# VIRTUAL VIEW SYNTHESIS THROUGH LINEAR PROCESSING WITHOUT GEOMETRY

*Akira Kubota*[1] *Kiyoharu Aizawa*[2] *Tsuhan Chen*[3]

[1]High Tech. Research Center, Kanagawa University
[2]Dept. of Electrical Engineering, University of Tokyo
[3]Dept. of Electrical and Computer Engineering, Carnegie Mellon University
kubota@hal.t.u-tokyo.ac.jp, aizawa@hal.t.u-tokyo.ac.jp, tsuhan@cmu.edu

**ABSTRACT**

This paper presents a new approach for virtual view synthesis that does not require any information of scene geometry. Our approach first generates multiple virtual views at the same position based on multiple depths by the conventional view interpolation method. The interpolated views suffer from blurring and ghosting artifacts due to the pixel mis-correspondence. Secondly, the multiple views are integrated into a novel view where all regions are focused. This integration problem can be formulated as the problem of solving a set of linear equations that relates the multiple views. To solve this set of equations, two methods using projection onto convex sets (POCS) and inverse filtering are presented that effectively integrate the focused regions in each view into a novel view. Experimental results using real images show the validity of our methods.

## 1. INTRODUCTION

Traditional geometry-based approaches to virtual view synthesis using reference images taken with multiple cameras recover the scene geometry (3D models) and render the novel view based on the geometry. The recovery of the scene geometry involves computationally expensive and complicated processing steps such as feature extraction and matching, etc. and it is generally hard to achieve appropriate quality for the real scene. An alternative approach is image-based rendering (IBR) [1], which does not require geometry information. However, hundreds of reference images taken by densely arranged cameras are necessary for rendering with adequate quality.

In this paper, we present a novel view synthesis method without estimating scene geometry information. The presented method allows more sparsely arranged cameras for capturing reference images compared with the conventional IBR. In our approach, we first assume multiple object planes at different depths in the scene and interpolate multiple novel views at the same fixed position based on the planes. In each interpolated view, although the region at the assumed depth appears in focus, the region far from the assumed depth has blurring and ghosting artifacts due to the pixel mis-correspondence between the reference images to be used for the interpolation. Secondly, from the multiple views, we reconstruct an all in-focus view using methods based on projection onto convex sets (POCS) [6] and inverse filtering.

Recently, two criteria have been presented to measure the sharpness (or focus) of a region in local for the purpose of extracting the focused region from the multiple views interpolated based on multiple depths. This is the same framework of classical image fusion for integrating an all in-focus image from multi-focus images; however, the artifact caused in the interpolated view differs from blur in that it contains both low and high frequency components. Isaksen et al. [2] have measured the smoothness (consistency) of the pixel values to be used for the interpolation at each depth. This idea is essentially equivalent to that underlying stereo matching. Takahashi et al. [3] have presented a stable focus measure using the difference of the views that are generated through different kinds of interpolation methods based on the same assumed object plane. Both approaches result in estimating the view dependent depth map.

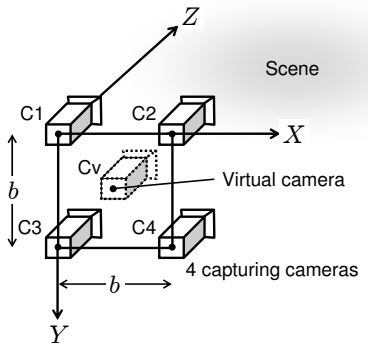



**Fig. 1**. Camera arrangement used in our method.

Our method presented in this paper can reconstruct an all in-focus view directly from the multiple interpolated views without depth map estimation. We model the multiple interpolated views and the desired all in-focus view as a set of linear equations with a combination of the textures at the assumed depths. We can solve this set of linear equations by using POCS in both the spatial and frequency domains. We also present an inverse filtering method that can reconstruct the all in-focus view in one shot in the frequency domain. These methods effectively integrate the focused regions in each view into an all in-focus view.

## 2. THE PROPOSED METHOD

The camera arrangement used in this paper is shown in Fig. 1. We capture 4 reference images with 4 cameras (C1, C2, C3 and C4) arranged on the $XY$ plane, with an interval of $b$ in parallel to the depth direction $Z$. Our goal is to synthesize a novel image at the virtual camera (Cv) located at the middle (i.e., at $(b/2, b/2)$) of the capturing cameras.

