# Intermediate Scene Generation using Fast Bidirectional Disparity Morphing and Three Occluding Patterns*

Dae-Hyun Kim and Jong-Soo Choi
Dept. of Image Eng., Graduate School of Advanced Imaging Science, Multimedia, Film, Chung-Ang Univ.
221 Huksuk-dong Dongjak-ku, Seolul 156-756, Korea
Tel. +82-2-820-5295, Fax.: +82-2-814-5404
e-mail : {vante77, jschoi}@imagelab.cau.ac.kr

**Abstract:** In this paper, we describe an algorithm to automatically generate an intermediate scene using the bidirectional disparity morphing from the parallel stereopair. To compute the disparity between two reference images, we use the 2-step fast block matching algorithm that restricts the searching range and accelerates the speed of the computation of the disparity. We also define three occluding patterns so as to smooth the computed disparities, especially for occluded regions. They are derived from the peculiar properties of the disparity map. The smoothed disparity maps present that the false disparities are well corrected and the boundary between foreground and background becomes sharper. We discuss the advantages of this algorithm compared to the commonly used schemes and we show some experimental results with real data.

## 1. Introduction

There has been increased interest both for computer vision and graphics in image-based rendering (IBR) methods, which deal with how to produce an scene from an arbitrary viewpoint, given a set of reference images. So far, the following two methods, such as *3D model based method* and *2D image based method*, have been invented to realize the view interpolation [1,2,5,6].

In this paper, we present an to automatically generate intermediate scenes using the bidirectional disparity morphing (BDM) from the parallel stereopair. Because it needs a dense disparity map, it is expensive for computing a disparity map and false disparity caused by occlusion makes low the quality of the intermediate scene.

In order to speed up the computation of the disparity, 2-step fast block matching algorithm (BMA) is proposed. This method, at first step, roughly determines the expected candidate and then closely investigates the corresponding block within a restricted region. In this procedure, an occlusion is detectable because the estimation error of the occluded block is larger than that of the visible block.

The distribution of the disparity is generally uniform because the disparity is continuous and similar in the same layer. From these properties, three occluding patterns are able to be defined and used for the median filter. The filtered disparity around the occluding boundary is remarkably refined and the occluding boundary becomes sharper.

Once the disparity map from each left and right image is computed, synthesis is performed separately from both left and right images, and then the individual intermediate scenes are dissolved into the final synthesized scene to minimize the effect of disparity estimation errors[1].

The paper is organized as follows. Section 2 describes the BDM for computing an intermediate coordinate and proposes how to fill the holes. Section 3 defines 2-step fast BMA and computes a disparity map while detecting the occluding regions. Then we define the three kinds of the occluding patterns used to correct the false disparities. Section 4 shows the experimental results and evaluates the proposed algorithm in terms of the performance. Finally section 5 makes the conclusions.

## 2. Bidirectional Disparity Morphing (BDM)

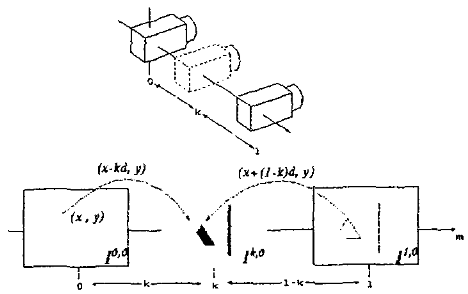



Figure 1. parallel camera configuration.

We assume that the cameras were separated by a horizontal distance and arranged in the parallel camera configuration as depicted in figure 1. A point in the scene generates corresponding points $p_l$ and $p_r$ in the left and right images, respectively. If the scene point is visible in both images, the disparity is defined as the distance, in pixels, between the corresponding points. Due to the parallel camera setup, the disparity is only in the horizontal direction and the relationship is given by

$$p_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_l + d_{L\to R}(x_l, y_l) \\ y_l \end{bmatrix} = p_l + \begin{bmatrix} d_{L\to R}(x_l, y_l) \\ 0 \end{bmatrix} \quad (1)$$

where $d_{L\to R}(x_l, y_l)$ is the disparity from the left pixel to the corresponding right pixel.

