Eurographics Symposium on Rendering (2004)
H. W. Jensen, A. Keller (Editors)

All-focused light field rendering

Akira Kubota[1,2†], Keita Takahashi[1], Kiyoharu Aizawa[1], and Tsuhan Chen[2]

[1]University of Tokyo, Tokyo, Japan
[2]Carnegie Mellon University, Pittsburgh, USA

Abstract

We present a novel reconstruction method that can synthesize an all in-focus view from under-sampled light fields, significantly suppressing aliasing artifacts. The presented method consists of two steps; 1) rendering multiple views at a given view point by performing light field rendering with different focal plane depths; 2) iteratively reconstructing the all in-focus view by fusing the multiple views. We model the multiple views and the desired all in-focus view as a set of linear equations with a combination of textures at the focal depths. Aliasing artifacts can be modeled as spatially (shift) varying filters. We can solve this set of linear equations by using an iterative reconstruction approach. This method effectively integrates focused regions in each view into an all in-focus view without any local processing steps such as estimation of depth or segmentation of the focused regions.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation Viewing algorithms ; I.4.3 [Image Processing and Computer Vision]: Enhancement Filtering

1. Introduction

Various types of image-based rendering (IBR) and image-based modeling and rendering (IBMR) techniques for rendering a novel view have been presented [SHC03]. IBR/IBMR has been one of the most attractive research in the fields of computer graphics, computer vision and image processing. IBMR methods (for instance [CW93] [SD96]) use computer vision methods for estimating geometrical information such as 3D geometry and feature correspondence of the scene and then apply computer graphics methods for rendering the novel view based on the obtained scene information. Using these methods, the errors in the obtained geometry crucially affect the quality of the final result, where undesirable deformations are visible, especially when the novel view point is changed. In addition, it is generally hard to obtain such information for real scenes with sufficient accuracy. In contrast to IBMR methods, IBR methods (for instance [LH96] [GGSC96] [SH99]) do not require geometry information. Instead, IBR requires a large number of reference images taken with densely arranged cameras for rendering a novel view with sufficient quality for a large scene. The required number of cameras (i.e., sampling density of light field on camera plane) is too large to build the camera array in practice.

In this paper, we present a novel IBR method that allows more sparsely arranged cameras for capturing reference images compared with the conventional IBR. In our approach, first we assume multiple focal planes at different depths in the scene, and render multiple novel views at the same view point using light field rendering (LFR) [LH96] at each of the focal planes. In each rendered view, although the regions in the focal depth appear in focus, the regions not in the focal depth suffer from aliasing artifacts. Second, by fusing the multiple views using an iterative reconstruction method, we reconstruct an all in-focus view where aliasing artifacts are significantly suppressed.

2. Background and related work

When using a single focal plane, we can not render the novel view using LFR without aliasing for a large depth scene. This limitation is shown by Plenoptic sampling theory [CTCS00]. One solution is to suppress aliasing artifacts by pre-filtering [LH96] as used in LFR. However, this pre-filtering results in the degradation of the rendering quality. Stewart et al. [SYGM03] have presented a new reconstruction filter that suppresses artifacts by cutting the high fre-

† kubota@hal.t.u-tokyo.ac.jp

quency components of regions at incorrect depths, and preserves textures at the correct depth by wide-aperture reconstruction [ILG00]. Nevertheless, aliasing artifacts still remain visibly in the final result. This method cannot render an all in-focus view, since the in-focus regions would be overlapping with respect to each other, due to the fact that wide aperture reconstruction picks up the occluded regions much more often.

Another idea is to use multiple focal planes. Recently, two criteria have been presented to locally measure the sharpness (or focus) of a region for the purpose of extracting the focused region from the multiple views rendered using LFR at multiple depths. Isaksen et al. [ILG00, ILG99] have measured the smoothness (consistency) of the pixel values to be used for the rendering at each depth. This idea is essentially equivalent to that underlying stereo matching. Takahashi et al. [TKN03] have presented a stable focus measure using the difference of the views that are generated through different kinds of reconstruction filters at the same focal plane. Both approaches result in estimating the view dependent depth map.

The method proposed in this paper can reconstruct an all in-focus view directly from the multiple interpolated views without depth map estimation. We model aliasing artifacts as spatially varying filters and the multiple rendered views as a set of linear equations with a combination of unknown textures at the focal depths. We can solve this set of linear equations for the textures by using an iterative reconstruction method and obtain the desired all in-focus view as the sum of the solved textures. This method effectively integrates the focused regions in each view into an all in-focus view with less error. Kubota et al. [KA00] have used the same linear combination model for representing multi-focus images captured with a physical camera and generated an all-focused image from them using filter in the Fourier domain. In this case, defocus effect in the captured images become a low-pass filter which is spacial invariant and can be modeled in the Fourier domain.

The proposed method does not use any computer vision techniques such as feature extraction and depth estimation. Our iterative reconstruction technique is a new idea that is very different from the conventional computer vision algorithms used for the depth-from-defocus/focus problem [Nay92] [BK93]. Conventional algorithms have tried to detect or extract the regions that are in-focus, which is equivalent to estimation of depth map, and combine them into an all in-focus image. However, those conventional algorithms can not be applied to the problem of creating an all in-focus view from multiple views rendered by LFR at multiple depths. This is because the ghosting artifacts differ from defocus; the ghosting artifacts are not just a low-pass filtering effect, and different artifacts occur in different pixels in the rendered view (i.e., they are shift varying artifacts), even at the same virtual view point. Of course, different ghosting artifact occur in a novel view at different view point. This property makes our fusion problem more difficult from the conventional depth-from-focus problem.

