Probability-Based Rendering for View Synthesis

Bumsub Ham, *Member, IEEE*, Dongbo Min, *Member, IEEE*, Changjae Oh, *Student Member, IEEE*,
Minh N. Do, *Fellow, IEEE*, and Kwanghoon Sohn, *Senior Member, IEEE*

Abstract—In this paper, a probability-based rendering (PBR) method is described for reconstructing an intermediate view with a steady-state matching probability (SSMP) density function. Conventionally, given multiple reference images, the intermediate view is synthesized via the depth image-based rendering technique in which geometric information (e.g., depth) is explicitly leveraged, thus leading to serious rendering artifacts on the synthesized view even with small depth errors. We address this problem by formulating the rendering process as an image fusion in which the textures of all probable matching points are adaptively blended with the SSMP representing the likelihood that points among the input reference images are matched. The PBR hence becomes more robust against depth estimation errors than existing view synthesis approaches. The MP in the steady-state, SSMP, is inferred for each pixel via the random walk with restart (RWR). The RWR always guarantees visually consistent MP, as opposed to conventional optimization schemes (e.g., diffusion or filtering-based approaches), the accuracy of which heavily depends on parameters used. Experimental results demonstrate the superiority of the PBR over the existing view synthesis approaches both qualitatively and quantitatively. Especially, the PBR is effective in suppressing flicker artifacts of virtual video rendering although no temporal aspect is considered. Moreover, it is shown that the depth map itself calculated from our RWR-based method (by simply choosing the most probable matching point) is also comparable with that of the state-of-the-art local stereo matching methods.

Index Terms—Image-based rendering (IBR), view interpolation, correspondence matching, image fusion, random walk with restart (RWR).

I. INTRODUCTION

IN CONTRAST to the conventional TV, three-dimensional (3D) TV aims to provide a user with an immersive 3D perception and interactivity. The viewer can perceive 3D impression throughout a 3D display with or without wearing glasses, and change the viewpoint according to an individual preference. By passing through the development phase, 3D TV has been successfully deployed on commercial markets, and is spotlighted as the core equipment of the next generation broadcasting system. To make 3D TV come into a wide use, however, there are still some challenges to be addressed.

First, most 3D contents have been made without reflecting the characteristics of the human visual system (HVS). On the one hand, a stereo sequence captured with a baseline longer than a pupil distance might have an excessive depth perception beyond the fusion range of the HVS, and incurs a conflict between convergence and accommodation, resulting in a severe eye strain and visual discomfort to viewers. On the other hand, the stereo video with a reduced baseline contains a 3D depth percept lower than the threshold of the just noticeable depth difference (JNDD) [1]. Besides, each individual may have different range of fusing a 3D perception and thus, it is needed to reconstruct virtual sequences according to the user preference for the degree of depth percept [2]. Second, an advanced compression scheme is essential to efficiently utilize a limited bandwidth and data storage. One alternative is to transmit a stream of images and their associated depth information followed by generating virtual views of 3D scenes at the receiver.

To address these challenges, one of the most important technologies is to synthesize intermediate views with two or more images captured at different viewpoints. Many methods have been proposed in the field of image-based rendering (IBR) for visualizing 3D scenes and objects in a realistic way. Depth image-based rendering (DIBR) is one of the most significant technologies in the IBR, and has been under active development for the next generation TV system [3]. In DIBR, the texture corresponding to the virtual view is re-sampled from available reference images by using depth maps [4]. Thus, inaccurate depth information degrades the quality of the synthesized view, especially around depth discontinuities. Though a number of methods have been proposed for mitigating visual artifacts on synthesized views by using pre-/post-processing or distortion metric [5], [6], there are certain limitations due to errors that inherently exist on depth maps (usually determined by hard decision in correspondence matching). When it comes to video rendering, such artifacts become even more serious due to annoying flickers.

This paper presents a novel method that effectively handles such rendering artifacts from erroneous depth data by formulating the rendering process as a probabilistic fusion framework. In the next section, we describe the motivation

of this work in more details, including related work of the DIBR approaches. Recent research findings of correspondence matching are also presented, since the rendering quality largely depends on the accuracy of estimated depth maps. A preliminary version of this paper can be found at [7].

II. RELATED WORKS AND MOTIVATION

A. Correspondence Matching

Correspondence estimation between/among images is an important and fundamental technique in a great variety of computer vision applications. There are mainly two classes (global methods and local methods) in correspondence matching [8].

Global methods find matching points between images by reasoning a globally optimal solution of an energy functional where the smoothness assumption is explicitly leveraged. Many global optimization schemes have been adapted for efficiently solving this NP-hard problem, e.g., belief propagation [9]. Local methods compute correlation between points with an assumption that all pixels in the matching window have similar disparities. Among many local approaches, one of the seminal works is the adaptive weight-based aggregation method, proposed by Yoon and Kweon [10]. A support weight at each point is computed according to range and spatial distances between points. The depth accuracy is comparable to that of other global methods, but it is computationally expensive due to the nonlinear weight computation.

Recently, many leading local methods have been proposed, especially focusing on yielding high-quality depth maps very efficiently. Min and Sohn presented a multi-scale approach in both image and cost domains for an efficient and reliable cost aggregation [11]. Rhemann *et al.* applied the guided filter [12], the complexity of which is independent of a matching window size, as an alternative to the bilateral filter in the aggregation step [13]. De-Maeztu *et al.* presented a geodesic diffusion-based approach where weights as well as matching costs are diffused (aggregated) so as to lower the computational cost [14]. While most methods have concentrated on reducing computational redundancies on an image resolution and a window size, Min *et al.* improved both runtime efficiency and accuracy of the stereo matching by compressing a search range in the label space and regularly sampling neighbors used in the matching window [15].

B. View Interpolation

Zhang *et al.* introduced an adaptive technique for interpolating an intermediate view [16]. Each reference view is projected onto the plane of an intermediate view. Then, the intermediate view was rendered by the weighted average of these projected images. Min *et al.* synthesized a virtual view by adapting a reverse warping instead of a forward warping, which enables texture information to be sampled in the viewpoint of the virtual camera, such that an appearance of holes could be prevented [17]. Fitzgibbon *et al.* reformulated the view synthesis problem as reconstructing the texture corresponding to the virtual view, not a depth map [18]. Similar to this, Mahajan *et al.* interpolated a virtual view based on the idea that the given pixel to be synthesized in

the virtual view traces out the path in reference images [19]. These methods are inherently based on a discrete formulation, so holes inevitably occur which should be filled to ensure high quality views. Recently, inspired by the image retargeting technique, Lang *et al.* presented a warping-based rendering scheme [20]. It is based on a continuous formulation similar to the mesh-based rendering, thus free from a hole filling problem. However, such warping process leads to geometric distortion around man-made structures, e.g, lines or circles.

There have been many studies on analyzing and alleviating the influence of erroneous depth information on view interpolation. Yang *et al.* constrained the reliability of each pixel in the virtual view via the maximum likelihood, so that the influence of inaccurate depth data could be relaxed [6]. Nguyen and Do quantitatively analyzed a rendering quality and derived error bounds by considering several factors such as depth errors and the number of actual cameras [21]. Takahashi quantitatively analyzed the effect of depth errors, leading to an optimized view interpolation scheme from the perspective of the mean-squared error metric [22]. Similarly, Zhao *et al.* developed the depth no-synthesis-error (D-NOSE) model to examine allowable depth distortions without incurring any geometry distortion in the virtual view [2]. Kunita *et al.* introduced the layered probability maps [23] for dealing with depth ambiguities in the view interpolation. Although the similar concept is used in the rendering process, this method explicitly leverages the geometric prior such as a camera position in order to avoid addressing an occlusion problem. More specifically, when densely sampled color images (more than 2) are given as inputs, intermediate view synthesis results corresponding to a virtual camera are generated with the input images in a probabilistic manner, and are then adaptively combined using reliability values defined by a baseline distance between real and virtual cameras. This fusion approach based on the geometric prior is not suitable to view synthesis using only two images with a relatively larger baseline.

C. Problem Statement and Overview

Given two reference images, our objective is to synthesize a virtual view without deterministic correspondences. By considering the rendering process as a probabilistic re-sampling problem, we re-formulate it as an image fusion process. In this context, the following problems should be considered. *First, how can the probability of all probable matching points be constructed? Second, how can an intermediate view be rendered using a set of all matching candidate points with the probability?* To address these challenges, we present the probability-based rendering (PBR) approach that robustly reconstructs an intermediate view with the steady-state matching probability (SSMP) density function.

1) SSMP: In our work, the matching cost, typically referred to as a cost volume in the correspondence matching literature, is re-defined as the probability of being matched between points [15], enabling the random walk with restart (RWR) to be applied to optimize the matching probability. The RWR uses edge weights between neighboring pixels to enhance

the matching probability similar to aggregation methods for local stereo matching. In addition, it gives the meaningful steady-state distribution of matching probability for each pixel, so visually consistent results are guaranteed by employing small number of adjacent neighbors only. It is worthy of noting that if needed, a depth map can be simply obtained by selecting a disparity label with a maximum matching probability. The depth map shows a comparable performance to that of the state-of-the-art local stereo matching methods.

2) PBR: The rendering process is re-formulated as an image fusion, so that all probable matching points represented by the SSMP can be considered together. The PBR has the following advantages over the aforementioned DIBR methods. First, it is more robust to depth distortions in that an intermediate view is synthesized using a reliability (probability) based fusion by the SSMP, not an explicit depth map which may contain matching errors. Instead of simply calculating the most probable matching point as a disparity value (hard decision) and subsequently generating the virtual view with a single pair of corresponding points only, we blend a set of the textures of all the matching points (soft decision) according to their probabilities (reliability). Moreover, it enables the PBR to be effective in suppressing the flicker artifacts of virtual video. Second, the intermediate view is free from a hole filling problem since the SSMP considers all positions of probable matching points. That is, the appearance of holes is inherently prevented although the view interpolation is formulated in a discrete sense. Third, it is not needed to explicitly handle occlusion regions. Instead, the PBR handles occlusion areas in a probabilistic manner, similar to the hole filling. Although it does not give ground truth textures at occluded pixels, the rendered results are visually more coherent on temporal aspects than those of the DIBR method.

The remainder of this paper is organized as follows. Section III describes new approach of inferring the SSMP. Then, Section IV present our PBR method using the SSMP. An extensive analysis of experimental results is presented in Section V. Finally, conclusion and suggestions for future works are given in Section VI.

