

# Regarding Multi Plane Image as a Layered Light Field Display

**Chisaki Sato, Chihiro Tsutake, Keita Takahashi, Toshiaki Fujii**

Corresponding Author: Chisaki Sato (c.sato@fujii.nuee.nagoya-u.ac.jp)

Department of Information and Communication Engineering, Graduate School of Engineering, Nagoya University, Japan

Keywords: layered light-field display, light field, multi plane image, multi-view images

## ABSTRACT

We demonstrate that a multi-plane image (MPI) can be regarded as a layered light-field display. We also present an iterative method for obtaining an MPI from a light field. We finally present performance comparisons between MPIs and layered displays.

## 1 Introduction

A multi plane image (MPI) [1, 2, 3] is a volumetric representation of a 3-D scene used for computer graphics applications. An MPI is composed of a stack of semi-transparent images, and its appearance varies continuously along the observed direction. Meanwhile, similar layer structures have also been used for light-field displays [4, 5], where the layers are physically implemented using liquid crystal display (LCD) panels or holographic optical elements (HOEs). However, the relation between them has rarely been discussed so far.

In this paper, we first analyze the relation between MPIs and layered light-field displays, and demonstrate that an MPI itself can be regarded as a layered light-field display. We then introduce a new iterative method for obtaining an MPI representation from a light field, which yields better light-field reconstruction quality than the learning-based counterpart. We finally present performance comparisons between MPIs and layered displays.

### 1.1 Notations

Each of the light rays traveling in 3-D  $x$ - $y$ - $z$  space is parameterized with  $\mathbf{p} = (s, t, u, v)$ , where  $(u, v)$  denotes the intersection point with  $z = 0$ , and  $(s, t)$  denotes the outgoing direction (the tangent against  $z$  axis). We use  $\mathbf{q}_p^z = (u - sz, v - tz)$  to indicate the point where a light ray represented with  $\mathbf{p} = (s, t, u, v)$  intersects with a plane located at  $z$ . We assume that the layers (for both of MPIs and layered displays) are located perpendicular to  $z$ -axis.

## 2 Relating MPI and Layered Display

We first mention the principle of layered light-field displays. Depending on the material of a layer device, different operations are carried out for the light rays that pass through the layer. As shown in Figs. 1(a) and (b), we consider two

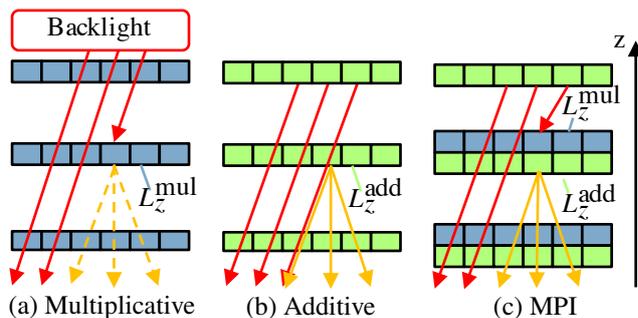


Fig. 1: Structure of layered displays and MPI

types of layers: multiplicative layers (LCD panels) [4] and additive (HOEs) layers [5]. With these layers stacked along  $z$  axis, the light rays are physically modulated as the result of operations along the paths, described respectively as

$$I_p^{\text{mul}} = \prod_{z \in Z} L_z^{\text{mul}}(\mathbf{q}_p^z), \quad (1)$$

$$I_p^{\text{add}} = \sum_{z \in Z} L_z^{\text{add}}(\mathbf{q}_p^z). \quad (2)$$

We can see that a light field is displayed by each of them; the set of light rays generated from the display constitutes a light field (a set of multi-view images), where  $(s, t)$  and  $(u, v)$  corresponds to the viewpoint (viewing direction) and pixel position, respectively.

Meanwhile, the light rays from an MPI [1, 2, 3] are computationally rendered by alpha blending operations along the paths, which is described as

$$I_p^{\text{MPI}} = \sum_{z \in Z} c_z(\mathbf{q}_p^z) \alpha_z(\mathbf{q}_p^z) \prod_{\substack{z' \in Z \\ z' < z}} \{1 - \alpha_{z'}(\mathbf{q}_p^{z'})\}, \quad (3)$$

where  $c_z$  and  $\alpha_z$  denote color and alpha (transmittance) channels of each layers. We assume that the rightmost factor with a product symbol returns 1 for the foremost layer where  $z' \in Z, z' < z$  is empty.