### 2.1. View interpolation based on multiple depths

In the first step of our method, we generate multiple views at the same virtual point by view interpolation based on multiple object

0-7803-8554-3/04/$20.00 ©2004 IEEE.

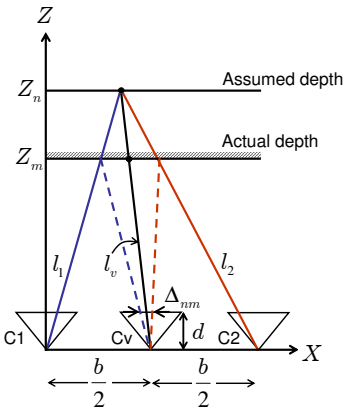



**Fig. 2**. Artifact due to pixel mis-correspondence in view interpolation based on multiple object planes.

planes at different depths in the scene. When using two cameras C1 and C2 on the $X$ axis and assuming the object depth to be $Z_n$ as shown in Fig. 2, we interpolate the pixel value $l_v$ on the virtual view as $(l_1 + l_2)/2$, that is the average of the corresponding pixel values $l_1$ and $l_2$ on the captured images, which are determined based on the assumed depth $Z_n$.

Assume that the scene consists of lambertian surface objects without occlusions. When the assumed depth $Z_n$ coincides with the actual object depth $Z_m$ ($Z_n = Z_m$), the same pixel values (i.e., $l_1 = l_2$) of the texture at depth $Z_n$ are used for interpolating the pixel value on the virtual image, resulting in $l_v = l_1 = l_2$. In the case of $Z_n \neq Z_m$, different pixel values (i.e., $l_1 \neq l_2$) of the texture at depth $Z_m$ are used for interpolation; thus, pixel mis-correspondence leads to blurring or ghosting artifacts on the interpolated image. The phenomenon of those artifacts was analyzed in the frequency domain from a sampling theory point of view [4]. We find that the texture at depth $Z_m$ in the virtual view interpolated based on the assumed depth $Z_n$ becomes the filtered version of that texture in the virtual view interpolated based on the actual depth $Z_m$. When using 4 reference images, we can model this filter as a 4-tap filter with all coefficients equal to 1/4:

$$\begin{aligned} h_{nm} = {} & 1/4 \cdot \delta(x - \Delta_{nm}/2, y - \Delta_{nm}/2) \\ & +1/4 \cdot \delta(x - \Delta_{nm}/2, y + \Delta_{nm}/2) \\ & +1/4 \cdot \delta(x + \Delta_{nm}/2, y - \Delta_{nm}/2) \\ & +1/4 \cdot \delta(x + \Delta_{nm}/2, y + \Delta_{nm}/2), \end{aligned} \quad (1)$$

where $\delta(\cdot,\cdot)$ indicates the 2-D Kronecker delta fuction and $\Delta_{nm}$ is the distance between the locations of the pixel values to be used for interpolation in the image plane of the virtual view (see Fig. 2) that is given by $bd|1/Z_m - 1/Z_n|$, where $d$ is the focal length of the virtual camera. It does not depend on the pixel location; hence the filter is spatially invariant.

### 2.2. Modeling with depth layers

Assuming the scene consists of $N$ depth layers at depth $Z_n$ $(n = 1, ..., N)$, we model the virtual view using a linear combination of textures at depth $Z_n$ that are visible from a virtual view point. Let $g_n$ be the virtual image interpolated based on the depth $Z_n$ and the texture at depth $Z_n$ be $f_n$. Based on the discussion in the previous section 2.1, we can model $g_n$ by

$$g_n = f_n + \sum_{m \neq n}^{N} h_{nm} * f_m, \ (n = 1, ..., N), \quad (2)$$

where $*$ is the 2-D convolution operation. The desired virtual image that we synthesize is modeled simply by the sum of the textures: $f = \sum_{m=1}^{N} f_m$. We formulate the virtual view synthesis problem as a problem of solving a set of linear equations (2). We have used this formulation using point spread fucntions as $h_{nm}$ for all in-focus image generation from multiple differently focused images [5].