III. SSMP WITH RWR

A. Matching Probability

Let $I_l(\mathbf{m}) : \Omega \to \mathbb{R}^3$ and $I_r(\mathbf{m}) : \Omega \to \mathbb{R}^3$ be left and right images defined on a discrete image domain where $\Omega \subset \mathbb{N}^2$ is an open and bounded space with $\mathbf{m} = (m_1, m_2) \in \Omega$ being spatial coordinates. We also denote $p_l^t(\mathbf{m}, d)$ and $p_r^t(\mathbf{m}, d)$ as the matching probability (MP) density functions of the left and right images, respectively, where t is time and d is a disparity hypothesis within a predefined search range. Without loss of generality, let us assume that two reference images are rectified, such that the probability $p_l^t(\mathbf{m}, d)$ measures how likely $I_l(m_1, m_2)$ is to be matched to $I_r(m_1 - d, m_2)$ and $p_r^t(\mathbf{m}, d)$ does how likely $I_r(m_1, m_2)$ to be matched to $I_l(m_1 + d, m_2)$. From here on, we will describe the case of $p_l^t(\mathbf{m}, d)$ only, and denote it as $p^t(\mathbf{m}, d)$ unless otherwise specified.

In general, the probability is inversely proportional to the cost and thus, an initial MP $p^0(\mathbf{m}, d)$ can be calculated with an initial matching cost $e^0(\mathbf{m}, d)$ as follows.

$$p^0(\mathbf{m}, d) = \frac{1}{Z(\mathbf{m})} \exp\left\{ -v e^0(\mathbf{m}, d) \right\}, \tag{1}$$

$$Z(\mathbf{m}) = \sum_d \exp\left\{ -v e^0(\mathbf{m}, d) \right\}, \tag{2}$$

where a positive constant v controls the shape of the MP, and $Z(\mathbf{m})$ represents a normalization term. Various metrics such as the l_p norm-based and gradient-based measure [13] can be utilized to calculate $e^0(\mathbf{m}, d)$.

B. Inferring Matching Probability With RWR

In this section, we estimate a SSMP using the RWR with the initial one given in the previous section. The random walk (RW) has been widely used to optimize probabilistic problems [24], [25]. It has been known that the RW and the Laplacian equation give the same solution [24], meaning that the steady-state can be captured by the RW although it gives trivial solutions, i.e., the steady-state solution of the RW is a constant signal. Recently, the RWR has become increasingly popular, since its restarting term gives the meaningful information in the steady-state, allowing the global relation to be considered at all scales [26], [27]. Therefore, the MP inferred with the RWR gives the meaningful solution in the steady-state in contrast to the conventional methods based on the RW, as will be described in Section III-C.

A random walker iteratively transits to its neighboring points according to an edge weight. Also, the random walker goes back to the initial position with a restarting probability α $(0 \le \alpha \le 1)$ at each iteration. A MP in the steady-state, SSMP, can be obtained by the RWR in an iterative fashion as follows:

$$p^{t+1}(\mathbf{m}, d) = (1 - \alpha) \frac{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n}) p^t(\mathbf{n}, d)}{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n})} + \alpha p^0(\mathbf{m}, d), \tag{3}$$

where $\mathcal{N}_m$ denotes the 4-neighborhood of a reference pixel $\mathbf{m}$. Note that (3) becomes the RW when the restarting probability is 0. With an assumption that neighboring pixels tend to have similar MP when the range distance between the reference pixel $\mathbf{m}$ and its neighboring pixel $\mathbf{n}$ is small, an edge weight $w(\mathbf{m}, \mathbf{n})$ is computed as follows:

$$w(\mathbf{m}, \mathbf{n}) = \exp\left(-\frac{\| I_l(\mathbf{m}) - I_l(\mathbf{n}) \|_2^2}{\gamma} \right), \tag{4}$$

where γ represents the bandwidth parameter, typically set to the intensity variance, and $\| \cdot \|_2$ denotes l_2 norm. Then, a steady-state solution $p^s(\mathbf{m}, d)$, referred as the SSMP in this literature, can be obtained by iteratively updating (3) until $p^{t+1}(\mathbf{m}, d) = p^t(\mathbf{m}, d)$.

The significance of our work is as follows. First, only the small number of adjacent neighbors (here, 4-neighborhood) is needed to infer the MP, so the method does not require specifying a window size for reliable matching as opposed to the conventional aggregation methods. Second, our method is free from specifying the number of iteration for meaningful

solution, since it gives a nontrivial solution in the steady-state. In contrast, the conventional methods give trivial steady-state solutions, thus specifying a proper window size and/or the number of iterations is very crucial to the performance. In the following section, these limitations of the conventional methods will be analyzed. Third, our method gives the optimal solution for given energy functional, and it is also a non-trivial SSMP, which will be described in detail in Section III-D. As shown in recent studies [13], an edge-aware cost volume filtering is an excellent alternative to existing discrete labeling techniques such as belief propagation [9]. Along this perspective, the RWR-based method also has very desirable properties (e.g., the number of parameters and optimality) in diverse discrete pixel-labeling problems.

C. Limitations of Conventional Cost Aggregation in Stereo Matching

The probability inference in (3) is conceptually related to the cost aggregation commonly used in correspondence matching, e.g., the probability inference with the RWR in (3) becomes the adaptive weight method [10] by setting $\alpha = 0$ and substituting the MP with the matching cost.

Specifically, let us consider an energy functional of the RW as follows [24]:

$$E_{RW}(p(\mathbf{m}, d))$$
$$= \frac{1}{2} \sum_m \sum_{n \in \mathcal{N}_m} (p(\mathbf{m}, d) - p(\mathbf{n}, d))^2 w(\mathbf{m}, \mathbf{n}) \quad (5)$$

Since this energy functional is linear and strictly convex, a global minimum is guaranteed. It has two types of solution, each of which can be obtained via the steepest descent method and the Gauss–Jacobi iteration, respectively.

1) Flow Solution: Since the derivative of the energy functional is

$$\frac{\partial E_{RW}(p)}{\partial p} = \sum_{\mathbf{n} \in \mathcal{N}_m} (p(\mathbf{m}, d) - p(\mathbf{n}, d)) w(\mathbf{m}, \mathbf{n}), \quad (6)$$

the flow solution with an initial condition $p^0(\mathbf{m}, d)$ is obtained using the steepest descent method after approximating a partial derivative w.r.t time via the forward difference, as follows:

$$p^{t+1}(\mathbf{m}, d)$$
$$= p^t(\mathbf{m}, d) + \Delta t \sum_{\mathbf{n} \in \mathcal{N}_m} (p^t(\mathbf{n}, d) - p^t(\mathbf{m}, d)) w(\mathbf{m}, \mathbf{n}), \quad (7)$$

where Δt is an evolution step size. Interestingly, it is very similar to the cost aggregation method of Min and Sohn [11] by substituting the MP with the matching cost. Since (7) can be seen as a type of diffusion, e.g., anisotropic diffusion [30], the results of the probability inference or the cost aggregation has no meaningful solution in an asymptotic state. Therefore, the number of iterations t and an evolution step size Δt should be carefully tuned for a proper regularization.

2) Steady-State Solution: The steady-state solution is given by

$$p^{t+1}(\mathbf{m}, d) = \frac{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n}) p^t(\mathbf{n}, d)}{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n})}, \quad (8)$$

which corresponds to the adaptive weight approach [10]. It indicates that inferring a probability via the RW is the same as the procedure of the adaptive weight method, which means the adaptive weight method does not provide a meaningful steady-state solution. This limitation can be remedied by adjusting scale parameters, i.e., by restricting the number of iterations and/or by enlarging a window size for considering as many paths between two points as possible. It is, however, not trivial to specify these parameters: there exist no theoretical backgrounds to justify this.

In summary, the existing cost aggregation approaches share the same origin with the RW, so there are several parameters (e.g. iteration numbers and window size) to be carefully tuned, which influence the computational complexity as well as the accuracy. In contrast, in our method, after weight $w(\mathbf{m}, \mathbf{n})$ is calculated using the bandwidth parameter γ as in other aggregation methods, the accuracy depends on only the restarting probability, α (see Fig. 5). It does not affect the computational complexity. Furthermore, using 4-neighborhood is sufficient in considering all paths and scales between points in an asymptotic state without compromising accuracy.

D. Properties of the Solution (SSMP)

1) Non-Trivial Steady-State Solution: The probability inference with the RWR can be represented via the vector notation. Let us denote a 2D slice corresponding to d^{th} label at the MP $p^t(\mathbf{m}, d)$ as a $M \times 1$ vector $\mathbf{P}_d^t$ where M is an image size. Then, (3) is represented as follows:

$$\mathbf{P}_d^{t+1} = (1 - \alpha)\mathbf{D}^{-1}\mathbf{W}\mathbf{P}_d^t + \alpha\mathbf{P}_d^0 = (1 - \alpha)\bar{\mathbf{W}}\mathbf{P}_d^t + \alpha\mathbf{P}_d^0. \quad (9)$$

The adjacency matrix $\mathbf{W} = [w(\mathbf{m}, \mathbf{n})]_{M \times M}$ is normalized as $\bar{\mathbf{W}} = \mathbf{D}^{-1}\mathbf{W}$, where $\mathbf{D} = diag(D_1, \ldots D_M)$ is a diagonal matrix and $D_m = \sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n})$. The steady state solution $\mathbf{P}_d^s$ is obtained as [26]

$$\mathbf{P}_d^s = \alpha[\mathbf{I} - (1 - \alpha)\bar{\mathbf{W}}]^{-1}\mathbf{P}_d^0 = \mathbf{R}\mathbf{P}_d^0, \quad (10)$$

where $\mathbf{I}$ is a $M \times M$ identity matrix. $\mathbf{R}$ can be further decomposed to an infinite geometric series:

$$\mathbf{R} = \alpha[\mathbf{I} - (1 - \alpha)\bar{\mathbf{W}}]^{-1} = \alpha \sum_{t=0}^{t=\infty} (1 - \alpha)^t \bar{\mathbf{W}}^t. \quad (11)$$

That is, $\mathbf{R}$ is a weighted sum of $\bar{\mathbf{W}}^t$, whose element is a probability that a random walker transits $\mathbf{n}$ to $\mathbf{m}$ after t iterations. Thus, the RWR optimizes an initial MP by considering all possible paths between two points.

2) Optimal Solution: Let us consider the following energy functional [28].

$$E_{RWR}(p(\mathbf{m}, d))$$
$$= \frac{1}{2} \sum_m \sum_{n \in \mathcal{N}_m} (p(\mathbf{m}, d) - p(\mathbf{n}, d))^2 \bar{w}(\mathbf{m}, \mathbf{n})$$
$$+ \frac{\lambda}{2} \sum_m (p(\mathbf{m}, d) - p^0(\mathbf{m}, d))^2, \quad (12)$$

where $\lambda > 0$ is a regularization parameter and $\bar{w}(\mathbf{m}, \mathbf{n}) = w(\mathbf{m}, \mathbf{n}) / \sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n})$ is a normalized weight. Note that a particular case of the energy functional E_{RWR}, i.e., $\bar{w}(\mathbf{m}, \mathbf{n}) = w(\mathbf{m}, \mathbf{n})$, was speculated in [29]. Since this energy functional

is convex for the probability $p(\mathbf{m}, d)$, the global minimum of the energy functional can be derived via the steepest descent method as follows:

$$p^{t+1}(\mathbf{m}, d)$$
$$= \frac{1}{1+\lambda} \frac{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n}) p^t(\mathbf{n}, d)}{\sum_{\mathbf{n} \in \mathcal{N}_m} w(\mathbf{m}, \mathbf{n})} + \frac{\lambda}{1+\lambda} p^0(\mathbf{m}, d). \quad (13)$$

It gives the same solution as (3) when $\lambda/(1 + \lambda)$ is equal to the restarting probability α. Therefore, the steady-state solution of (3) guarantees a global minimum of the energy functional of (12) as well as is non-trivial. Note that in the discrete labeling framework, (10) and (13) can be seen as a semi-global solution in that it is a global minimum for the specific d hypothesis only, i.e., an optimization within the same disparity space, not over the disparity space [11].