The distance between the left and right images is normalized to one. The desired intermediate scene is then parameterized by morphing coefficient($s$) where $0 < s < 1$. The mapping from the given left image to the position in the intermediate scene is a function of disparity as shown by the linear relationship,

---

*This research was supported by the Ministry of Education, Seoul, Korea, under the BK21 project, and by the Ministry of Science and Technology, Seoul, Korea, under the NRL project (2000-N-NL-01-C-285).

$$p_s = \begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} x_l + s \cdot d_{L\to R}(x_l, y_l) \\ y_l \end{bmatrix} \quad (2)$$

Equation (2), however, does not model the effects that changes in visibility have on image content[8]. From the standpoint of morphing, change in visibility results in two types of conditions : *folds* and *holes* [7,8,12].

*Folds* can be easily resolved by using Z-buffering algorithm because the disparity is inversely proportional to depth so that Z-buffer algorithm may be directly applied with inverse disparity substituted for depth.

Unlike *folds*, *holes* cannot always be eliminated by using image information alone. Even though it is prevalent in existing image interpolation[8,4], the neighborhood interpolation approach has a limit for larger disparity scenes. In this paper, therefore, we first generate the intermediate scenes of both left and right images using BDM, and then these two intermediate scenes are used to fill the opposite image's holes because the holes in one's intermediate scene correspond to the visible regions in the opposite image. As a result, we can improve the visible realism of the intermediate scenes.

Synthesis is performed separately from both left and right images, and then the each intermediate scene is dissolved into the final synthesized scene. This seemingly redundant process allows us to minimize the effect of disparity estimation errors[1]. The relationship is given by

$$IP_s(x,y) = \begin{cases} w_{L\to R} \cdot IP_{s,L\to R}(x,y) + w_{R\to L} \cdot IP_{s,R\to L}(x,y) & \text{if } IP_{s,L\to R} \neq 0,\ IP_{s,R\to L} \neq 0 \\ IP_{s,L\to R}(x,y) & \text{if } IP_{s,L\to R} \neq 0,\ IP_{s,R\to L} = 0 \\ IP_{s,R\to L}(x,y) & \text{if } IP_{s,L\to R} = 0,\ IP_{s,R\to L} \neq 0 \\ \text{Nearest neighbor pixel} & \text{if } IP_{s,L\to R} = 0,\ IP_{s,R\to L} = 0 \end{cases} \quad (3)$$

where $IP_s(x,y)$ is a dissolved intensity value at $(x,y)$, and $IP_{s,L\to R}(x,y)$ and $IP_{s,R\to L}(x,y)$ are an interpolated intensity values from both left and right image, respectively. If $IP_{s,L\to R}(x,y)$ and $IP_{s,R\to L}(x,y)$ are not zero, the intensity value is determined by the weighted average of the intensity values at the corresponding points. The weighted average gives a priority to the information from the nearer viewing position[2]. This enables the interpolated image to represent the scene with highlight parts whose observed intensity changes depending on the viewing position. On the contrast, if all intensity values are zero, we use the prevalent neighborhood interpolation approach because it rarely exists.

## 3. Computation of Disparity Map

### 3.1 Definition of 2-step fast block matching algorithm

*A) Restriction of searching range*

Because we assume the parallel camera setup as in [3], we have $y = y_L = y_R$ and $x_L \geq x_R$. This constraint can restrict the range to the half-direction.

*B) Definition of 2-step fast block matching algorithm*

We propose a 2-step search procedure, which is modified the 3-step search procedure. As shown in figure 2, we first search roughly the expected candidate block within all searching range and then closely investigate the corresponding block within the restricted searching range.

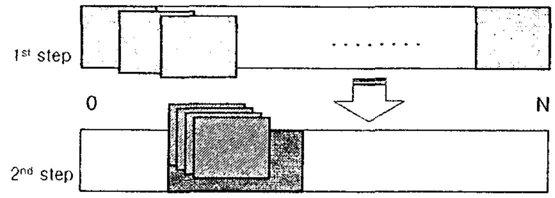



Figure 2. 2-step fast block matching algorithm

*C) Detecting an occluding region*

A block which different layers are mutually overlapped, in general, presents larger estimation error compared to other blocks and it has a higher probability that the occlusion occurs. Therefore we deem the 8×8 block to be an occlusion if its estimation error is larger than a threshold.