3. All-focused light field rendering through fusion

3.1. Light field parameterization and rendering

In this section, we define the light field parameterizations used in this paper and describe the conventional light field rendering method with a constant depth based on those parameterizations. For the most part, we follow the notation used in [ILG00].

The two-plane parameterization was originally used for parameterizing the light field; each light ray is uniquely parameterized by its intersections with two planes, a camera plane parameterized with $(s,\ t)$ and a focal plane parameterized with $(u,\ v)$. As with the parameterization in [ILG00], in this paper we index the camera position using $(s,\ t)$ and use $(u,\ v)$ as the pixel position on the imaging plane at each camera. The focal plane is defined as a depth plane that we assume in the scene when rendering a novel view by LFR. The depth of the focal plane is called the focal depth. We also express the virtual camera position (novel view point) as $(s_r,\ t_r,\ z_r)$ in three-dimensional space (s,t,z) and its pixel coordinates as $(x,\ y)$. Axis z indicates the depth from the camera plane. Each light ray is sampled by cameras $C_{i,j}$; the cameras are located at $(s_i,\ t_j)$ on the camera plane with interval of Δs and Δt along the s and t axes, respectively, (i.e., $\Delta s = s_{i+1} - s_i$ and $\Delta t = t_{j+1} - t_j$). We express the sampled light ray as $l(s_i,\ t_j, u_i,\ v_j)$.

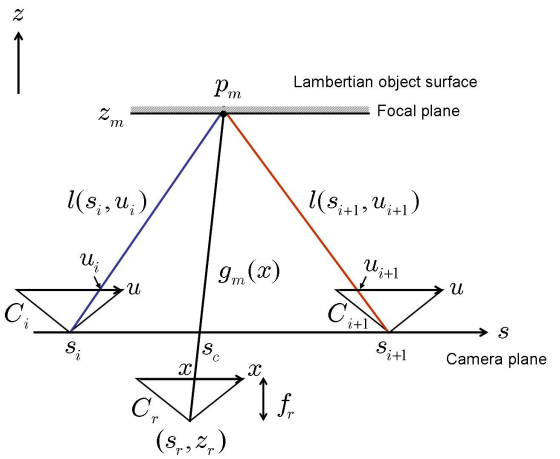
The conventional light field rendering with a constant focal depth is performed as follows. For simplicity, consider a two dimensional version of our light ray parameterization where parameters t and v are fixed (the camera's y coordinate is also fixed), as shown in Figure 1(a). Let $g_m(x)$ be a novel ray that is rendered with the virtual camera C_r at view position (s_r, z_r) using LFR with focal depth z_m. First, the two intersections of the novel ray with the camera plane and the focal plane are calculated, say s_c and p_m, respectively. Second, the two camera positions near s_c are calculated, say s_i and s_{i+1}. Projecting p_m onto imaging planes at the two cameras gives us the two corresponding pixel positions u_i and u_{i+1}. The novel ray $g_n(x)$ is computed as the weighted average of the two sampled rays $l(s_i, u_i)$ and $l(s_{i+1}, u_{i+1})$:

$$g_m(x) = w_i l(s_i, u_i) + w_{i+1} l(s_{i+1}, u_{i+1}) \qquad (1)$$

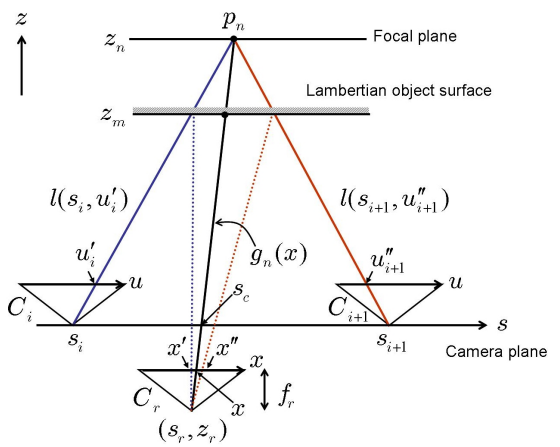
where w_i and w_{i+1} are weighting values that are determined based on the proximity of each sampled ray to the novel ray as below:

$$w_i = \left(1 - \frac{|s_c - s_i|}{\Delta s}\right) \text{ and } w_{i+1} = \left(1 - \frac{|s_c - s_{i+1}|}{\Delta s}\right) \qquad (2)$$

Note that $w_i + w_{i+1} = 1$ holds.



(a) The focal is equal to the depth of object surface.



(b) The focal is not equal to the depth of object surface.

Figure 1: *Light field paprameterizations, rendering and modeling aliasing artifacts.*

3.2. Modeling aliasing artifacts

If the sampling density of the light rays in the camera plane is low (i.e., $1/\Delta s$ is low), the rendered novel view suffers from aliasing artifacts. Plenoptic sampling theory states that aliasing artifacts are caused by the overlap of spectra replicas of the sampled light field, the interval of which is given by $2\pi/\Delta s$, and that there is a trade-off between the sampling density and the depth resolution avaliable.