IV. PBR WITH SSMP

A. PBR With SSMP

Now, the two reference images, $I_l(\mathbf{m})$ and $I_r(\mathbf{m})$, and the sets of their corresponding SSMPs, $p_l^s(\mathbf{m}, d)$ and $p_r^s(\mathbf{m}, d)$ are given. It is worthy of noting that any other MPs can be also applicable to the PBR, e.g., $p_l^t(\mathbf{m}, d)$ and $p_r^t(\mathbf{m}, d)$. We cast the rendering process into the probabilistic image fusion. In [25], a fusion of multi-exposure images was introduced in a way that multiple reference images were sampled and fused with the probabilities. Unlike this work, we consider an image fusion where the reference images are not registered. Therefore, for inferring probabilities, we should consider perturbations, i.e., disparity candidate d, between reference images.

Re-sampled textures: Let us assume that a baseline between the left and right cameras is normalized to 1, and $0 \le \beta \le 1$ denotes the location of a virtual camera. When the perturbation between images is d, $I_l^v(\mathbf{m}, d)$ and $I_r^v(\mathbf{m}, d)$ describe color images re-sampled on the virtual view from the left and right reference images, respectively. That is, $I_l^v(\mathbf{m}, d)$ ($I_r^v(\mathbf{m}, d)$) is a sub-sampled point between $I_l(\lfloor m_1 + \beta d \rfloor, m_2)$ and $I_l(\lceil m_1 + \beta d \rceil, m_2)$ ($I_r(\lfloor m_1 - (1 - \beta) d \rfloor, m_2)$ and $I_r(\lceil m_1 - (1 - \beta) d \rceil, m_2)$), where $\lfloor x \rfloor$ and $\lceil x \rceil$ denote the floor and ceiling functions, respectively. Here, 2D point $\mathbf{m}$ is defined as (m_1, m_2). Note that the view synthesis is performed based on the coordinate of the virtual camera, not the reference camera, to prevent the holes from occurring. Thus, the sampling position of the pixel in the reference image is not an integer, so the pixel is sub-sampled to avoid an aliasing artifact.

Matching probability: $P_l(\mathbf{m}, d)$ and $P_r(\mathbf{m}, d)$ encode the matching probability of $I_l^v(\mathbf{m}, d)$ and $I_r^v(\mathbf{m}, d)$, respectively, as follows:

$$P_l(\mathbf{m}, d) = \frac{1}{Z_l(\mathbf{m})} p_l^s(<m_1 + \beta d>, m_2, d),$$
$$P_r(\mathbf{m}, d) = \frac{1}{Z_r(\mathbf{m})} p_r^s(<m_1 - (1-\beta)d>, m_2, d), \quad (14)$$

where $Z_l(\mathbf{m}) = \sum_d p_l^s(<m_1 + \beta d>, m_2, d)$ and $Z_r(\mathbf{m}) = \sum_d p_r^s(<m_1 - (1-\beta)d>, m_2, d)$. $<\ \cdot\ >$ represents a rounding operator. Note that unlike $I_l^v(\mathbf{m}, d)$ and $I_r^v(\mathbf{m}, d)$, the

matching probability of (14) is not sub-sampled since the sub-sampling in the matching probability alters the properties of the probability density function, whose sum over all variables is always 1.

The virtual view $I_v(\mathbf{m})$ is then synthesized via an image fusion process. Specifically, a probabilistic average, $E_l(I_l(\mathbf{m}))$ and $E_r(I_r(\mathbf{m}))$, for two reference images are computed with corresponding probability, $P_l(\mathbf{m}, d)$ and $P_r(\mathbf{m}, d)$, and the textures, $I_l^v(\mathbf{m}, d)$ and $I_r^v(\mathbf{m}, d)$, *along the disparity hypothesis d* and then blended as follows:

$$I_v(\mathbf{m}) = \beta E_l(I_l(\mathbf{m})) + (1 - \beta) E_r(I_r(\mathbf{m}))$$
$$= \sum_d \beta I_l^v(\mathbf{m}, d) P_l(\mathbf{m}, d) + (1-\beta) I_r^v(\mathbf{m}, d) P_r(\mathbf{m}, d),$$
$$(15)$$

1) Comparison of PBR and DIBR: The PBR is closely related to the conventional DIBR. Let us denote the left and right disparity maps as $d_l^w(\mathbf{m})$ and $d_r^w(\mathbf{m})$, respectively, each of which is warped to the virtual camera from the reference camera. The sampled points $I_l^v(\mathbf{m}, d)$ and $I_r^v(\mathbf{m}, d)$ is then converted as functions of $\mathbf{m}$, $I_l^v(\mathbf{m})$ and $I_r^v(\mathbf{m})$, respectively. That is, $I_l^v(\mathbf{m})$ is a sub-sampled point between $I_l(\lfloor m_1 + \beta d_l^w(m) \rfloor, m_2)$ and $I_l(\lceil m_1 + \beta d_l^w(m) \rceil, m_2)$. $I_r^v(\mathbf{m})$ is similarly defined.

Furthermore, the matching probability functions $P_l(\mathbf{m}, d)$ and $P_r(\mathbf{m}, d)$ on the virtual camera coordinate is simplified as a set of shifted Dirac delta function as follows:

$$P_l(\mathbf{m}, d) = \begin{cases} 1 & d = d_l^w(\mathbf{m}) \\ 0 & d \ne d_l^w(\mathbf{m}), \end{cases}$$

and

$$P_r(\mathbf{m}, d) = \begin{cases} 1 & d = d_r^w(\mathbf{m}) \\ 0 & d \ne d_r^w(\mathbf{m}). \end{cases} \quad (16)$$

Then, the PBR in (15) becomes

$$I_v(\mathbf{m}) = \beta I_l^v(\mathbf{m}) + (1 - \beta) I_r^v(\mathbf{m}), \quad (17)$$

which is identical to the conventional DIBR except that the function encoding the visibility of a pixel in the virtual view is not explicitly used [17]. The PBR decides the pixel visibility in a probabilistic manner instead, i.e., the visibility information is implicitly embedded in $P_l(\mathbf{m}, d)$ and $P_r(\mathbf{m}, d)$ of (14) in a way that a visible pixel is likely to have a higher matching probability than an invisible pixel. We will show in experimental results that such implicit handling method can effectively suppress the rendering artifacts from depth estimation errors.

Fig. 1 compares two approaches, the PBR and the DIBR. Let us suppose that an intermediate view is rendered with the left image only. With (a) the left reference image and (b) the corresponding depth map and matching probability map,[1] the intermediate views ($\beta = 0.5$) were rendered via (c) the DIBR and (d) the PBR. In the DIBR, no post-processing was applied for fairly comparing their naïve results. In order to clearly exhibit error relaxation capability of the PBR, the intermediate views were rendered with a severely degraded

---

[1]Note that the 3D matching probability map was not shown here.

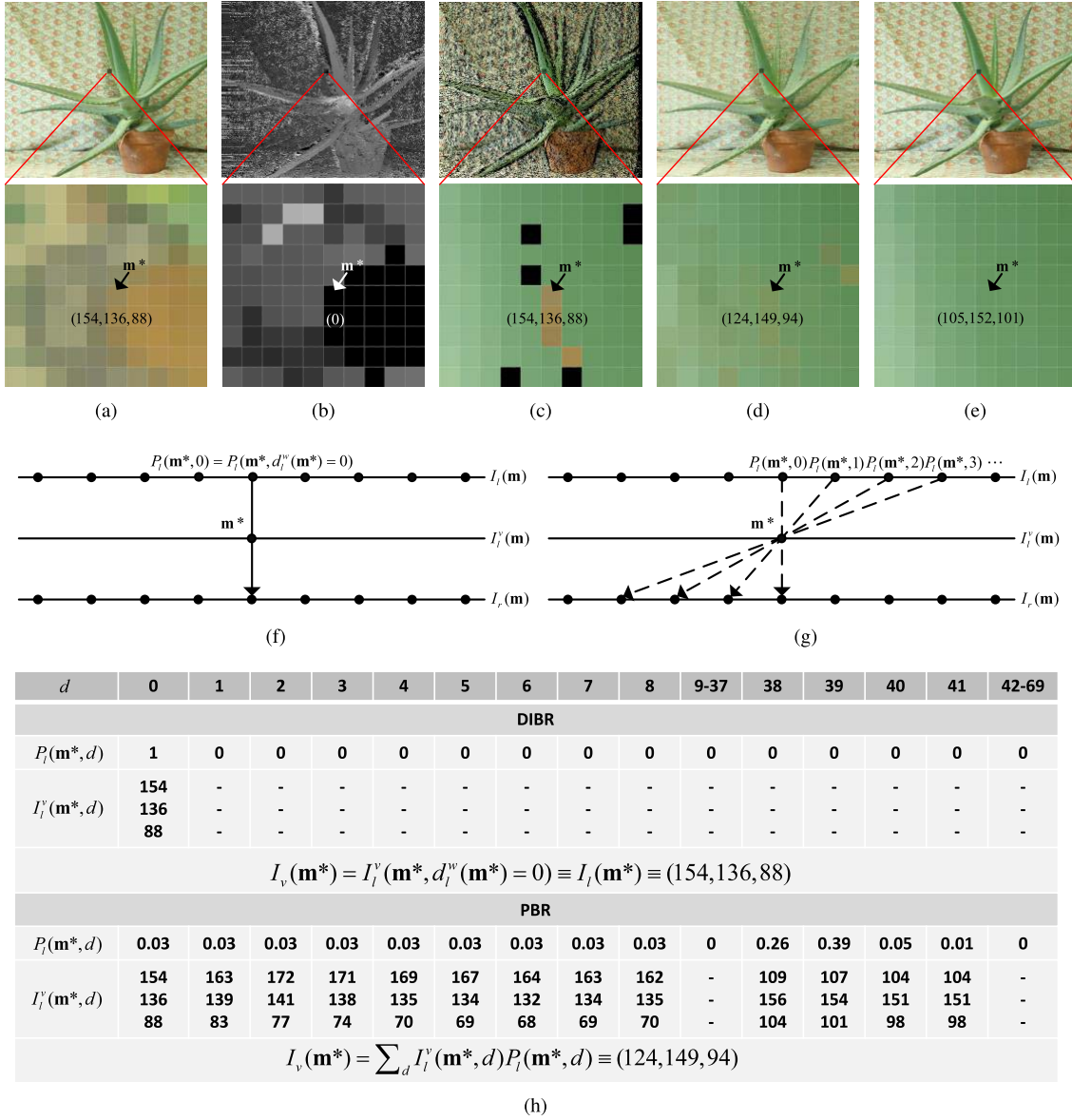| d	0	1	2	3	4	5	6	7	8	9-37	38	39	40	41	42-69
DIBR															
$P_l(\mathbf{m}^*, d)$	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$I_l^v(\mathbf{m}^*, d)$	154	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	136	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	88	-	-	-	-	-	-	-	-	-	-	-	-	-	-
$I_v(\mathbf{m}^*) = I_l^v(\mathbf{m}^*, d_l^w(\mathbf{m}^*) = 0) \equiv I_l(\mathbf{m}^*) \equiv (154, 136, 88)$															
PBR															
$P_l(\mathbf{m}^*, d)$	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0	0.26	0.39	0.05	0.01	0
$I_l^v(\mathbf{m}^*, d)$	154	163	172	171	169	167	164	163	162	-	109	107	104	104	-
	136	139	141	138	135	134	132	134	135	-	156	154	151	151	-
	88	83	77	74	70	69	68	69	70	-	104	101	98	98	-
$I_v(\mathbf{m}^*) = \sum_d I_l^v(\mathbf{m}^*, d) P_l(\mathbf{m}^*, d) \equiv (124, 149, 94)$															