We here derive the relation between an MPI and layered light-field displays. By substituting  $L_z^{\text{add}} = c_z \alpha_z$  and  $L_z^{\text{mul}} = 1 - \alpha_z$ , we can rewrite Eq. (3) as

$$I_p^{\text{MPI}} = \sum_{z \in Z} L_z^{\text{add}}(\mathbf{q}_p^z) \prod_{\substack{z' \in Z \\ z' < z}} L_{z'}^{\text{mul}}(\mathbf{q}_p^{z'}). \quad (4)$$

Table 1: Experimental methods

	Iterative	CNN
Add	Maruyama [6]	Table. 2
Mul		
MPI	Section. 3	

This equation clearly shows that an MPI can be regarded as a layered display where additive and multiplicative layers are alternately stacked along  $z$  axis, as illustrated in Fig. 1(c).

We finally mention the advantage of an MPI for handling occlusions. As shown in Fig. 1(a), a multiplicative layer can block the light rays coming behind of it with the transmittance value set to 0. However, once a light ray is blocked at a depth, there is no way to produce non-zero luminance. Meanwhile, as shown in Fig. 1(b) an additive layer can generate a new luminance point at any depth, but cannot nullify the light rays coming behind of it. In contrast, an MPI layer can do the both as shown in Fig. 1(c); it can block the light rays coming behind it, and produce a new luminance point at any depth. This capability is advantageous for handling occlusions, where an foreground object not only blocks the light rays from the background, but also produces new light rays from its own surface.

### 3 Iterative Method for MPI

We discuss how to obtain an MPI representation from a target light field (a set of multi-view images). For this purpose, learning-based methods were adopted in previous works [1, 2, 3], where a neural network, which was pre-trained on the training dataset, was utilized to find an MPI for the given images of the target 3-D scene. Meanwhile, in the context of layered light-field displays, both iterative [4, 5, 6] and learning-based [6] approaches have been investigated. In this section, we derive an iterative method for MPIs, following the formulation by Maruyama et al. [6].

We define  $\mathbf{I}^{\text{MPI}}$ ,  $\mathbf{c}_z$  and  $\alpha_z$  as the vectors that contain all the elements of  $I^{\text{MPI}}$ ,  $c_z$  and  $\alpha_z$ , respectively. Given the light field of a target 3-D scene,  $\mathbf{I}$ , the goal of optimization is to minimize the error between the target light field and the one produced from an MPI.

$$\min_{\{\mathbf{c}_z\}\{\alpha_z\}} \|\mathbf{I} - \mathbf{I}^{\text{MPI}}\|^2. \quad (5)$$

Since Eq. (5) is a non-convex problem, we resort to an alternative optimization, where the layers  $(\mathbf{c}_z, \alpha_z)$  for all  $z$  are initialized randomly, and then, updated alternatively until convergence.

Let us consider a case where a layer  $\mathbf{c}_z$  for a specific depth  $z$  are updated while others are fixed to the current values. Using Eq. (3), the light field generated from the MPI is described as  $\mathbf{I}^{\text{MPI}} = \mathbf{A}_z \mathbf{c}_z + \mathbf{b}_z$ . Here, the effect from the other layers than  $\mathbf{c}_z$  is absorbed by a matrix  $\mathbf{A}_z$  and a vector  $\mathbf{b}_z$ . From Eq. (5), we want to obtain

Table 2: Network architecture for CNN-based method.  $C_*$  and  $R_*$  indicate 2-D convolutional layer and residual connection, respectively.  $n_{\text{in}}$  views from input light field is stacked along channel dimension. Output is represented by  $n_{\text{out}}$  channels.  $D$  is number of layers.