In this section, we analyze the aliasing artifacts in the spatial domain and model them as spatially varying filters. Consider a scene with an object plane at depth z_m and assume the surface of the object is lambertian. If the focal depth z_n is equal to the actual object depth z_m, the novel view $g_m(x)$ is rendered based on the LFR method in Equation (1) with a depth z_n and simply given by

$$\begin{aligned} g_m(x) &= l(s_i,u_i) \\ &= l(s_{i+1},u_{i+1}) \end{aligned} \tag{3}$$

because $l(s_i,u_i) = l(s_{i+1},u_{i+1})$. If the focal depth z_n is not equal to z_m, the novel ray $g_n(x)$ at the given pixel x is rendered using different light rays due to pixel miscorrespondence on the object surface as shown in Figure 1(b), and is given by

$$g_n(x) = w_i l(s_i,u_i') + w_{i+1} l(s_{i+1},u_{i+1}''), \tag{4}$$

where u_i' and u_{i+1}'' are the pixel positions corresponding to point p_n at focal depth z_n (see Figure 1(b)). From Figure 1(b), and Equation (3), we find that $l(s_i,u_i')$ and $l(s_{i+1},u_{i+1}'')$ can be expressed by the pixel values (rays) on the novel view that is rendered when the focal depth is z_m, which would be $g_m(x')$ and $g_m(x'')$ in Figure 1(b). Therefore, $g_n(x)$ is expressed as

$$g_n(x) = w_i g_m(x') + w_{i+1} g_m(x''). \tag{5}$$

This means that the novel view that is rendered by LFR using the incorrect object depth is a filtered version of the novel view that is rendered by LFR with the actual depth. We show that aliasing artifacts can be modeled as a filter whose coefficients are the weighting values w_i and w_{i+1}. This filter is linear and shift varying (i.e., it changes depending on the pixel coordinate x), since x' and x'' change with x. The filter varies with the virtual view point and the focal depth as well.

3.3. Layered representation

In the first step of our proposed method, we render multiple novel views using the conventional LFR with different focal depths. In this section, we model the multiple views using a linear combination of textures at different depths with different aliasing artifacts, which are modeled by the filtering as analyzed in Section 3.2. We also model a desired all in-focus view and then formulate the reconstruction problem in the second step of our method.

We assume that the object's surface in the scene can be approximated by a set of planes at N different depths z_n ($n = 1,2,...,N$). For a given view point, we first define the n'th texture as

$$f_n(x,y) \stackrel{\text{def}}{=} \begin{cases} f(x,y), & \text{if } d(x,y) = z_n \\ 0, & \text{otherwise} \end{cases}, \quad \text{for } n = 1,2,...,N, \tag{6}$$

where $f(x,y)$ is the ideal all in-focus view that we want to reconstruct and $d(x,y)$ denotes a depth map from the novel view point. In other words, texture $f_n(x,y)$ is defined as an image that has an intensity value only in the regions of depth z_n that are visible from the novel view point. Note that the true depth map of the scene and the textures $f_n(x,y)$ are unknown.

Second, letting $g_n(x,y)$ be the novel views that are rendered by LFR with focal depths z_n $(n = 1,2,...,N)$, we model $g_n(x,y)$ as a linear combination of the textures $f_n(x,y)$ filtered with the corresponding aliasing artifacts at the depth z_n as follow:

$$\begin{cases} g_1 &= f_1 + h_{12} \star f_2 + h_{13} \star f_3 + \cdots + h_{1N} \star f_N \\ g_2 &= h_{21} \star f_1 + f_2 + h_{23} \star f_3 - \cdots + h_{2N} \star f_N \\ & \vdots \\ g_N &= h_{N1} \star f_1 + \cdots + h_{NN-1} \star f_{N-1} + f_N. \end{cases} \quad (7)$$

where h_{nm} is the filter that causes aliasing artifacts on the m'th texture $(m = 1,2,...,N)$, as described in the previous section, and "$\star$" means a filtering operation. Note that h_{mm} is an identity operation. This linear combination model has been used for representing multi-focus images captured with a physical camera in an all-focus image fusion [KA00]. In the model in Equation (7), however, spatially varying filters are used unlike defocus that is space invariant low-pass filter.

The desired all in-focus view is simply modeled as the sum of the textures without any artifacts as below:

$$f = f_1 + f_2 + \cdots + f_N. \quad (8)$$

The reconstruction problem in the second step of our method is formulated as the problem of solving a set of linear equations in Equation (7) for f_n and reconstructing f in Equation (8), given g_n and h_{nm}.

3.4. Iterative reconstruction

The filters h_{nm} are spatially varying and their inverse filter cannot be uniquely obtained; therefore, it is hard to inversely solve a set of linear equations in Equation (7) for each f_n. In this paper, we present an iterative method for solving those equations without calculating the inverse of the filters.

By solving each equation for f_n, we rewrite Equation (7) as:

$$\begin{cases} f_1 &= g_1 - h_{12} \star f_2 - h_{13} \star f_3 - \cdots - h_{1N} \star f_N \\ f_2 &= g_2 - h_{21} \star f_1 - h_{23} \star f_3 - \cdots - h_{2N} \star f_N \\ & \vdots \\ f_N &= g_N - h_{N1} \star f_1 - h_{N2} \star f_2 - \cdots - h_{NN-1} \star f_{N-1}. \end{cases} \quad (9)$$

Let $\{f_1^{(0)}, f_2^{(0)}, ..., f_N^{(0)}\}$ be a set of the initial solutions. First, we substitute it into the first equation in Equation (9) to update $f_1^{(0)}$ to $f_1^{(1)}$ as below:

$$f_1^{(1)} = g_1 - h_{12} \star f_2^{(0)} - h_{13} \star f_3^{(0)} - \cdots - h_{1N} \star f_N^{(0)}. \quad (10)$$