### 3.2 Three occluding patterns

Typically we have used the 3×3 median filter so as to smooth the computed disparity map.

| | | |
|---|---|---|
| $m_1$ | $m_2$ | $m_3$ |
| $m_4$ | $\mathbf{m_0}$ | $m_5$ |
| $m_6$ | $m_7$ | $m_8$ |

(a)

| | | |
|---|---|---|
| $m_1$ | $m_2$ | $m_3$ |
| $m_4$ | $m_5$ | $\mathbf{m_0}$ |
| $m_6$ | $m_7$ | $m_8$ |

(b)

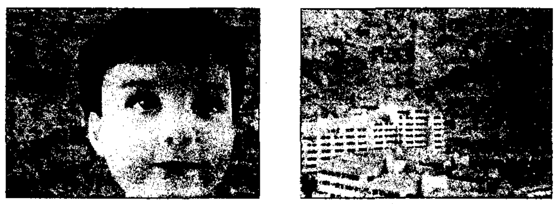
| | | |
|---|---|---|
| $m_1$ | $m_2$ | $m_3$ |
| $\mathbf{m_0}$ | $m_4$ | $m_5$ |
| $m_6$ | $m_7$ | $m_8$ |

(c)

Figure 3. Three occluding patterns.

The boundary of the occlusion intends to be vertical because of the parallel camera setup. The occlusion of the left image is occurred when either the foreground moves to left or the camera moves to right. And the boundary of the occlusion is seen in the left side of the foreground. That is, the disparity of the occlusion is similar to the disparity of the left layer of the occluding boundary. The occlusion of the right image represents the opposite properties as well. From these properties, we can define three kinds of the occluding patterns. Figure 3(a) presents the conventional median mask for both visible blocks. Figure 3(b) and (c) present the left and right occluding patterns, respectively. In that figure, the grayed $m_0$ presents the central pixel of the support of the filter and it is located on the boundary of the occlusion.

## 4. Experimental results



(a) Kid (b) City

Figure 4. Experimented image sequences

We make an experiment on the proposed algorithm with two image sequences shown in figure 4. The experimented images are 640×480 RGB color image sequence, which are acquired from the 1×8 camera arrays in the university of Tsukuba in Japan.

8×8 block is used for computing a disparity and the overall searching range is $0 \le d_x \le 127$, and local searching range is $-7 \le d_x \le 7$. In the first step, a block is moved by 4 pixels to determine the candidate block and then closely investigated with full search procedure.

### 4.1 Detecting an occluded block

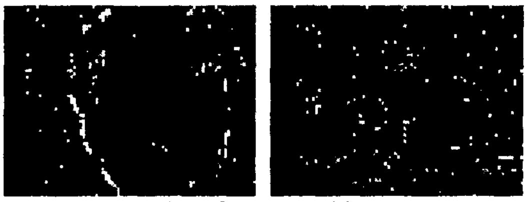
We use the mean absolute difference (MAD) of Eq.(4) as a matching criteria and determine the corresponding block which MAD is minimum. However, it is not reasonable to simply select the minimum erroneous block as a correspondence since the occluded block doesn't have any correspondence.

$$E(d_x) = \sum_{(x,y)\in B} \left| I_m(x,y) - I_{m+1}(x+d_x,y) \right| \qquad (4)$$
$$\left[\hat{d}_x\right] = \arg\min_{(d_x)}[E(d_x)]$$

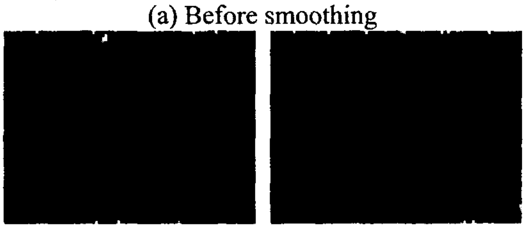
We set the threshold up 10% of the maximum estimation error through the repeated experiments and decide whether it is a correspondence or not. In other words, if the estimation error were smaller than the threshold it would be considered of a correct correspondence, otherwise it would be an occlusion.