(h)

Fig. 1. A comparison of the PBR and the DIBR: with (a) the left reference image and (b) corresponding depth map and MP, intermediate views ($\beta = 0.5$) were rendered via (c) the DIBR and (d) the PBR, respectively. The per-pixel MP $p_l^0(\mathbf{m}, d)$ was used for rendering. In contrast to the DIBR with (f) the hard decision, the PBR leverages (g) the soft decision, thus (h) the errors incurred by incorrect depth information are dispersed, and the hole regions do not occur.

MP, e.g., per-pixel raw MP $p_l^0(\mathbf{m}, d)$. For fair comparison, the depth map of Fig. 1(b) was obtained via the WTA using $p_l^0(\mathbf{m}, d)$. We could find that the rendered image of the PBR visually outperforms that of the DIBR, even with a severely degraded MP. More specifically, for a given fixed point $\mathbf{m}^*$, the PBR synthesizes the intermediate view $I_v(\mathbf{m}^*)$ with the function of reference view $I_l^v(\mathbf{m}^*, d)$ and the probability $P_l(\mathbf{m}^*, d)$ as follows:

$$I_v(\mathbf{m}^*) = \sum_d I_l^v(\mathbf{m}^*, d) P_l(\mathbf{m}^*, d).$$

In contrast, the DIBR renders the view $I_v(\mathbf{m}^*)$ with the function of reference view and the warped disparity $I_l^v(\mathbf{m}^*, d_l^w(\mathbf{m}^*))$ only:

$$I_v(\mathbf{m}^*) = \sum_d I_l^v(\mathbf{m}^*, d) P_l(\mathbf{m}^*, d)$$
$$= I_l^v(\mathbf{m}^*, d_l^w(\mathbf{m}^*)).$$

Fig. 1(h) shows how two methods synthesize the intermediate view at $\mathbf{m}^*$ with real measurements. It should be pointed out that the PBR scheme is described with the viewpoint of a virtual camera, not a reference camera. Namely, the set of probability values, listed in the table of the PBR method in Fig. 1(h), is not from a reference pixel $\mathbf{m}^*$ on the left image. These values are collected from different reference pixels, e.g. $\{(x^* + d, y^*)|d \in [0, 1, 2, \ldots, D]\}$ where $\mathbf{m}^* = (x^*, y^*)$ and D is a predefined search range. As in (16) and (17), we described the DIBR after warping the (already estimated) disparity $d_l^w(\mathbf{m}^*)$ to the virtual camera. Thus, the matching probability $P_l(\mathbf{m}^*, d)$ is 1 only at $d = d_l^w(\mathbf{m}^*)$, not $d = \arg \max_d P_l(\mathbf{m}^*, d)$. In fact, the left disparity value $d_l^w(\mathbf{m}^*) = 0$ at $\mathbf{m}^*$ is estimated by seeking the maximum among the set of the probability values obtained by varying the disparity d based on $\mathbf{m}^*$ of the left image.
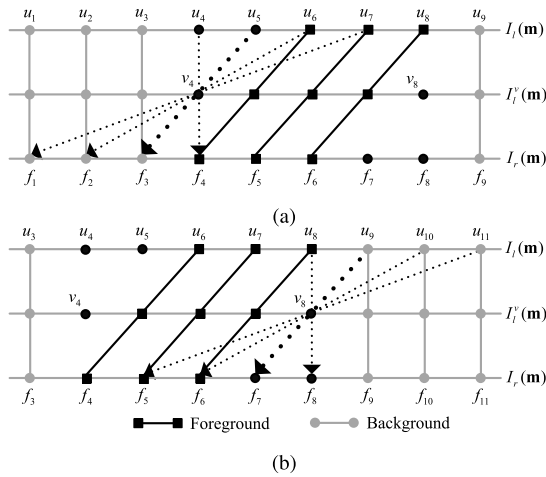
(a)

Fig. 2. Examples of (a) occlusion and (b) dis-occlusion region. u_4 and u_5 (f_7 and f_8): occluded pixels in the left (right) image. v_4 and v_8: occlusion and dis-occlusion (i.e., holes) in the virtual view. $I_l(u_4)$ and $I_r(f_8)$: the ground truth texture at v_4 and v_8. In the PBR method, $I_l(u_5)$ and $I_l(u_9)$ in the left image are likely to have the largest MP (contribution) values in rendering $I_l^v(v_4)$ and $I_l^v(v_8)$, respectively. With the assumption that the background texture smoothly varies, i.e., $I_l(u_4) \approx I_l(u_5)$ and $I_r(f_8) \approx I_r(f_7) \approx I_l(u_9)$), the PBR method yields visually pleasing textures ($I_l(u_5)$ and $I_l(u_9)$) into the holes, although they are not the ground truth textures. The rendering results using the right image is also similar. Please refer to Section IV-B for more details.

This figure demonstrates that wrong depth estimates incurred by small errors on the MP seriously affect the view rendering quality in the DIBR. For example, the rendered texture of the virtual camera at $\mathbf{m}^*$ is $I_l^v(\mathbf{m}^*, d_l^w(\mathbf{m}^*) = 0) = I_l(\mathbf{m}^*)$ as shown in Fig. 1(c). In contrast, the proposed probabilistic fusion (i.e., a weighted sum using the MP) effectively disperses such errors in the rendering process as shown in Fig. 1(d).

Furthermore, since the PBR considers all probable matching points, holes do not occur even at the occluded region. Instead, the occlusion is implicitly handled. Detail analysis on occlusion and dis-occlusion handling is presented in the next section, and experimental results will be described in Section V-C.

B. PBR Analysis on Occlusion and Dis-occlusion

In this section, we will describe how the PBR handles the occlusion and dis-occlusion (hole) regions in a probabilistic manner. Our method does not require the Z-buffer [31] for depth ordering commonly used to handle occlusion in the existing DIBR approaches. An underlying assumption for our *implicit* occlusion handling and hole filling method is that the background texture smoothly varies [18].

Let us suppose that the virtual view is synthesized using the left image only. Fig. 2 shows two possible cases of occlusion and dis-occlusion regions. The points u_4 and u_5 in the left image and f_7 and f_8 in the right image describe pixels occluded on the right and left images, respectively. The points v_4 and v_8 in the virtual view depicts pixels to be filled in occlusion and dis-occlusion regions, respectively. The dotted lines denote the set of all probable matching points between
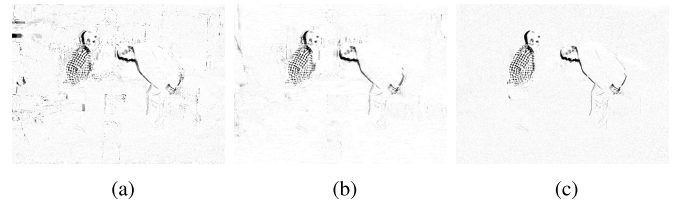

Fig. 3. Examples of temporal coherence in the virtual video rendering: difference images between consecutive frames obtained by (a) the DIBR, (b) the PBR and (c) the ground truth. It shows that the probabilistic blending (*soft* decision) used in the PBR can effectively resolve temporal artifacts, e.g., 'flickering' problem.

the reference images on the virtual camera coordinate. The solid lines connect the most probable matching (visible) point between images, meaning pixels with larger displacement (black solid lines) and smaller displacement (grey solid line) are likely to belong to the foreground and the background, respectively.

1) Occlusion [Fig. 2(a)]: In the DIBR, the depth values at occlusion regions (u_4 and u_5) are usually inferred from the background depth at the depth estimation stage. Ideally, the ground truth texture that should be filled in v_4 is that of u_4 in the left image. If there are no depth matching errors in the DIBR, the ground truth texture (from u_4) will be successfully assigned into v_4.

Now, let us examine how the PBR handles this case. Consider the MPs of four pairs of points (linked with the dotted line) between two reference images for v_4 on the virtual view. Two points u_4 and u_5, which are likely to be at the background, are linked to the pixels at the foreground (f_4) and the background (f_3) in the right image, respectively. Thus, the point u_5 is likely to have a higher MP than that of the point u_4. Other points (u_6 and u_7), belonging to the foreground in the left image, are linked to the background pixels (f_2 and f_1) in the right image, resulting in low MPs. Accordingly, in the weighted combination as in (15), the point u_5 linked to the background point f_3 has the largest contribution on the point v_4. Thus, it can be said that our probabilistic handling method does not give the ground truth texture at the occlusion. It is because our method does not consider the *geometric constraint* that occluded pixels tend to have a similar depth to background pixels. However, the PBR yields reasonable texture in the occlusion, assuming that the background texture varies smoothly, $I(u_4) \approx I(u_5)$ [18]. In contrast, when there exist depth estimation errors around the occlusion, visually annoying artifacts often occur in the view synthesized using the DIBR. It becomes even more serious on temporal aspects, e.g., 'flickering' problem on the synthesized video. Fig. 3 shows simple examples showing the temporal artifacts are effectively resolved in the proposed method.

2) Dis-occlusion [Fig. 2(b)]: Ideally, in the DIBR, the ground truth texture for v_8 is that of f_8 in the right image, when the occlusion is assumed to be perfectly handled in the depth estimation stage.