Second, we substitute the updated set of solutions $\{f_1^{(1)}, f_2^{(0)}, ..., f_N^{(0)}\}$ into the second equation in Equation (9) to update $f_2^{(0)}$ to $f_2^{(1)}$ as below:

$$f_2^{(1)} = g_2 - h_{21} \star f_1^{(1)} - h_{23} \star f_3^{(0)} - \cdots - h_{2N} \star f_N^{(0)}. \quad (11)$$

Similarly, the obtained new set of solutions is substituted into the n'th equation in Equation (9) to update $f_n^{(0)}$ to $f_n^{(1)}$. The updated solution is immediately subtituted into the next equation. The k'th solutions are given by:

$$\begin{cases} f_1^{(k)} &= g_1 - h_{12} \star f_2^{(k-1)} - h_{13} \star f_3^{(k-1)} \cdots - h_{1N} \star f_N^{(k-1)} \\ f_2^{(k)} &= g_2 - h_{21} \star f_1^{(k)} - h_{23} \star f_3^{(k-1)} \cdots - h_{2N} \star f_N^{(k-1)} \\ & \vdots \\ f_N^{(k)} &= g_N - h_{N1} \star f_1^{(k)} - h_{N2} \star f_2^{(k)} \cdots - h_{NN-1} \star f_{N-1}^{(k)}. \end{cases} \quad (12)$$

It should be noted that any local processing such as segmentation to find the in-focus regions or detection of the correct depth are not performed in this algorithm.

4. Results and discussions

4.1. Simulation results for synthetic test images

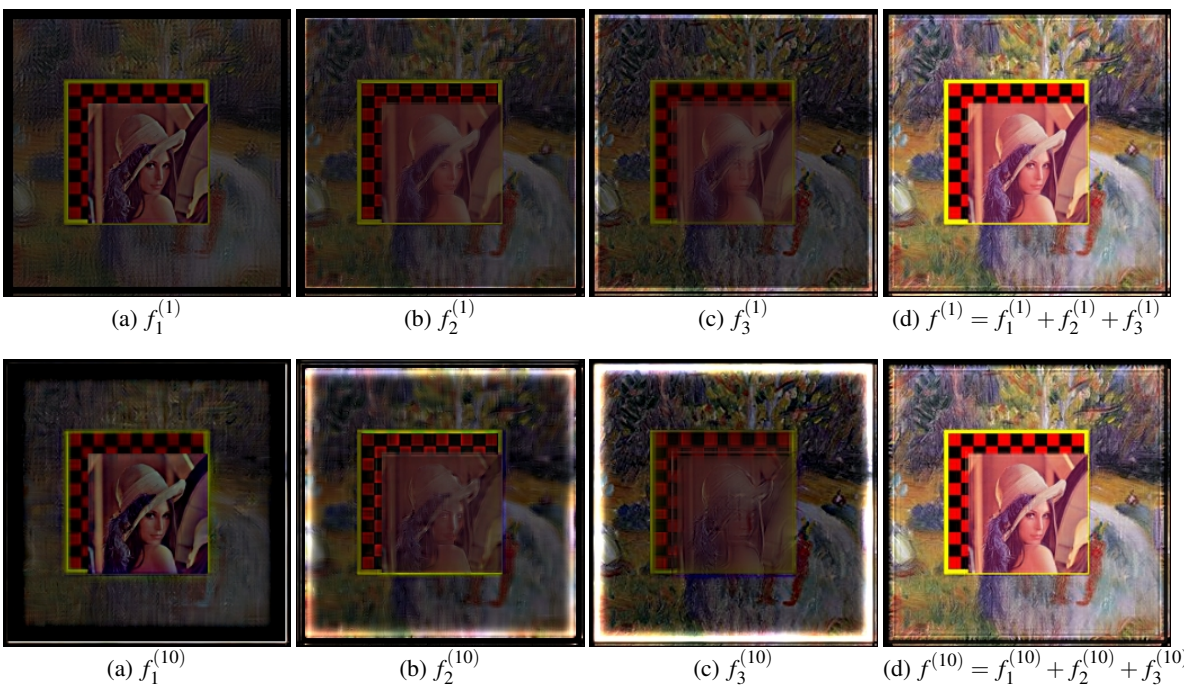
We tested the performance of our algorithm using synthetic images. We created 64 images with an 8x8 camera array for a scene consisting of three layers at different depths (500, 625 and 1000) [mm]. The foreground is "lena", the middle plane object is a checker board and the background is a painting, as shown in Figure 2. Image resolution is 256x256 pixels and the distance between the cameras is set to 10 [mm] in both the horizontal and vertical directions. Figure 2 (a), (b) and (c) show the novel views g_1, g_2 and g_3 at (s_r, t_r, z_r) = (35, 35, -100) [mm] (i.e. behind from the center of the camera array by 100 [mm]) rendered with conventional LFR at different focal depths, z_1 =500, z_2 =625 and z_3 =1000 [mm], which are the exact depths of the three object planes. Although the regions in the focal plane appear in focus, the regions not in the focal plane appear blurry and contain ghosting artifacts.