### 4.2 Smoothing with three occluding patterns

We smooth the disparity maps with three occluding patterns defined in section 3.2 in order to improve the reliability of the computed disparities. The block that is visible in the both left and right images is filtered by the general median mask like figure 3(a). The block that is occluded in either left or right image is smoothed by the proposed median filters such as figure 3(b) and (c), respectively. Figure 5 presents the disparity maps of figure 4 before and after the smoothing is performed. As shown in figure 5, the boundary between the foreground and background becomes sharper and the false disparities are well corrected.



(a) Before smoothing



(b) After smoothing

Figure 5. Disparity maps before and after smoothing

### 4.3 Synthesis

Once we compute the disparity maps from both left and right images, we can interpolate the intermediate scenes by using BDM. Then two interpolated scenes are dissolved to minimize the effect of the disparity estimation errors. Figure 6 presents the examples of the intermediate scenes for various morphing coefficients( $s$ ) and it is also visibly satisfactory.

### 4.4 Performance evaluation

We use the peak signal noise ratio (PSNR) of Eq.(5) so as to evaluate the proposed algorithm in terms of the performance.

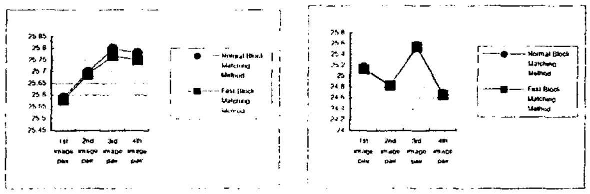
$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I(x,y) - I'(x,y)]^2 \qquad (5)$$
$$PSNR = 20 \times \log 10\left(255/\sqrt{MSE}\right)$$

where $M$ , $N$ present the width and height of the given image, $I(\cdot)$ and $I'(\cdot)$ are the original image and the interpolated image, respectively.

Table 1. Profile for computation of disparity map

| Methods / Time | Typical BMA | 2-step fast BMA |
|---|---|---|
| msec | 955.574 | 511.634 |

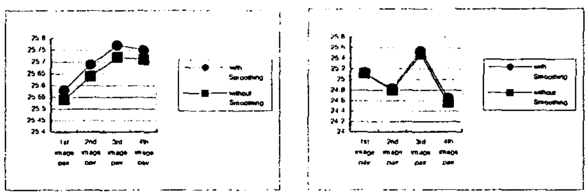
The proposed 2-step fast BMA is compared to both the pixel based method and the normal BMA by using the PSNR. In our experimentation, the pixel based method is averagely 2~3dB higher, but it is very expensive for computing the disparity map. On the contrary, the block based methods (including both the normal BMA and the proposed method) seriously decrease the computational costs, but on the other hand, they are rather inferior in quality of the interpolated scene.



(a) Kid (b) City

Figure 7. Performance for various morphing coefficients

Table 1 compares the amount of the computation via the PC profile, and figure 7 illustrates the individual performance before and after applying the fast algorithm. As a result, the proposed 2-step fast BMA cannot only be even cheaper about 53%, but also preserve its performance.



(a) Kid (b) City

Figure 8. Performance for three occluding patterns

Figure 8 compares the results of whether we apply the median filter with three occluding patterns or not. This also presents that the proposed algorithm is averagely 0.05dB higher. Even though they don't improve the overall quality of the images, three occluding patterns can remarkably refine the intermediate images around the occluding boundary and make sharper the occluding boundary.

## 5. Conclusion

In this paper, we proposed a fast algorithm to automatically generate an intermediate scene using the bidirectional disparity morphing (BDM) and three occluding patterns from the parallel stereopair. In addition, we proposed the 2-step fast BMA, which is able to restrict the searching range and accelerate the speed of the computation of the disparity thanks to the parallel camera setup. As a result, we can decrease the amount of the computation about 53%.

Because the occluded regions usually present larger estimation error than those of both visible regions, we can separate the occluded region and the visible region. We set the threshold up 10% of the maximum estimation error, and decide whether the BMA succeeds or not.

We defined three occluding patterns to smooth the computed disparity map. These patterns vary the location of the central pixel of the median mask considering both the direction and the boundary of the occlusion as the camera moves. As a result, we can well correct the false disparities and obtain the sharper boundary of the occluded region.