In the PBR, the newly exposed region v_8 is also filled in a probabilistic manner. Similarly, we also assume four pairs of linked points for v_8 in the virtual view: one point exists at the

foreground (u_8), and the others do at the background (u_9, u_{10}, and u_{11}) in the left image. In this case, u_9 has the highest MP, since it is connected with the background pixel f_7 in the right image, meaning its texture is highly likely to be filled with that of u_9 linked with f_7, whose texture is assumed to be very similar to that of f_8. Thus, similar to Fig. 2(a), one can render a visually pleasing texture into the pixel v_8, not a ground texture. As mentioned earlier, such rendering scheme is effective in suppressing the visual artifacts (e.g. flicker) of virtual video rendering, since the texture is synthesized via the probabilistic averaging process.

In the right image, the same process is applied except that the occlusion and dis-occlusion regions exist in the opposite direction to the case of the left image.

V. EXPERIMENTAL RESULT

We analyzed the performance of the proposed method through various experiments with the Middlebury data sets [32]. It is hard to perform a quantitative evaluation by only speculating an inferred MP. Instead, we compared the performance of depth estimation, assuming that a depth quality gets higher as a MP is inferred better. A depth map $d(\mathbf{m})$ was simply calculated with the inferred SSMP or MP as in (18), and was compared with that of state-of-the-art local stereo matching methods in terms of the depth accuracy and the number of parameters used:

$$d(\mathbf{m}) = \arg\max_d p^s(\mathbf{m}, d). \tag{18}$$

Afterwards, the PBR with different MPs and the SSMP was compared. Such evaluation clearly shows the effectiveness of our proposed inference method. Namely, the SSMP produces a high-quality MP even without specifying the window size and the number of iterations, enabling virtual view synthesis of reasonable quality. We also performed the view synthesis with less accurate MP and its corresponding low-quality depth map, inferred by the adaptive weight method [10]. The benefits from the proposed PBR method are more emphasized in this challenging condition. Lastly, we compared our rendering scheme with the DIBR approaches with various post-processing methods [5], [6]. Then, we will show that our PBR approach can even further improve the rendering quality, especially around depth discontinuities and occlusion. Furthermore, it will be shown that the PBR is effective in reducing the flicker artifacts of virtual video rendering significantly.

A. Experimental Environments and Implementation

As mentioned earlier, inferring MP is conceptually similar to aggregating cost commonly used in the local stereo matching methods. Thus, by employing the same initial MP (or cost), we can easily and fairly compare the proposed method with state-of-the-art local stereo methods: the adaptive weight (AW) [10], the anisotropic diffusion (AD)[2] [11], the cost filter (CF) [13], and the geodesic diffusion (GD) [14].

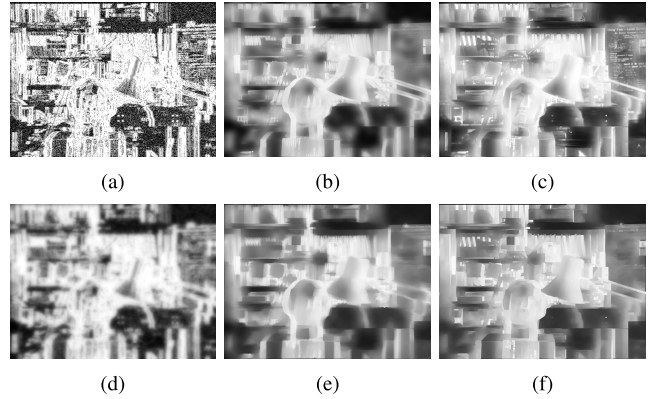[2]Note that the AD is not a main proposal of [11], but just a part of their method.

Fig. 4. An inferred matching probability in *Tsukuba* image when the disparity hypothesis is 0. With (a) an initial MP $p_l^0(\mathbf{m}, 0)$, an optimal probability is inferred by (b) the AW [10], (c) the CF [13], (d) the AD [11], (e) the GD [14], and (f) the SSMP. The low intensity indicates high probability of being matched, and *vice versa*.

For the left disparity estimation, we calculated an initial matching cost $e_l^0(\mathbf{m}, d)$ in (19), and then converted it into the MP $p_l^0(\mathbf{m}, d)$ of (1) with ν being set to 3000 in all experiments:

$$e_l^0(\mathbf{m}, d) = \begin{bmatrix} \delta \cdot \min\{||\psi(\mathbf{m}, d)||_1, \sigma_1\} \\ +(1 - \delta) \cdot \min\{||\nabla_x \psi(\mathbf{m}, d)||_1, \sigma_2\} \end{bmatrix} \tag{19}$$

$\psi(\mathbf{m}, d)$ describes differences between $I_l(\mathbf{m})$ and 'd'-shifted $I_r(\mathbf{m})$, i.e., $I_l(m_1, m_2) - I_r(m_1 - d, m_2)$, while ∇_x and $|| \cdot ||_1$ denote a gradient operator along a horizontal direction and l_1 norm, respectively. δ controls the leverage between an intensity and a gradient term, and it was set to 0.11. The truncation values σ_1 and σ_2 for handling outliers [13] were set to 15 and 2, respectively.

Since the performance of conventional local stereo matching methods largely depends on the window size and/or the number of iterations, they were carefully set through intensive experiments. The window size of the AW [10] and the CF [13] was set to 35×35 and 19×19, respectively. The AD [11], the GD [14] and the proposed method used 4-neighborhood. For the AD, the number of iterations was set to 200 with the step size being 0.01. In the GD, the number of iterations was set to 18 [14]. The range bandwidth of the AD, the GD, and the AW was set to 50. In the AW, the spatial bandwidth was set equal to twice of the window size. Other parameters were set equal to those of the original papers. The SSMP method requires only two parameters, the range bandwidth γ and the restarting probability α, and they were set to 50 and 0.003, respectively. All the parameters were fixed in experiments, unless otherwise specified. In the proposed method, the SSMP was inferred via (10). The results of the CF were simulated with the source code provided by the author [13]. Other methods were performed with our own implementations.

B. SSMP

We first demonstrate the capability of inferring the MP. Fig. 4 shows an initial MP $p_l^0(\mathbf{m}, 0)$ and inferred probability maps when a disparity hypothesis is 0. The AW [10]

TABLE I

OBJECTIVE COMPARISON FOR THE AW [10], THE CF [13], THE AD [11], THE GD [14], AND THE SSMP WITH VARYING A WINDOW SIZE AND THE NUMBER OF ITERATIONS

Sequence	Tsukuba	Venus	Teddy	Cones
Method	nonocc	nonocc	nonocc	nonocc
SSMP	2.12	1.28	7.52	3.50
AD(10)	12.3	16.1	16.70	8.06
AD(50)	5.88	4.94	10.90	4.20
AD(100)	4.97	3.40	10.40	4.00
AD(200)	4.56	2.95	10.20	3.98
GD(3)	10.10	12.50	14.00	6.92
GD(5)	6.83	6.24	10.80	4.88
GD(10)	4.00	2.17	8.49	3.90
GD(20)	2.99	1.40	7.89	4.38
CF(3)	20.40	31.20	35.10	27.70
CF(9)	5.74	4.03	10.60	4.19
CF(15)	3.13	1.98	8.68	2.96
CF(21)	2.34	1.56	8.03	3.05
CF(35)	2.47	1.08	8.42	4.17
AW(3)	20.70	29.60	29.80	19.70
AW(9)	6.64	5.87	10.60	3.76
AW(15)	3.95	1.87	8.75	2.70
AW(21)	3.08	1.20	8.07	2.57
AW(35)	2.41	1.01	7.59	2.61

shows better performance than the CF [13] in terms of edge-preserving smoothing. However, in both methods, non-smoothed texture regions are observed. The GD [14] localizes the probability map better than the AD [11], but both methods do not preserve the boundaries. In the SSMP, the texture region is effectively smoothed as well as the boundary is well-preserved.

1) Performance Analysis With Varying Parameters: To explore the influence of the parameters on the matching algorithms, we performed experiments by varying some parameters in Table I. The window size varied from 3×3 to 35×35 in the AW [10] and the CF [13]. The AD [11] and the GD [14] were performed by varying the number of iterations. In the AD [11] and the GD [14], the number in parentheses denotes the number of iterations. In the CF [13] and the AW [10], this means the diameter of the window. To demonstrate that the performance of the conventional methods largely depends on the parameters, we measured bad matching errors (%), i.e., the percentage of errors between estimated depth maps and the ground truth, via the Middlebury test bed [32]. Note that the bad matching error was measured only at non-occluded regions (*nonocc*) for fair comparison. The shadow cell describes the lowest error among parameters for each method. The dark cell depicts the lowest depth error among all methods including the SSMP method.

This table shows that the performance of the existing methods except the AD is seriously influenced by the number of iterations or the window size. The bad matching errors of the AD and the AW are reduced by enlarging the window size or increasing the iteration numbers. It, however, results in drastically increasing the processing time. Although the AD shows the smallest performance variation, the depth quality is far from the state-of-the-art methods. The GD also shows an acceptable quality. However, it is non-trivial to specify the proper number of iterations. The performance of the CF largely fluctuates, although it gives highly accurate results. Also, it is hard to find out a suitable window size. In contrast, the
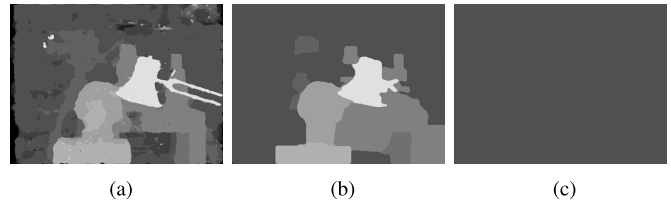

(a)   (b)   (c)

Fig. 5.   The results of the SSMP with varying a restarting parameter α. (a) 3×10^{-1}, (b) 3×10^{-5}, and (c) 3×10^{-7}. The results get smoother as the restarting probability α becomes smaller. When the random walker does not return to the initial position, i.e., $\alpha = 0$, the RWR gives trivial solution in the steady-state, i.e., the RWR-based approach becomes equivalent to the conventional RW-based approach.

(a)   (b)   (c)
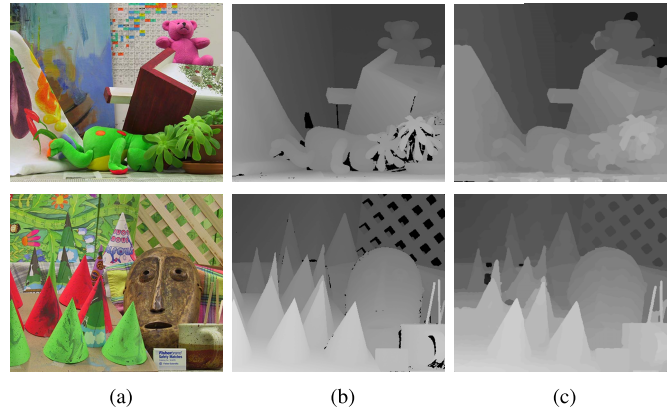
Fig. 6.   Results for (from top to bottom) *Teddy* and *Cones* [32] (a) left image, (b) ground truth, and (c) depth maps obtained from the SSMP. The proposed method provides high quality depth maps especially around depth boundaries, although only 4-neighborhood is used for inferring matching probability. See the Table II for an object comparison between the SSMP and other state-of-the-art matching methods.