Figure 3 shows the solved textures at the focal depths (i.e., f_1, f_2, f_3) and the final reconstructed result (i.e., $f = f_1 + f_2 + f_3$) by the proposed method after the first (at the top of the figure) and tenth (at the bottom of the figure) iterations. We set the initial solutions $\{f_1^{(0)}\ f_2^{(0)}\ f_3^{(0)}\}$ to $\{g_1/3, g_2/3, g_3/3\}$. Therefore, the mean value of each texture is roughly equal to 1/3 that of the final view, resulting in the luminance value of all textures becoming darker than the final view. In each solved texture, the regions within the corresponding focal depth appear in focus and sharp. The regions that are not at the focal depth are blurry and more blurry in the tenth solutions than in the first ones.

In the final solution f, the focused regions of the three views in Figure 2 are well fused, even though each obtained texture f_n does not purely consist of the corresponding texture at the focal depth. Our goal is not to segment the image nor to estimate the depth map, but to reconstruct an all in-focus view without aliasing artifacts. Errors in each texture are well cancelled in the final result. This is analyzed in Section 4.3. The tenth iteration result is reconstructed slightly with better quality than the first result, although the difference is not visible.

(a) g_1 (the focal depth is on the near object plane) (b) g_2 (the focal depth is on the middle object plane) (c) g_3 (the focal depth is on the far object plane)

Figure 2: *Novel views reconstructed from the virtual view point (35,35,-100) [mm], synthesized by the conventional light field rendering with different constant depths. The regions appear in focus when the depth of the focal plane is on their corresponding depth. Aliasing artifacts are observed in the regions not in the focal plane.*



(a) $f_1^{(1)}$ (b) $f_2^{(1)}$ (c) $f_3^{(1)}$ (d) $f^{(1)} = f_1^{(1)} + f_2^{(1)} + f_3^{(1)}$

(a) $f_1^{(10)}$ (b) $f_2^{(10)}$ (c) $f_3^{(10)}$ (d) $f^{(10)} = f_1^{(10)} + f_2^{(10)} + f_3^{(10)}$

Figure 3: *The novel views at (35,35,-100) [mm] reconstructed by our proposed method using the three views in Figure 2. Top: the results after 1 iteration. Bottom: the results after 10 iteration. (a), (b) and (c) show the textures at each focal depth solved by our method after 1 and 10 iterations. (d) shows the final reconstructed view where all three regions appear in focus and the ghosting artifacts are strongly suppressed.*

4.2. Optimal arrangement of focal planes

In the above simulation, we set the focal planes at the depths of the three objects. In general, we assume that the minimum and the maximum depth of the scene are given, but we do not know the depths of the intermediate objects of the scene. Therefore, we have to consider the best arrangement and the number of focal planes in order to reconstruct the best result. Plenoptic sampling theory dictates that one focal plane can cover the depth range where the disparity of the elements is less than 1 pixel. It follows that we should arrange the focal planes such that they divide the disparity space equally with an interval less than 2 pixels as follow:

$$z_n = \left[\frac{1}{z_{min}} - \left(\frac{1}{z_{min}} - \frac{1}{z_{max}} \right) \frac{n-1}{N-1} \right]^{-1}, \qquad (13)$$

where N is the number of focal planes. (We have to consider the term z_r in Equation (13) for exact calculations; we ig-

(a) $N = 2$ (b) $N = 3$ (c) $N = 5$ (d) $N = 10$

Figure 4: *Effect of choosing the number of focal planes. The novel views are reconstructed by our method after 10 iterations using different numbers of focal planes.*

nore this term here.) Letting D_{max} be the difference between the minimum and the maximum disparities that are observed between the adjacent reference cameras for the target object, we can approximately determine N as $\lceil D_{max}/2 \rceil$.

We test the effect of the number of focal planes for the same test images in the previous section. We reconstruct the novel views by our method using 2, 3, 5, and 10 focal planes based on Equation (13). The results after 5 iterations are shown in Figure 4. In this test scene, D_{max} is 5 pixels, so, N is determined as 3. Thus, three or more focal planes are needed. In the case N =2 (Figure 4(a)), two focal planes are arranged at the foreground and the background; therefore, their textures appear in focus. However, the middle object plane has artifacts due to the lack of the number of focal planes. The results in the cases using 3 or 5 focal planes (Figures 4(b) and (c)) are sharply reconstructed as expected, even in the middle object region, even though no focal plane is exactly located at the middle depth in either test case. In the results using 10 focal planes, it can be seen that ringing artifacts in the occluded boundaries and textures are undesirably emphasized. Increasing the number of focal planes requires much more computation, resulting in an increase of the accumulated errors, which are mainly caused by the modeling error due to occlusion and the interpolation error in the (u, v) plane. The above results suggest that the optimal number of focal planes is $N = 3$.

4.3. Error analysis

In this section, we discuss the convergence of the proposed iterative algorithm by analyzing the errors of $f_n^{(k)}$ and $f^{(k)}$. Since it is hard to mathematically show the proof of the convergence of the proposed method for arbitrary input images and an arbitrary number of focal planes, we show the convergence of the errors by using given initial error signals for the case of a three-depth scene. The parameters used in this simulation are the same as those in the Section 4.1.

