, y)$ are unequal and 3) two pixels in $I^\alpha_a$ and $I^\alpha_b$ exist at a particular location $(x_a, y)$, and their disparities $d^\alpha_a(x_a, y)$ and $d^\alpha_b(x_a, y)$ are equal. For the first case, we simply choose the existing pixel for $I_a(x_a, y)$. In the second case, choose the pixel with the greater disparity for $I_a(x_a, y)$ because the greater is closer to the camera and lesser is behind greater [10]. In the third case, combine $I^\alpha_a(x_a, y)$ and $I^\alpha_b(x_a, y)$ into $I_a(x_a, y)$.

$$I_a(x_a, y) = (1 - \alpha) I^\alpha_a(x_a, y) + \alpha I^\alpha_b(x_a, y) \quad (17)$$

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

Let’s compare the methods explained in Section III and IV with others. Parameters in the implementation are denoted in Table I. Here, $\lambda$, $K_D$ and $K_V$ are for BP algorithm that is described in Section II. We referred to Felzenszwalb’s paper [17] for $\lambda$, $K_D$ and $K_V$. Also, $T_h$ is the threshold 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.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K_h$</th>
<th>$K_V$</th>
<th>$T_h$</th>
<th>$t_1$</th>
<th>$t_2$</th>
<th>$T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.07</td>
<td>15</td>
<td>1.7</td>
<td>2</td>
<td>20</td>
<td>25</td>
<td>60</td>
</tr>
</tbody>
</table>

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 have 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 $\alpha = 0.25, 0.5, 0.75$.

5.1 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, $\alpha = 0.5$. Figure 16 shows the results. The first row images are the ground truth images.

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.
5.2 Cones Experiments

For the qualitative evaluations, we used the left and right images of Cones images as input image, and generated intermediate image at center with $\alpha = 0.5$. Figure 17 include the results for Cones.

The first row images are the ground truth images. 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 and the third rows are the results from Jin, and Ours. Notice that in the second and third column, the patches around the cone tips are apparent in the other results and are invisible in our result. In the fourth column, the spurious thin edges in other results are also not clearly removed in our result. We can observe that our method outperformed in the sharpen areas. The shapes of cone tips and chopsticks in our proposed method result are clearer than the other method.

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.
For the quantitative evaluations, we used the left and right images from *Cones*, *Teddy*, and *Venus* images, and generated intermediate views $\alpha = 0.25$, $\alpha = 0.5$ 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 2. 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 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.

Table 2. PSNR of the synthetic images by Jin, Wang, and ours

<table>
<thead>
<tr>
<th></th>
<th>Cones</th>
<th>Teddy</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jin</td>
<td>27.38 dB</td>
<td>29.33 dB</td>
<td>31.36 dB</td>
</tr>
<tr>
<td>Wang</td>
<td>27.69 dB</td>
<td>30.37 dB</td>
<td>31.50 dB</td>
</tr>
<tr>
<td>Ours</td>
<td>28.01 dB</td>
<td>30.44 dB</td>
<td>32.88 dB</td>
</tr>
</tbody>
</table>

6. 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.
7. References


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)