SSMP method does not need to specify the window size and the number of iterations for the meaningful solution, since it reaches an asymptotic state while considering all paths between points with a fixed 4-neighborhood.

In the SSMP method, the restarting probability α is an important parameter since it determines the degree of smoothing of an initial MP. The results obtained with varying α are shown in Fig. 5. The disparity results get smoother as α becomes smaller, which implies that a random walker rarely returns to the initial position. Note that when $\alpha = 0$, the proposed scheme becomes equivalent to the RW-based methods such as the AW [10] and the AD [11], resulting in a trivial solution in an asymptotic state as shown in Fig. 5(c). As mentioned earlier, in our experiments, the restarting probability was fixed to 0.003.

2) Performance Analysis With Optimal Parameters: We also performed additional experiments including the occlusion handling. All the parameters for the AW [10], the CF [13], the GD [14], and the AD [11] were carefully set through intensive experiments for achieving the best performance. The left-right consistency criterion [11] was used to detect outliers, followed by a weighted median filter with default parameters [13].

Fig. 6 shows the results for (from top to bottom) *Teddy* and *Cones* [32] obtained by the SSMP. Even though the SSMP method uses a 4-neighborhood in inferring the matching probability, high quality depth maps were provided, especially

TABLE II
OBJECTIVE EVALUATION FOR THE SSMP WITH THE MIDDLEBURY TEST BED [32]

Sequence	Tsukuba			Venus			Teddy			Cones		
Method	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
CoorpRegion+	0.87	1.16	4.61	0.11	0.21	1.54	5.16	8.31	13.00	2.79	7.18	8.01
DoubleBP+	0.88	1.29	4.76	0.13	0.45	1.87	3.53	8.30	9.63	2.90	8.78	7.79
CrossLMF [33]	2.46	2.78	6.26	0.27	0.38	2.15	5.50	10.60	14.20	2.34	7.82	6.80
SSMP	1.60	1.97	6.44	0.20	0.38	2.51	6.15	11.50	15.80	2.60	7.92	7.48
CF* [13]	1.76	2.14	8.34	0.19	0.46	2.51	6.24	11.50	16.00	2.48	8.01	7.20
CF [13]	1.51	1.85	7.61	0.20	0.39	2.42	6.16	11.80	16.00	2.71	8.24	7.66
AdaptOvrSegBP+	1.69	2.04	5.64	0.14	0.20	1.47	7.04	11.13	16.45	3.60	8.96	8.84
GlobalGCP+	0.87	2.54	4.69	0.16	0.53	2.22	6.44	11.54	16.24	3.59	9.49	8.95
GeoSup	1.45	1.83	7.71	0.14	0.26	1.90	6.88	13.27	16.14	2.94	8.89	8.32
GD* [14]	2.39	2.96	11.50	0.26	0.44	3.35	7.20	12.40	17.50	3.19	8.78	9.14
GD [14]	1.88	2.35	7.64	0.38	0.82	3.02	5.99	11.30	13.30	2.84	8.33	8.09
Recursive BF	1.85	2.51	7.45	0.35	0.88	3.01	6.28	12.15	14.32	2.80	8.91	7.79
AW* [10]	2.43	2.77	11.60	0.24	0.46	2.45	7.18	12.50	17.40	2.59	8.23	7.38
AW [10]	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.30	18.60	3.97	9.79	8.26
AD* [11]	2.82	3.76	11.90	1.02	1.167	11.90	8.72	14.10	20.20	3.21	8.86	9.24

TABLE III
OBJECTIVE COMPARISON FOR THE PBR WITH THE AD [11],
GD [14], CF [13], AW [10], AND THE SSMP

Method	AD [11]		GD [14]		CF [13]		AW [10]		SSMP	
Sequence	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Tsukuba	29.95	0.93	29.37	0.93	30.85	0.94	30.27	0.93	30.84	0.94
Venus	34.45	0.93	31.69	0.92	34.33	0.93	33.59	0.92	33.47	0.92
Teddy	29.91	0.88	28.94	0.88	29.89	0.88	29.61	0.89	29.72	0.88
Cones	26.83	0.85	26.79	0.85	27.21	0.85	26.85	0.84	27.09	0.85
Art	26.07	0.87	25.97	0.86	26.85	0.88	26.42	0.87	26.74	0.88
Reindeer	25.69	0.90	25.93	0.90	26.05	0.90	28.57	0.90	26.11	0.90
Aloe	29.60	0.92	28.97	0.90	29.84	0.92	29.28	0.90	29.64	0.91
Cloth3	35.29	0.97	34.66	0.97	34.44	0.96	33.79	0.96	34.68	0.97
Average	29.72	0.91	29.04	0.90	29.93	0.91	29.80	0.90	29.79	0.91

Fig. 7. Intermediate views for the PBR with the SSMP. (From left to right, and from top to bottom) *Art*, *Reindeer*, *Aloe*, and *Cloth3* [32].

around depth boundaries. For an object comparison, we evaluated these methods in Table II according to three criteria: non-occluded region (*nonocc*), all region (*all*), and discontinuous region (*disc*). The symbol '*' indicates the results of our implementations. It shows that the SSMP shows competitive results with state-of-the-art methods including global methods denoted as the symbol '+'. The SSMP ranks second within local methods followed by CrossLMF [33].

C. PBR

Next, we evaluate the performance of the PBR. An intermediate views were rendered with the left and right images, and the PSNR and structural similarity (SSIM) index [34] were measured with ground truth color images [32]: *Tsukuba* (from the view point of 2 and 4 to that of 3), *Venus/Teddy/Cones* (from the view point of 2 and 6 to that of 4), and *Art/Reindeer/Aloe/Cloth3* (from the view point of 1 and 5 to that of 3). Furthermore, in the virtual video rendering, the capability of the PBR for maintaining temporal coherence is demonstrated with the MPEG sequences [35], [36]: *Vassar* (from the view point of 0 and 2 to that of 1), *BookArrival* (from the view point of 6 and 10 to that of 8), *Poznan* (from

the view point of 3 and 5 to that of 4), and *GtFly* (from the view point of 1 and 9 to that of 5).

1) Performance Analysis With different MPs and SSMP: As mentioned earlier, the MPs can also be applicable to the PBR. To validate this, we synthesized intermediate views via the PBR with different MPs and the SSMP. Except for the SSMP, all parameters needed for inferring MPs were set through exhaustive searches for achieving the best rendering results. Table III compares the performance of the PBR with the SSMP and MPs inferred from various methods [10], [11], [13], [14]. It shows that the PBR can successfully render intermediate views without handling the occlusion and dis-occlusion regions as mentioned in Section IV-B. All methods tend to give better results when sequences with relatively simple geometry are used such as *Venus* and *Cloth3*. The PBR with the SSMP method does not need to specify the window

TABLE IV
OBJECTIVE COMPARISON FOR THE DIBR [5], [6], [17] AND THE PBR WITH THE AW [10]

Method	DIBR [17] (NOcc)		DIBR [17] (NOcc+POST1)		DIBR [17] (Occ)		DIBR [17] (Occ+POST1)		DIBR [5] (Occ+POST2)		DIBR [6] (Occ+POST3)		PBR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
AW(1)	13.43	0.22	24.28	0.70	15.67	0.34	23.44	0.68	24.22	0.74	24.08	0.74	25.10	0.76
AW(7)	19.42	0.70	27.51	0.86	23.65	0.81	27.32	0.87	27.78	0.89	27.68	0.88	28.20	0.88
AW(21)	22.83	0.83	28.97	0.89	29.07	0.91	29.58	0.91	29.10	0.91	28.99	0.92	29.80	0.90

size and the number of iterations for inferring the probability as shown in Section V-B. Nevertheless, it achieved the comparable performance to the PBR with other MPs, in which optimal parameters were estimated through intensive experiments for obtaining the best rendering quality. Fig. 7 shows examples of virtual views synthesized by the PBR with the SSMP. In the following sections, we will show the superiority of the PBR over the DIBR for various image and video sequences.

2) Comparison to DIBR With Still Images: To verify the performance of the PBR more deeply, the intermediate views were rendered via the PBR and the DIBR with MPs and its corresponding depth maps, obtained by the AW [10] using an aggregation window with varying sizes: AW(1), AW(7), and AW(21). Note that AW(1) can be seen as using an initial raw MP with no aggregation.

Table IV shows an objective comparison for the DIBR [17] including various post-processing methods [5], [6] and the PBR with the AW [10]. It shows average PSNR and SSIM scores for eight image sequences used in Section V-C-1). The results rendered with the depth maps, where the occlusion is handled, are denoted as 'DIBR(Occ)', and 'DIBR(NOcc)' otherwise. The results of the DIBR inherently have hole regions in contrast to the PBR. 'DIBR(Occ+POST1)' denotes the results after interpolating such hole regions: 1) The depth maps are warped onto the virtual camera. 2) The holes in the warped depth maps are then interpolated with a simple median filtering for a dense photometric sampling. 3) The virtual view is synthesized through a backward warping. 4) Finally, the interpolation on hole regions, which may exist in the synthesized view, is executed. That is, the interpolation technique is used in both warped depth maps and synthesized views. 'DIBR(Occ+POST2)' denotes the results after reducing the rendering errors in a way that artifacts caused by unreliable pixels from one reference image are replaced with the reliable ones from another reference image [5]. We also compare our results with 'DIBR(Occ+POST3)' which constrains the reliability of each pixel in the sense of maximizing likelihood [6].