Layer	Input	Kernel	Ch <sub>in</sub> /Ch <sub>out</sub>	Act
input	light field			
$C_{\text{in}}$	input	$3 \times 3$	$n_{\text{in}}/64$	ReLU
$C_{1a}$	$C_{\text{in}}$	$3 \times 3$	$64/64$	ReLU
$C_{1b}$	$C_{1a}$	$3 \times 3$	$64/64$	
$R_1$	$C_{\text{in}} + C_{1b}$			ReLU
$C_{2a}$	$R_1$	$3 \times 3$	$64/64$	ReLU
$C_{2b}$	$C_{2a}$	$3 \times 3$	$64/64$	
$R_2$	$R_1 + C_{2b}$			ReLU
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$C_{9a}$	$R_8$	$3 \times 3$	$64/64$	ReLU
$C_{9b}$	$C_{9a}$	$3 \times 3$	$64/64$	
$R_9$	$R_8 + C_{9b}$			ReLU
$C_{\text{out}}$	$R_9$	$3 \times 3$	$64/n_{\text{out}}$	Hard Sigmoid
output	$C_{\text{out}}$			

	Add	Mul	MPI (4ch)	MPI (6ch)
$n_{\text{in}}$	81	81	243	81
$n_{\text{out}}$	$D$	$D$	$4D$	$2D$

$$\mathbf{c}_z = \arg \min_{\mathbf{c}_z} \|\mathbf{I} - (\mathbf{A}_z \mathbf{c}_z + \mathbf{b}_z)\|^2, \quad (6)$$

This square error can be minimized using the update rule:

$$\mathbf{c}_z \leftarrow \mathbf{c}_z \odot \{(\mathbf{A}_z)^T (\mathbf{I} - \mathbf{b}_z)\} // \{(\mathbf{A}_z)^T \mathbf{A}_z \mathbf{c}_z\} \quad (7)$$

where  $\odot$  and  $//$  indicate element-wise product and division, respectively. After Eq. (7) is applied, all elements of  $\mathbf{c}_z$  are clipped to  $[\varepsilon, 1]$ . We also update  $\alpha_z$  in the same manner.

### 4 Experiments

We experimentally compared the performance of MPIs against the previous multiplicative and additive layered displays. For MPIs, we tested two channel structures; 4 channels and 6 channels. For the former case, each pixel on the layer is represented by three color (RGB) values and a single alpha value, while for the latter case, each of the color values has an individual alpha value. Both the iterative and learning-based (CNN) methods were used to obtain the layer patterns, the details of which are summarized in Tables 1 and 2. To train CNN-based methods, 53,040 light field samples with RGB colors and  $9 \times 9$  views were collected from public datasets [7, 8]. Except for the case with MPI (4ch), three color channels were treated as three individual samples, because the network processed three color channels individually. Meanwhile, iterative methods did

Table 3: PSNR of different methods (averaged over 9 scenes)

Layers	Add		Mul		MPI (4ch)		MPI (6ch)	
	iter	CNN	iter	CNN	iter	CNN	iter	CNN
3	26.93	26.35	27.05	26.86	29.06	26.61	<b>29.19</b>	28.19
5	28.51	27.74	28.91	27.77	<u>31.61</u>	27.99	<b>31.80</b>	29.60
9		29.87		<u>31.70</u>		29.91		<b>32.94</b>