The result of the evaluation of the proposed algorithm presents that 2-step fast BMA lowers the cost with keeping up the previous performance, and the smoothing with three occluding patterns slightly increases overall PSNR of the interpolated images as it makes the boundary between the foreground and the background sharper.

## Reference


[1] J. S. McVeigh, M. W. Siegel, and A. G. Jordan, "Intermediate view synthesis considering occluded and ambiguously referenced image regions," *Signal Processing: Image Communication*, pp. 21 - 28 , Vol. 9, 1996.

[2] K. Satoh, I. Kitahara, Y. Ohta, ``3D Image Display with Motion Parallax by Camera Matrix Stereo," *Proc. of Third Int. Conf. on Multimedia Computing and Systems (ICMCS'96)*, 1996.

[3] Redert, E. Hendriks, J. Biemond, "3D Scene Reconstruction with Viewpoint Adaptation on Stereo Displays," *IEEE Trans. On Circuits and Systems for Video Technology*, pp.550-562, Vol.10, No.4, June 2000.

[4] K. Satoh, Y. Ohta, "Occlusion Detectable Stereo Using A Camera Matrix," *Proceedings Second Asian Conference on Computer Vision (ACCV'95)*, pp.331-335, Vol.2, 1995.

[5] Takeo Kanade, P. J. Narayanan, P. W. Rander, "Virtualized reality: concepts and early results," *Proceedings IEEE Workshop on Representation of Visual Scenes*, pp.69-76, 1995.

[6] Hideo Saito, S. Baba, M. Kimura, S. Vedula, T. Kanade, "Appearance-based virtual view generation of temporally-varying events from multi-camera images in the 3D room," *Proc. of Second Int. Conf. on 3-D Digital Imaging and Modeling*, pp.516–525, 1999.

[7] S. M. Seitz, C. R. Dyer, "View Morphing," *Proc. Of SIGGRAPH'96*, pp.21-30, 1996.

[8] S. E. Chen, L. Williams, "View Interpolation for Image Synthesis," *In Proc. SIGGRAPH 93*, pp.279-288, 1993.

[9] C. L. Zitnicj, T. Kanade, "A Cooperative Algorithm for Stereo Matching and Occlusion Detection," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, pp.675-684, Vol.22, No.7, July 2000.

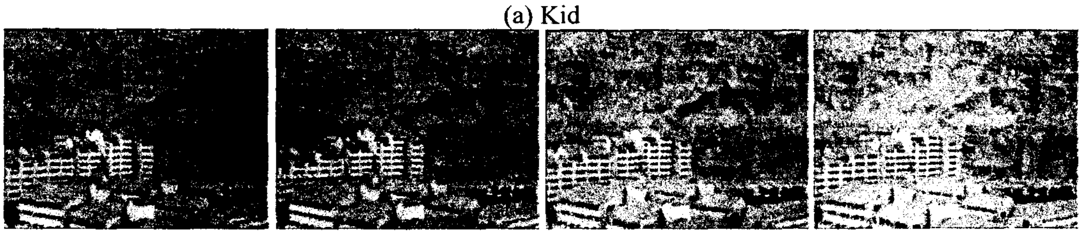
[10] J. R. Ohm, E. Izquierdo M., "An Object-Based System for Stereoscopic Viewpoint Synthesis," *IEEE Trans. On Circuits and Systems for Video Technology*, pp.801-811, Vol.5, No.5, Oct. 2000.

[11] Y. Sugaya, Y. Ohta, "Stereo by integration of two algorithms with/without occlusion handling," *Proc. 15th International Conference on Pattern Recognition(ICPR2000)*, Vol.1, pp.109-113, Sep. 2000.

[12] S. M. Seitz, C. R. Dyer, "Physically-valid view synthesis by image interpolation," *In Proc. IEEE Workshop on Representation of Visual Scenes*, pp.18-25, 1995.

[13] M. Lhuillier, L. Quan, "Image Interpolation by Joint View Triangulation," *Proc. Of Computer Vision and Pattern Recognition(CVPR'95)*, 1995.


(a) Kid



(b) City

Figure 6. Intermediated scenes for various morphing coefficient $s$ =0.2, 0.4, 0.6, 0.8