There are four observations. First, the post processing is inevitably needed in the DIBR. Averagely, it achieves 7.77 (*dB*) PSNR gain compared to the results without the hole filling, e.g., DIBR(Occ) and DIBR(Occ+POST1) using AW(1). Second, enlarging the window size improves the performance of all methods, since it increases the quality of the MP and the depth map. Although some post-processing methods [5], [6] relax the rendering error to some extent, they are not effective in rendering with a low quality depth and
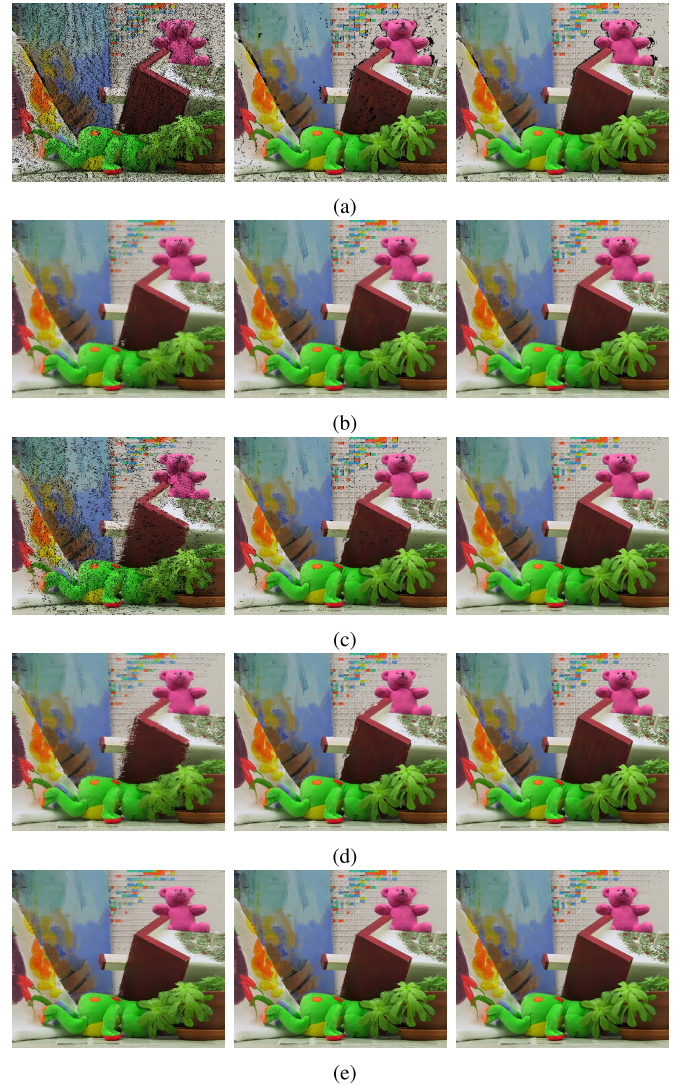

Fig. 8. Intermediate views of *Cones* [32] for (a) the DIBR(NOcc), (b) the DIBR(NOcc+POST1), (c) DIBR(Occ), (d) DIBR(Occ+POST1), and (e) the PBR with (from left to right) the AW(1), the AW(7), and the AW(21), respectively. Although the PBR implicitly handle the occlusion and dis-occlusion regions, it show the sharp transition around the boundaries in contrast to the DIBR. Furthermore, the PBR is more powerful when the low quality MP is given, since it relaxes the errors from local minima. See the Table IV for the object comparison.

a relatively high quality depth, i.e., the depth maps obtained by AW(1) and AW(21). Third, the PSNR gain of the PBR over the DIBR becomes more remarkable as the window size becomes smaller, i.e., the PSNR gains of the PBR over the DIBR(Occ+POST1) [17] are 1.66 (*dB*) in AW(1), and
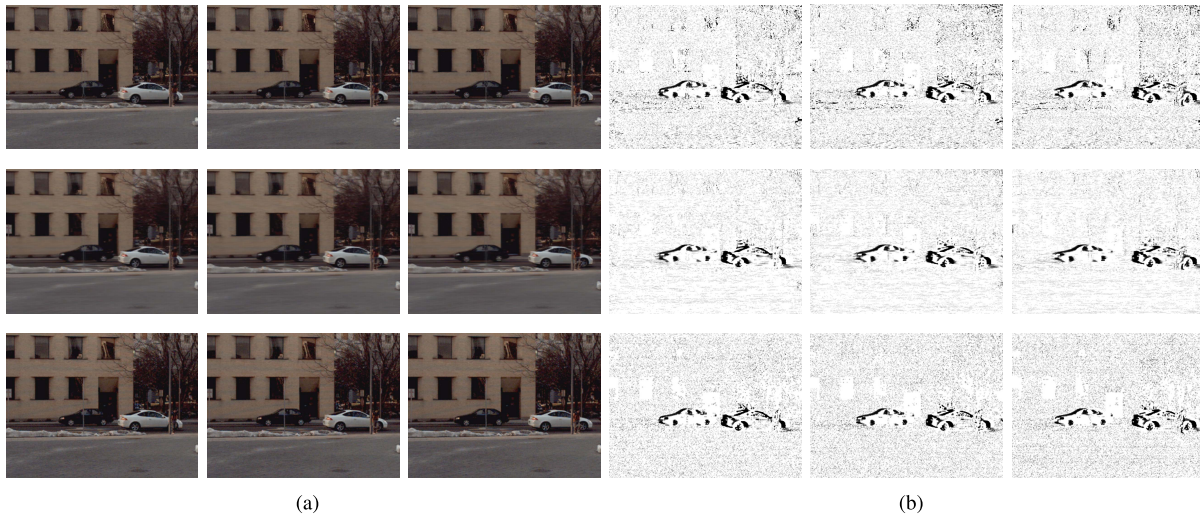
<div align="center">(a)   (b)</div>

Fig. 9. (a) Intermediate views for (from left to right) 169th, 170th, and 171th frames of *Vassar* sequences rendered by (from top to bottom) the DIBR(Occ+POST1), the PBR, and the ground truth, respectively. (b) the difference images between consecutive sequences corresponding to (a). The *soft* decision in the PBR reduces temporal noises and flickering artifacts in the rendered video, maintaining temporal coherence compared to the DIBR.

TABLE V
OBJECTIVE COMPARISON FOR THE DIBR(OCC+POST1) [17]
AND THE PBR WITH VIDEO SEQUENCES

Method	DIBR [17] (Occ+POST1)			PBR		
Sequence	PSNR	SSIM	$\mathcal{F}$	PSNR	SSIM	$\mathcal{F}$
Vassar	28.26	0.78	1.47	28.68	0.80	0.53
BookArrival	27.90	0.85	1.53	30.50	0.88	0.55
Poznan	34.06	0.95	0.56	35.00	0.95	0.29
GtFly	39.15	0.98	0.50	39.00	0.98	0.38

0.88 (*dB*) in AW(7). It indicates that the PBR is more powerful when the low-quality MP is employed. Fourth, the occlusion handling significantly reduces the hole regions, i.e., the results of the DIBR(Occ) [17] show better performance than those of the DIBR(NOcc) [17], since wrong depth values in the occlusion make pixels of the reference images warped to wrong positions, eventually causing many holes. However, when the post-processing is applied with the low-quality depth maps of AW(1) and AW(7), the occlusion handling does not work well. Comparing the results of the DIBR(NOcc+POST1) [17] and DIBR(Occ+POST1) [17], the methods with an occlusion handling sometimes produces even worse rendered results. It is because an accurate detection and a handling of occluded regions are very hard in case of using severely degraded depth maps, so lots of rendering errors still exist. We can also find these phenomena visually in Fig. 8(b) and (d), which shows view synthesis results of *Cones*. In contrast, the results of the PBR always show a sharp transition at the boundary even with a low quality MP, thanks to the capability of effectively handling matching outliers. Therefore, we can conclude that as stated in Section IV-B, although the PBR does not provide a ground truth texture in the occluded regions, it can be an excellent alternative when serious errors exist in the depth maps, which is very common in practical environments.

3) Comparison to DIBR With Video Sequences: The above argument is even more strongly supported in the virtual *video* rendering. The *soft* decision in the PBR relaxes temporal artifacts, which is a significant problem in the conventional DIBR caused by inconsistent depths in the temporal domain. To verify this, we synthesized virtual videos using the DIBR and the PBR with the sequences provided by the MPEG [35], [36]: *Vassar*, *BookArrival*, *Poznan* and *GtFly*. The depth maps and the MPs were inferred by the AW with a fixed aggregation window size, i.e., AW(31).

Fig. 9(a) shows examples of synthesized views for 169th, 170th, and 171th frames of *Vassar* sequences rendered by (from top to bottom) the DIBR(Occ+POST1) [17], the PBR, and the ground truth, respectively. For observing the temporal fluctuation, difference images between consecutive frames were obtained by (from top to bottom) the DIBR(Occ+POST1) [17], the PBR, and the ground truth, respectively as shown in Fig. 9(b). It demonstrates that some temporal artifactes such as 'flicker' problem can be successfully resolved by the PBR. One interesting observation is that the PBR handled even the artifacts in the original video. Namely, it can be seen that the temporal noise in the ground truth color video was also reduced in the video rendered by the PBR, since the *soft* decision in the PBR enables the scene to be smoothly varied without any temporal reasoning, e.g., optical flow.

The temporal coherence was measured for an objective evaluation as follows [37]:

$$\mathcal{F} = \frac{1}{MT} \sum_{\mathbf{m}} \sum_{t=1}^{T} \mathcal{F}_t(\mathbf{m})$$

$$\mathcal{F}_t(\mathbf{m}) = \max\{0, |I_t^v(\mathbf{m}) - I_{t-1}^v(\mathbf{m})| - |R_t(\mathbf{m}) - R_{t-1}(\mathbf{m})|\} \quad (20)$$

where M and T denote the image size and the total number of sequences, respectively. $I_t^v(\mathbf{m})$ and $R_t(\mathbf{m})$ are the synthesized view and the ground truth view at time t, respectively. The $\mathcal{F}$
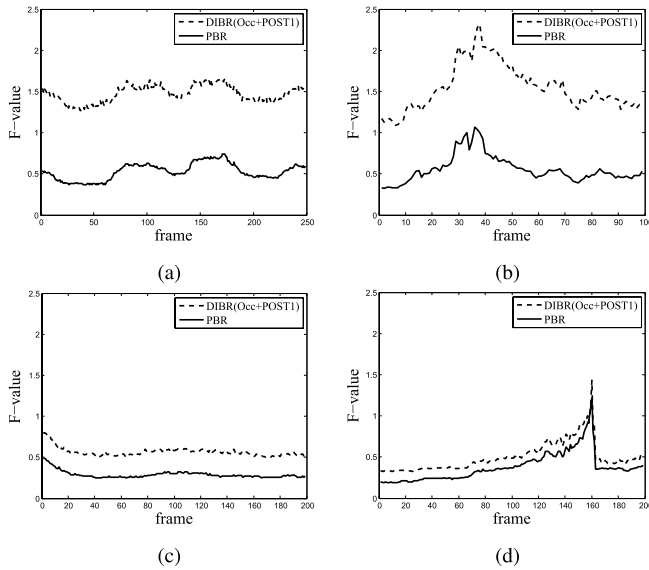
Fig. 10. The flutuation measure $\mathcal{F}$ for synthesized videos rendered by the DIBR(Occ+POST1) and the PBR. (a) *Vassar*, (b) *BookArrival*, (c) *Poznan*, and (d) *GtFly*. It is shown that the PBR can drastically reduce the temporal artifacts compared to the DIBR, enforcing temporal coherence.

Fig. 11. The textures from the foreground and the background are mixed if the smoothness assumption of the background texture is not valid, e.g., thin object, making such objects transparent in the synthesized view. (See thin wooden sticks.)

value measures the temporal coherence with the just noticeable difference for the flicker artifacts $|R_t(\mathbf{m}) - R_{t-1}(\mathbf{m})|$, and is highly correlated with an actual viewing experience [37]. The low $\mathcal{F}$ value means that the synthesized video is temporally coherent, and has low flickering artifacts, and *vice versa*. Fig. 10 demonstrates that the PBR can drastically reduce the temporal artifacts compared to the conventional DIBR. Table V shows a numerical comparison between the DIBR and the PBR for average PSNR, SSIM, and $\mathcal{F}$ values of entire sequences. The PSNR and SSIM gains of the PBR over the DIBR seem relatively low, since they are measured for whole images, e.g., including textureless regions, which are hard to verify the quality of the rendering results. But, it is noteworthy that the PBR decreases the fluctuation degree ($\mathcal{F}$) near 50% compared to the DIBR. All the rendering results can be found at [38].