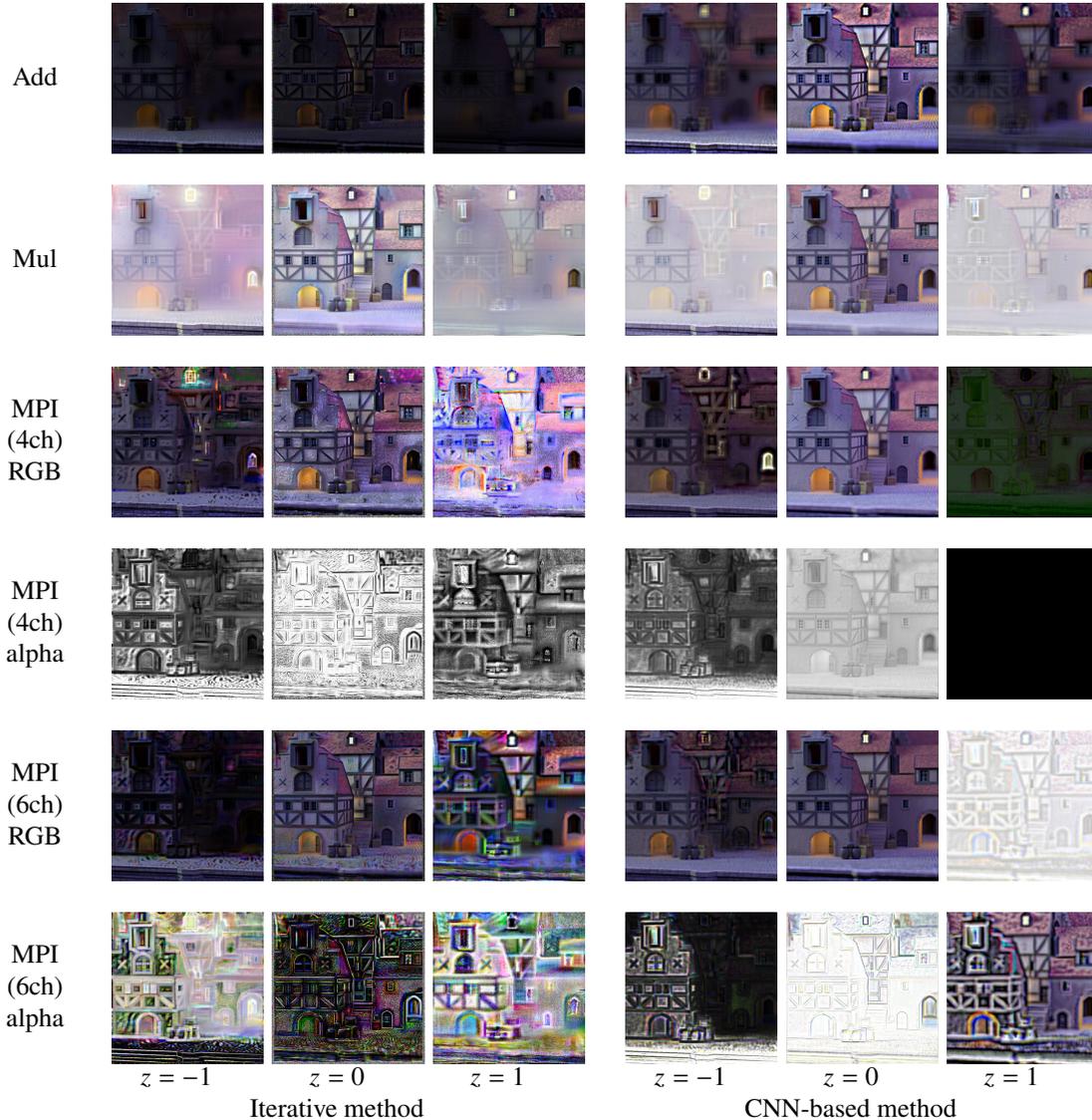


Fig. 2: Layer patterns obtained by different methods (3 layers)

not need any training. For evaluation, we took target light fields with  $9 \times 9$  views from HCI datasets [7]. We obtained a set of layer patterns for each of the target light fields using the methods mentioned above. Finally, from the layer patterns, a light field was computationally reconstructed and compared against the ground truth to obtain quantitative (PSNR) scores.

We present averaged PSNR scores over 9 light fields in Table 3. Here, the layers' depth were set to  $Z^{(3)} = \{-2, 0, 2\}$ ,

$Z^{(5)} = \{-2, -1, 0, 1, 2\}$ , and  $Z^{(9)} = \{-2, -1.5, \dots, 1.5, 2\}$  for all the methods. For MPIs, the iterative method resulted in better quality than the CNN-based counterparts. Moreover, MPIs obtained by the iterative method outperformed the previous multiplicative and additive layered displays.

We visualize some layer patterns in Fig. 2. We found that the iterative method produced small vibrations on the layer patterns, while the CNN-based method yielded rather smooth results. Shown in Fig. 3 are some top-left views