Let $e_n^{(k)}$ $(n = 1,2,3)$ and $e^{(k)}$ be the errors of $f_n^{(k)}$ $(n = 1,2,3)$ and $f^{(k)}$, respectively; they are defined as

$$e_n^{(k)} = f_n^{(k)} - \tilde{f}_n, \quad (n = 1,2,3) \tag{14}$$

$$e^{(k)} = f^{(k)} - \tilde{f}, \tag{15}$$

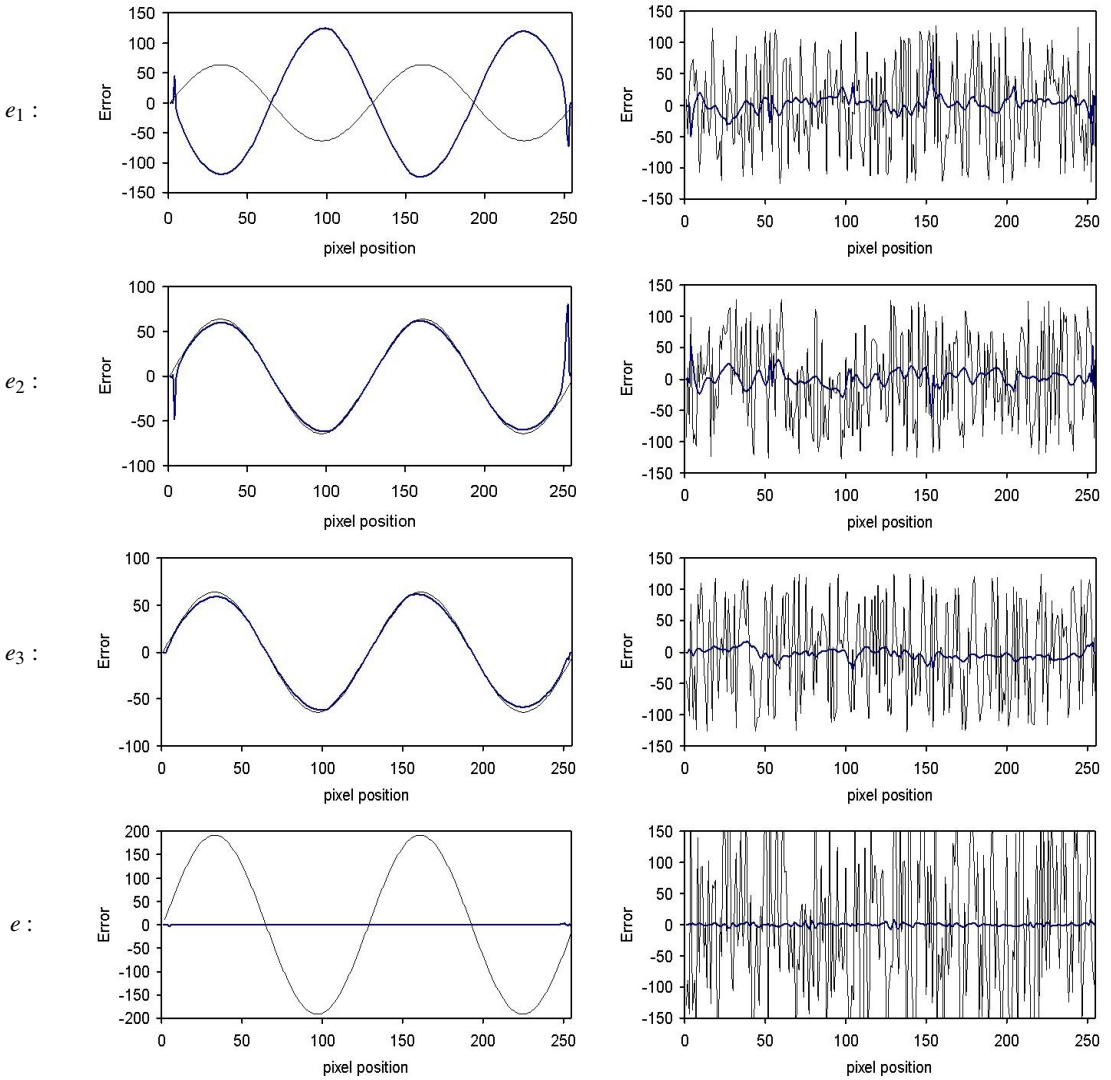
where the $\tilde{f}_n$ are the corresponding true textures, and $\tilde{f}$ is the all in-focus view, and $e^{(k)} = e_1^{(k)} + e_2^{(k)} + e_3^{(k)}$ holds. The formulation of each error at the k'th iteration can be derived from Equations (9) and (12). Since the true texture images $\tilde{f}_n$ satisfy the set of equations in (9), we substitute them into it. By subtracting the obtained equations from the equations in (12) on both sides, we get

$$\begin{cases} e_1^{(k)} = -h_{12} \star e_2^{(k-1)} - h_{13} \star e_3^{(k-1)} \\ e_2^{(k)} = -h_{21} \star e_1^{(k)} - h_{23} \star e_3^{(k-1)} \\ e_3^{(k)} = -h_{31} \star e_1^{(k)} - h_{32} \star e_2^{(k)}. \end{cases} \tag{16}$$

This shows that the errors do not depend on the scene texture. We simulated the errors using sinusoidal and random signals as the initial errors $e_1^{(0)}$, $e_2^{(0)}$ and $e_3^{(0)}$. The results are shown in Figure 5 for the luminance values along the horizontal line at y =100; thin lines indicate the initial error and the heavy line indicates the error after the tenth iteration. In the case of the sinusoidal error (Figure 5(a)), although the convergence of each error e_n is very slow, error e after the tenth iteration has almost converged to zero. In the case of the random error (Figure 5(b)), each error e_n is reduced but is still significant. Nevertheless, after the tenth iteration, the error e has almost converged to zero as well. Note that in our algorithm, since $e_1^{(0)}$ does not affect any error, $e_1^{(10)}$ may be larger than $e_1^{(0)}$ as shown in Figure 5(a). Those results shows that even though each error e_n does not converge to zero, the total error e rapidly converges to zero; therefore, the desired all in-focus view can be reconstructed with little error. This is another significant advantage of our algorithm.