4) Limitation: Although the PBR gives reasonable rendering results, some problems still occur at the regions such as thin objects where the smoothness assumption of the background texture is not valid. In these regions, the textures

from the foreground and the background are mixed, since two modes for the foreground and the background in the MP may often have similar quantities. It makes the objects in the synthesized view transparent, e.g. thin wooden sticks in the *Art* sequence of Fig. 11.

VI. DISCUSSION AND CONCLUSION

We have described the PBR method for the robust reconstruction of a high-quality intermediate view. Two main issues were addressed: The first is how the optimal MP is inferred. After reformulating the cost aggregation into the probability optimization, we applied the RWR to obtain the SSMP, which always guarantees a meaningful solution in the steady-state unlike RW or filtering-based methods. The second is how the intermediate view is robustly rendered with the SSMP or MP. Motivated by the observation that all pixels in the intermediate view come from reference images, we formulated a rendering process as an image fusion in a way that the intermediate view is rendered by adaptively blending textures of all possible matching pixels. Thus, the PBR method using a soft decision effectively addresses the errors which might be incurred by an inaccurate depth estimation. The intensive experiments demonstrated that the PBR is superior to the DIBR in both qualitative and quantitative manners. It is also free from the dis-occlusion (hole filling) and occlusion problems. Though our method does not render a ground truth texture on the occluded pixels, rendered results are visually more coherent on temporal aspects than those of the DIBR method.

The procedure of the PBR is parallel and thus, it can be efficiently implemented via the parallel processing unit such as the graphic processing unit (GPU). Furthermore, the frame rate up-conversion is fundamentally similar to the intermediate view rendering, i.e., it also consists of the correspondence matching and rendering stages dealing with a 2D displacement vector. We will generalize the proposed scheme to more complicated problem, e.g., the frame rate up-conversion and the virtual video rendering on a non-parallel camera configuration.

REFERENCES

[1] S.-W. Jung and S.-J. Ko, "Depth sensation enhancement using the just noticeable depth difference," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3624–3637, Aug. 2012.

[2] Y. Zhao, C. Zhu, Z. Chen, and L. Yu, "Depth no-synthesis-error model for view synthesis in 3D video," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2221–2228, Aug. 2011.

[3] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, and Y. Mori, "Reference softwares for depth estimation and view synthesis," MPEG, Tech. Rep. M15377, 2008.

[4] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *Proc. 20th Annu. Conf. Comput. Graph. Interact. Tech.*, 1993, pp. 279–288.

[5] L. Yang, T. Yendo, M. P. Tehrani, T. Fujii, and M. Tanimoto, "Artifact reduction using reliability reasoning for image generation of FTV," *J. Vis. Commun. Image Represent.*, vol. 21, nos. 5–6, pp. 542–560, 2010.

[6] L. Yang, T. Yendo, M. P. Tehrani, T. Fujii, and M. Tanimoto, "Probabilistic reliability based view synthesis for FTV," in *Proc. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1785–1788.

[7] C. Oh, B. Ham, and K. Sohn, "Probabilistic correspondence matching using random walk with restart," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 37–57.

[8] D. Scharstein and R. Szeliski, "A taxanomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, nos. 1–3, pp. 7–42, 2002.

[9] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 41–54, 2006.

[10] K. Yoon and I. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 650–656, Apr. 2006.

[11] D. Min and K. Sohn, "Cost aggregation and occlusion handling with WLS in stereo matching," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1431–1442, Aug. 2008.

[12] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 1–14.

[13] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 3017–3024.

[14] L. De-Maeztu, A. Villanueva, and R. Cabeza, "Near real-time stereo matching using geodesic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 410–416, Feb. 2012.

[15] D. Min, J. Lu, and M. N. Do "A revisit to cost aggregation in stereo matching: How far can we reduce its computational redundancy?" in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 1567–1574.

[16] L. Zhang, D. Wang, and A. Vincent, "Adaptive reconstruction of intermediate views from stereoscopic images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 102–113, Jan. 2006.

[17] D. Min, D. Kim, S. Yun, and K. Sohn, "2D/3D freeview video generation for 3DTV system," *Signal Process., Image Commun.*, vol. 24, nos. 1–2, pp. 31–48, 2009.

[18] A. Fitzgibbon, Y. Wexler, and A. Zisserman, "Image-based rendering using image-based priors," *Int. J. Comput. Vis.*, vol. 63, no. 2, pp. 141–151, 2005.

[19] D. Mahajan, F.-C. Huang, W. Matusik, R. Ramamoorthi, and P. Belhumeur, "Moving gradients: A patch-based method for plausible image interpolation," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1–11, 2009.

[20] M. Lang, A. Hornung, O. Wang, S. Poulakos, A. Smolic, and M. Gross, "Nonlinear disparity mapping for stereoscopic 3D," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–10, 2010.

[21] H. T. Nguyen and M. N. Do, "Error analysis for image-based rendering with depth information," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 703–716, Apr. 2009.

[22] K. Takahashi, "Theoretical analysis of view interpolation with inaccurate depth information," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 718–732, Feb. 2012.

[23] Y. Kunita, M. Ueno, and K. Tanaka, "Layered probability maps: Basic framework and prototype system," in *Proc. ACM Symp. Virtual Reality Softw. Technol.*, 2006, pp. 181–188.

[24] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.

[25] R. Shen, I. Cheng, J. Shi, and A. Basu, "Generalized random walks for fusion of multi-exposure images," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3634–3646, Dec. 2011.

[26] J. Y. Pan, H. J. Yanh, C. Faloutsos, and P. Duygulu, "Automatic multimedia cross-modal correlation discovery," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 653–658.

[27] T. Kim, K. Lee, and S. Lee, "Generative image segmentation using random walks with restart," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 264–275.

[28] B. Ham, D. Min, and K. Sohn, "A generalized random walk with restart and its application in depth up-sampling and interactive segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2574–2588, Jul. 2013.

[29] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 106–128, Jan. 2013.

[30] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[31] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[32] *Middlebury Stereo Benchmarks* [Online]. Available: http://vision.middlebury.edu/stereo/

[33] J. Lu, K. Shi, D. Min, L. Lin, and M. N. Do, "Cross-based local multipoint filtering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognir.*, Jun. 2012, pp. 430–437.

[34] Z. Wang, A. C. Bovik, H. Rahim, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[35] A. Vetro, M. McGuire, W. Matusik, A. Behrens J. Lee, and H. Pfister, "Multiview video test sequences from MERL," Doc. M12077, Apr. 2005.

[36] Poznan University of Technology, "Call for proposals on 3D video coding technology," Doc. M17900, Mar. 2011.

[37] H. Yang, J. M. Boyce, and A. Stein, "Effective flicker removal from periodic intra frames and accurate flicker measurement," in *Proc. IEEE Conf. Image Process.*, Oct. 2008, pp. 2868–2871.

[38] *Supplementary Materials* [Online]. Available: http://diml.yonsei.ac.kr/~ham/pbr

Bumsub Ham (M'14) received the B.S. and Ph.D. degree in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2008 and 2013, respectively. He is currently with Yonsei University as a Post-Doctoral Researcher. He was a recipient of the Honor Prize in 17th Samsung Human-Tech Prize in 2011, and the Grand Prize in Qualcomm Innovation Fellowship in 2012. His research interests include variational methods and geometric partial differential equations, both in theory and applications in computer vision and image processing, particularly regularization, stereo vision, super-resolution, and HDR imaging.
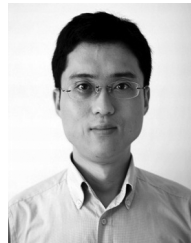
Dongbo Min (M'09) received the B.S., M.S., and Ph.D. degree in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2003, 2005, and 2009, respectively. He then worked with the Mitsubishi Electric Research Laboratories (MERL) as a Post-Doctoral Researcher from 2009 to 2010. He is currently with the Advanced Digital Sciences Center, which was jointly founded by the University of Illinois at Urbana-Champaign and the Agency for Science, Technology, and Research, a Singapore government agency. His research interests include 3D computer vision, video processing, 3D modeling, and hybrid sensor systems.

Changjae Oh (S'13) received the B.S. and M.S. degree in electrical and electronic engineering from Yonsei University, Seoul, Korea, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree with Yonsei University. His current research interests include 3D computer vision, 3D visual fatigue assessment, and image denoising.

Minh N. Do (M'01–SM'07–F'14) was born in Vietnam, in 1974. He received the B.Eng. degree in computer engineering from the University of Canberra, Canberra, Australia, in 1997, and the Dr.Sci. degree in communication systems from the Swiss Federal Institute of Technology Lausanne, Lausanne, Switzerland, in 2001.

Since 2002, he has been with the faculty of the University of Illinois at Urbana-Champaign (UIUC), Urbana, where he is currently an Associate Professor with the Department of Electrical and Computer Engineering and holds joint appointments with the Coordinated Science Laboratory, Beckman Institute for Advanced Science and Technology, and the Department of Bioengineering. His research interests include image and multidimensional signal processing, wavelets and multiscale geometric analysis, computational imaging, augmented reality, and visual information representation.

Prof. Do is a member of the IEEE Signal Processing Theory and Methods and Image, Video, and Multidimensional Signal Processing Technical Committees. He is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He was a recipient of the Silver Medal in the 32nd International Mathematical Olympiad in 1991, the University Medal from the University of Canberra in 1997, the Doctorate Award from the EPFL in 2001, the CAREER Award from the National Science Foundation in 2003, the Xerox Award for Faculty Research from the College of Engineering, UIUC, in 2007, and the Young Author Best Paper Award from IEEE in 2008. He was named a Beckman Fellow in the Center for Advanced Study, UIUC, in 2006.

Kwanghoon Sohn (M'92–SM'12) received the B.E. degree in electronic engineering from Yonsei University, Seoul, Korea, in 1983, the M.S.E.E. degree in electrical engineering from the University of Minnesota in 1985, and the Ph.D. degree in electrical and computer engineering from North Carolina State University in 1992. He was a Senior Member of the Research Staff with the Satellite Communication Division, Electronics and Telecommunications Research Institute, Daejeon, Korea, from 1992 to 1993, and as a Post-Doctoral Fellow with the MRI Center in the Medical School, Georgetown University in 1994. He was a Visiting Professor with Nanyang Technological University from 2002 to 2003. He is currently a Professor with the School of Electrical and Electronic Engineering, Yonsei University. His research interests include 3D image processing, computer vision, and image communication. He is a member of SPIE.