### 2.3. POCS method

In the second step of our method, we reconstruct the desired all in-focus view $f$ from multiple virtual images $g_n$ $(n = 1, ..., N)$. In this section, we present a novel iterative reconstruction method using projection onto convex sets (POCS) [6] for recursively reconstructing each depth texture $f_n$.

We introduce $N$ constraint sets for a vector $\boldsymbol{f} = (f_1 \ f_2 \cdots f_N)$ using the models (2) themselves as

$$S_n = \left\{ \boldsymbol{f} : f_n = g_n - \sum_{m \neq n}^{N} h_{nm} * f_m \right\} \ (n = 1, ..., N) \quad (3)$$

Unlike the conventional image restoration methods using POCS, any additional sets are not necessary. Since all the sets $S_n$ are shown to be convex (see appendix), by iteratively projecting an arbitrary $\boldsymbol{f}$ onto the sets $S_n$, we can obtain a feasible solution in the intersection set of the sets $S_n$. Let $\boldsymbol{f}^{(0)}$ be the initial solution vector $(f_1^{(0)} \ f_2^{(0)} \cdots f_N^{(0)})$. First, we project it onto the set $S_1$ to update the element $f_1^{(0)}$ to $f_1^{(1)}$ as

$$f_1^{(1)} = g_1 - \sum_{m=2}^{N} h_{1m} * f_m^{(0)}, \quad (4)$$

and we obtain a new vector $(f_1^{(1)} \ f_2^{(0)} \cdots f_N^{(0)})$. Second, we project this vector onto the set $S_2$ to update the second element $f_2^{(0)}$ to $f_2^{(1)}$ and obtain a new vector $(f_1^{(1)} \ f_2^{(1)} \ f_3^{(0)} \cdots f_N^{(0)})$. Similarly, the new vector obtained is projected onto $S_n$ until the last element $f_N^{(0)}$ is updated, and the first iteration solution vector $\boldsymbol{f}^{(1)} = (f_1^{(1)} \ f_2^{(1)} \cdots f_N^{(1)})$ is obtained. Let the projection operator onto the set $S_n$ be $P_n$. In any $k$th solution vector, it is given by

$$P_n(f_1^{(k+1)} \cdots f_n^{(k)} \cdots f_N^{(k)}) = (f_1^{(k+1)} \cdots f_n^{(k+1)} \cdots f_N^{(k)}) \quad (5)$$

where $f_n^{(k)}$ is updated to $f_n^{(k+1)}$:

$$f_n^{(k+1)} = g_n - \sum_{m=1}^{n-1} h_{nm} * f_m^{(k+1)} - \sum_{m=n+1}^{N} h_{nm} * f_m^{(k)}. \quad (6)$$

We can find the $k$th iteration solution vector $\boldsymbol{f}^{(k)}$ by successive projection

$$\boldsymbol{f}^{(k)} = P_N P_{N-1} \cdots P_1 \boldsymbol{f}^{(k-1)} \quad (7)$$

The $k$th solution of the desired image $f$ is finally reconstructed as $f^{(k)} = \sum_{m=1}^{N} f_m^{(k)}$.

Whereas the conventional POCS methods have generally used non-linear projections, our method uses only linear projections so that it can be performed in the frequency domain using the Fourier transform.

### 2.4. Inverse method in the frequecny domain

In this section, we present the inverse method in the frequency domain that is derived from the direct solution of the set of equations in the frequency domain. The Fourier transformed version of the model (2) can be written by matrix notion as

$$\boldsymbol{G} = \boldsymbol{H}\boldsymbol{F} \tag{8}$$

where

$$\boldsymbol{G} = \begin{pmatrix} G_1 \\ G_2 \\ \vdots \\ G_N \end{pmatrix}, \quad \boldsymbol{F} = \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_N \end{pmatrix}, \quad \boldsymbol{H} = \begin{pmatrix} 1 & H_{12} & \cdots & H_{1N} \\ H_{21} & 1 & \cdots & H_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ H_{N1} & H_{N2} & \cdots & 1 \end{pmatrix}.$$