4.4. Results for real images

We used 81 real images captured with a 9x9 camera array, which are provided from "The Multiview Image Database,"



(a) Sinusoidal error (low frequency) (b) Random error (high frequency)

Figure 5: *Error analysis on a single scanline of the test image using (a) sinusoidal and (b) random signals as initial errors. Thin line: initial error; Heavy line: error after 10 iterations. Top to bottom: e_1, e_2, e_3 and e*

courtesy of the University of Tsukuba, Japan. Image resolution is 480x360 pixels and the distance between cameras is 20 [mm] in both the horizontal and vertical directions. The scene contains an object ("Santa Claus doll") in the depth range of 590–800 [mm], which is the target depth range in this experiment. The maximum and minimum disparities of the object between adjacent cameras are about 36 and 26 pixels, respectively, so that the maximum difference of the disparities is about 10 pixel. From Plenoptic sampling theory, this shows that the sampling density is lower, or the distance between cameras is more sparse by about 5 times than that required for anti-aliased LFR.

Figure 6(a) shows the novel views reconstructed by the conventional LFR with the corresponding optimal depth at 5 different view points. In our experiment, we assume that the view direction is the depth direction, i.e., perpendicular to the camera plane. The optimal depth z_{opt} is calculated as

$$z_{opt} = 2\left(\frac{1}{z_{min}} + \frac{1}{z_{max}}\right)^{-1}, \tag{17}$$

where z_{min} and z_{max} are the minimum and maximum depths of the target scene. In Figure 6(a), the face of the doll appears in focus, while other regions far from the face appear blurry or ghosted. The conventional LFR algorithm cannot recon-

struct all in-focus views at this sampling density. In other words, the depth of field of conventional LFR is too small to clearly render a novel view for a scene of this depth range.

The novel views reconstructed by the proposed method at the same view points are shown in Figure 6(b). It can be seen that all the regions of the object are reconstructed in focus without visible artifacts except ringing artifacts around the edges. This is due to the error in the occluded boundaries. In this reconstruction, we set five focal planes at different depths based on Equation (13), and render the novel views using LFR at those depths. Examples of the views are shown in Figure 6(c), from which the final view at the bottom of Figure 6(b) is reconstructed. From the top to the bottom in Figure 6(c), the focal depth is changed from near to far. Although many artifacts occur in the regions that are not in-focus, most of those artifacts cannot be observed in the final views.

The proposed rendeing method is very computational expensive, because it requires many iterative filtering operations in the spatial domain. It takes about 15 sec. to render a novel view when using 2GHz Pentium CPU. For the case when using 5 iterations and 5 depth layers, 100 times of filtering operations are required. We could reduce the rendering time by using texture mapping in the filtering operation.

5. Conclusions

We propose a novel IBR method for reconstructing all in-focus views where an aliasing artifact is much suppressed. In the proposed method, we model the multiple views and the desired all in-focus view as a set of linear equations of unknown depth textures, and by using an iterative reconstruction method, we can effectively solve those equations without any local processing such as depth estimation. The advantage of the proposed method is that we can reduce the number of images needed for anti-aliased rendering. In addition, the presented method is feasible for implementation using texture mapping.

References

[BBM*01] BUEHLER C., BOSSE M., MCMILLAN L., GORTLER S., COHEN M.: Unstructured lumigraph rendering. In *Proc. SIGGRAPH '01* (2001), pp. 425–432.

[BK93] BURT P. J., KOLCZYNSKI R. J.: Enhanced image capture through fusion. In *Proc. IEEE International Conference on Computer Vision* (1993), pp. 173–182. 2

[CTCS00] CHAI J.-X., TONG X., CHANY S.-C., SHUM H.-Y.: Plenoptic sampling. In *Proc. SIGGRAPH '00* (2000), pp. 307–318. 1

[CW93] CHEN S., WILLIAMS L.: View interpolation for image synthesis. In *Proc. SIGGRAPH '96* (1993), pp. 279–288. 1

[GGSC96] GORTLER S. J., GRZESZCZUK R., SZELISKI R., COHEN M. F.: The lumigraph. In *Proc. SIGGRAPH '96* (1996), pp. 43–54. 1

[ILG99] ISAKSEN A., LEONARD M., GORTLER S. J.: Dynamically reparameterized light fields. In *MIT-LCS-TR-778* (1999). 2

[ILG00] ISAKSEN A., LEONARD M., GORTLER S. J.: Dynamically reparameterized light fields. In *Proc. SIGGRAPH '00* (2000), pp. 297–306. 2

[KA00] KUBOTA A., AIZAWA K.: Inverse filters for reconstruction of arbitrarily focused images from two differently focused images. In *Proc. IEEE International Conference on Image Processing* (2000), pp. 101–104. 2, 4

[LH96] LEVOY M., HANRAHAN P.: Light field rendering. In *Proc. SIGGRAPH '96* (1996), pp. 31–42. 1