The function with capital letters denote the Fourier transforms (FTs) of the respective functions. The FT of the desired all in-focus virtual view, $F$, can be given by

$$F = \boldsymbol{I}^T \boldsymbol{H}^{-1} \boldsymbol{G}, \tag{9}$$

where $\boldsymbol{I} = (1\ 1 \cdots 1)^T$. The coefficient $\boldsymbol{I}^T\boldsymbol{H}^{-1}$ corresponds to the vector matrix consisting of inverse filters for the multiple views, say $\boldsymbol{K} = (K_1 K_2 \cdots K_N)$. In our previous method [5] that have been applied for generating an all in-focus image from multiple focused images captured with physical camera; we have shown that the inverse filters for multiple focused images uniquely exist for two depths scene. In the case of all in-focus view generation from the multiple views that we deal with in this paper, the inverse of $\boldsymbol{H}$ does not exist at some frequencies, since the determinant of $\boldsymbol{H}$, $|\boldsymbol{H}|$, become zero at some frequencies. In this paper, setting the threshold value $\theta$, we determine the inverse filters as $K_n = 1/N$, if $-\theta < |\boldsymbol{H}| < \theta$.

## 3. EXPERIMENTAL RESULTS ON REAL IMAGES

We test the performance of our algorithm using real images (320 x 280 pixels) as shown in Fig. 3, which are 4 images of the multiview image database courtesy of the University of Tsukuba, Japan. Four images were captured from 4 different camera positions with baseline length (distance between cameras) $b$ of 20 [mm] for a scene of an object ("Santa Claus doll"). The depth range of the object is 590–800 [mm]. The maximum and the minimum disparities of the object are about 23 and 17 pixels, respectively. The expanded images (50 x 50 pixel) of the same part in the captured images are also shown in Fig. 6 (a). We can see that different disparities and occlusions are observed among those region of the captured images.

Three different object planes for view interpolation are assumed at $Z_1$=590, $Z_2$=680, and $Z_3$=750 [mm]. The three virtual views at the center of the 4 camera positions were generated through the view interpolation based on the three depths, which are $g_1$, $g_2$ and $g_3$ as shown in Fig. 4. The region near the assumed depth appears in focus, while the regions far from the assumed depth appear blurry or ghosted.

Figure 5 shows the reconstructed virtual views by the presented methods using POCS in the spatial and frequency domain and using inverse filtering. In the POCS methods, we used a vector $(g_1\ 0\ 0)$ as the initial solution vector $\boldsymbol{f}^{(0)}$ (= $(f_1^{(0)} f_2^{(0)} f_3^{(0)})$). In the inverse method, we set $\theta$ as 0.1. $\theta$ needs to be optimized according to the assumed depths. In the reconstructed views, all regions of the target object are integrated in focus. The results of the POCS method in the frequency domain suffer from the artifact


**Fig. 3**. The reference images (320 x 240 pixel) captured by 4 cameras located with interval of 20 [mm].

of the amplification of certain frequency components that make $|\boldsymbol{H}|$ = 0, in which case since all the convex sets become same, the results are affected by noise. The result of the inverse method is affected by the threshold process in determining the inverse filters. This is clearly seen in the comparison of the expanded images between the original reference images and the reconstructed images as shown in Fig 6. The POCS method in the spatial domain performs well and the view is reconstructed with proper parallax and no visible artifacts.

## 4. CONCLUSION AND FUTURE WORK

We presented a new approach to virtual view generation that does not require the recovery of the scene geometry. Assuming multiple object planes at different depths, we interpolate multiple virtual views based on their depths and integrate them through the POCS method and the inverse method into an all in-focus virtual view. The experimental results on real images show that the POCS method in the spatial domain performs well and can reconstruct the novel view without visible artifact.

One application of our method is for teleconferencing or videophone systems with eye contact using a virtual view synthesis. Using four images captured by cameras at the four corner of the monitor, we can synthesis the virtual image at the center of the monitor where the full face view is synthesized for eye contact. For this application, since the distance between cameras is large, we need to analyze the limitations of our approach in terms of the distance between cameras, the target depth range, and the quality of the virtual view.