[Nay92] NAYAR S. K.: Shape from focus system. In *Proc. IEEE Computer Vision and Pattern Recognition* (1992), pp. 302–308. 2

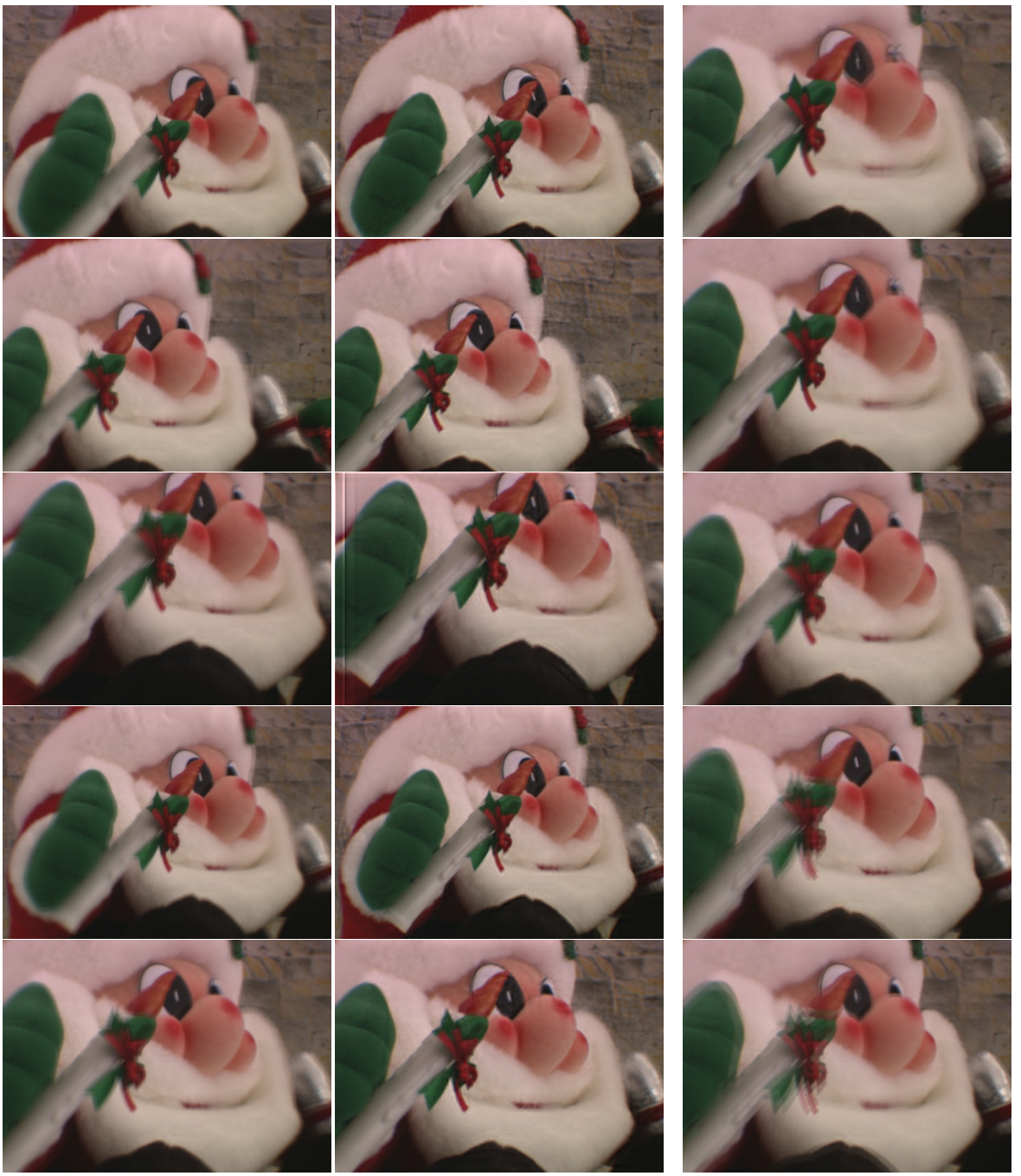
[SD96] SEITZ S. M., DYER C. M.: View morphing. In *Proc. SIGGRAPH '96* (1996), pp. 21–30. 1

[SH99] SHUM H., HE L.-W.: Rendering with concentric mosaics. In *Proc. SIGGRAPH '99* (1999), pp. 299–306. 1

[SHC03] SHUM H.-Y., HE S. B., CHAN S.-C.: Survey of image-based representations and compression techniques. *IEEE Trans. on Circuits and Systems for Video Technology 13*, 11 (Nov. 2003), 1020–1037. 1

[SYGM03] STEWART J., YU J., GORTLER S. J., MCMILLAN L.: A new reconstruction filter for undersampled light fields. In *Proc. of Eurographics Symposium on Rendering 2003* (2003), pp. 150–156. 1

[TKN03] TAKAHASHI K., KUBOTA A., NAEMURA T.: All in-focus view synthesis from undersampled light fields. In *Proc. VRSJ ICAT* (2003), pp. 249–256. 2



(a) (b) (c)

Figure 6: *Experimental results using real images. (a) and (b): Novel views at 5 different view points from top to bottom. The views in (a) are reconstructed using conventional LFR at the optimal depth. The views in (b) are reconstructed by the proposed method after 5 iterations, where 5 focal depths are set. (c): 5 novel views reconstructed by conventional LFR at 5 different focal depths, which are used for reconstructing the bottom view in (b) by our method.*