## A. THE PROOF OF THE CONVEXITY OF THE SET $S_n$

Given two arbitrary vectors $\boldsymbol{f}'$ and $\boldsymbol{f}''$ that belong to the set $S_n$: $\boldsymbol{f}' = (f_1'\ f_2' \cdots\ f_N') \in S_n$ and $\boldsymbol{f}'' = (f_1''\ f_2'' \cdots\ f_N'') \in S_n$, whose elements satisfy

$$f_n' = g_n - \sum_{m \neq n}^{N} h_{nm} * f_m' \text{ and } f_n'' = g_n - \sum_{m \neq n}^{N} h_{nm} * f_m''. \tag{10}$$

$g_1$ $g_2$ $g_3$

**Fig. 4**. The interpolated virtual views based on the different depths

POCS method in the spatial domain | POCS method in the frequency domain | Inverse filtering in the frequency domain

**Fig. 5**. The reconstructed all in-focus virtual view by our methods.

Consider a vector $\boldsymbol{f}'''$ that is a linear combination of $\boldsymbol{f}'$ and $\boldsymbol{f}''$: $\boldsymbol{f}''' = \alpha\boldsymbol{f}' + (1-\alpha)\boldsymbol{f}''$, where $0 \leq \alpha \leq 1$. The proof of the convexity of $S_n$ can be given by showing $\boldsymbol{f}''' = (f_1''' \; f_2''' \cdots \; f_N''') \in S_n$. The $n$th element of $\boldsymbol{f}'''$, $f_n'''$, is calculated as below from eq.(10) and the fact that $h_{nm}$ is a linear operator.

$$f_n''' = \alpha f_n' + (1-\alpha) f_n'' = g_n - \sum_{m \neq n}^{N} h_{nm} * (\alpha f_m' + (1-\alpha) f_m'')$$

$$= g_n - \sum_{m \neq n}^{N} h_{nm} * f_m''' \; (n = 1, ..., N). \qquad (11)$$

This means $\boldsymbol{f}''' \in S_n$ and therefore the set $S_n$ is convex.

## 5. REFERENCES


[1] H-Y. Shum, S. B. He, and S-C. Chan, "Survey of Image-Based Representations and Compression Techniques", IEEE Trans. on CSVT Vol. 13, No. 11, pp. 1020-1037, 2003

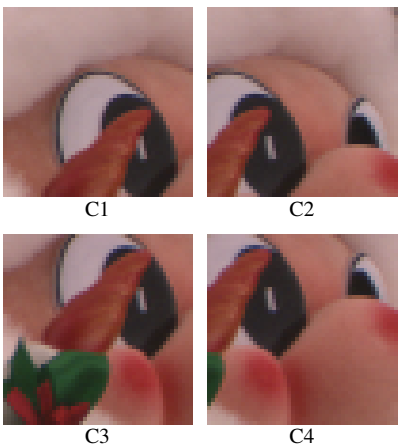
[2] A. Isaksen, M. Leonard, S. J. Gortler "Dynamically Reparameterized Light Fields," MIT-LCS-TR-778, 1999

[3] K. Takahashi, A. Kubota, T. Naemura "All in-Focus View Synthesis from Under-Sampled Light Fields," Proc. VRSJ ICAT, pp. 249–256, 2003

[4] Jin-Xiang Chai, X. Tong, S.-C. Chany and H.-Y. Shum, "Plenoptic Sampling," SIGGRAPH2000, pp.307-318, 2000

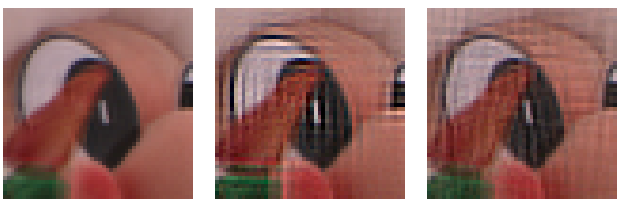
[5] A.Kubota, K.Aizawa "Inverse filters for reconstruction of arbitrarily focused images from two differently focused images," proc. in ICIP2000 Vol.I, pp.101-104, 2000

[6] D. C. Youla and H. Webb, "Image restoration by the method of convex projections: part I - theory," IEEE trans. on MI, Vol. MI-1, pp. 81-94, 1982

C1 C2 C3 C4

(a) Four reference images



POCS method in the spatial domain | POCS method in the frequency domain | Inverse filtering in the frequency domain

(b) Reconstructed novel views by three different methods

**Fig. 6**. The expanded versions of the reference images and the reconstructed all in-focus views.