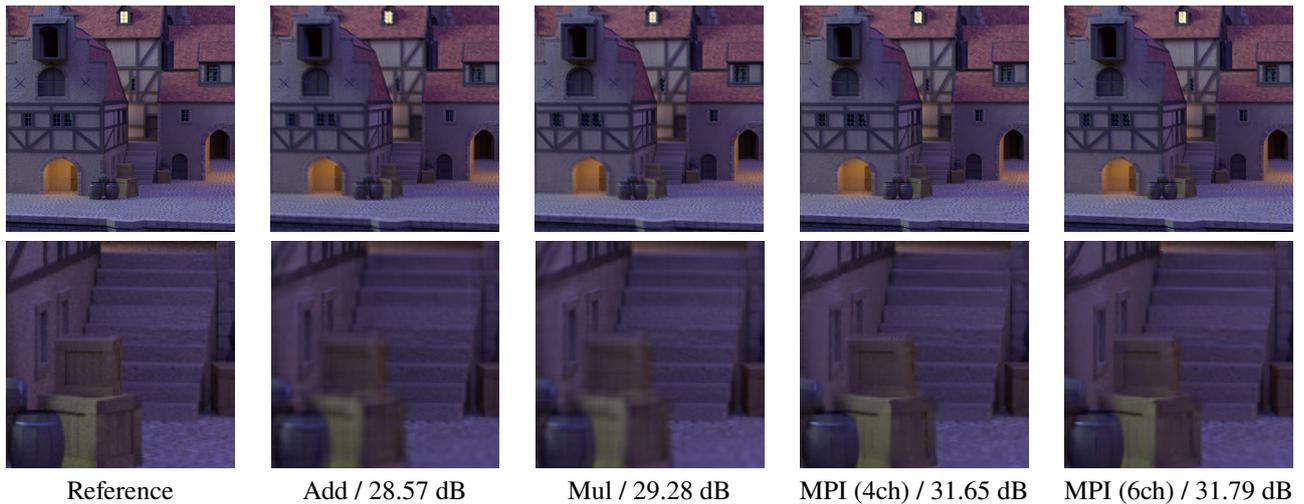


Fig. 3: Top-left views computed from 5 layers, which were obtained by iterative methods.

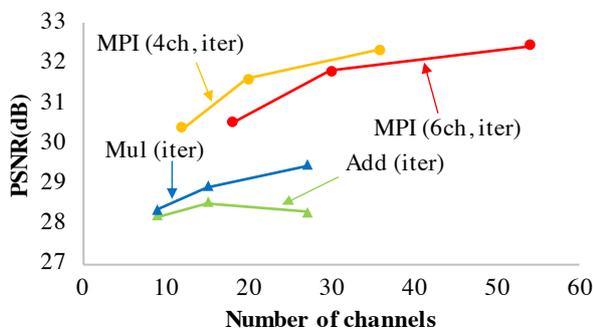


Fig. 4: Image quality and number of channels

computed from 5 layers. We can see that MPIs yielded better visual quality than the previous multiplicative and additive layered displays.

A comparison from another perspective is presented in Fig. 4. Here, the layers' depths were set to  $Z^{(3)} = \{-1, 0, 1\}$ ,  $Z^{(5)} = \{-2, -1, 0, 1, 2\}$ ,  $Z^{(9)} = \{-4, -3, \dots, 3, 4\}$ , and the performance was evaluated against the number of channels. We can see that MPI (4ch) is superior to MPI (6ch) in terms of the total number of channels, and both of them outperform the previous multiplicative and additive layered displays.

## 5 Conclusion

We demonstrated that an MPI can be regarded as a layered display that has both multiplicative and additive layers. We also derive an iterative method for obtaining an MPI from a light field. We finally compared the performance of MPI against the previous layered display to show its superiority. We hope to see that the display panels that can conduct both multiplicative and additive operations become available in the future. This development will not only revolutionize 3-D displays but also enhance the affinity between 3-D displays and computer graphics.

## References

- [1] T. Zhou, R. Tucker, J. Flynn, G. Fyffe and N. Snavely, "Stereo Magnification: Learning View Synthesis using Multiplane Images," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 37, no 4, pp. 1-12, 2018.
- [2] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng and A. Kar, "Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 38, no 4, pp. 1-14, 2019.
- [3] R. Tucker and N. Snavely, "Single-View View Synthesis with Multiplane Images," *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [4] G. Wetzstein, D. Lanman, M. Hirsch, and R. Raskar, "Tensor Displays: Compressive Light Field Synthesis using Multi-layer Displays with Directional Backlighting," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 111, 2012.
- [5] S. Lee, C. Jang, S. Moon, J. Cho, and B. Lee, "Additive light field displays: Realization of augmented reality with holographic optical elements," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 35, no. 4, pp. 113, Jul. 2016.
- [6] K. Maruyama, K. Takahashi, and T. Fujii, "Comparison of Layer Operations and Optimization Methods for Light Field Display," *IEEE Access*, vol. 8, pp. 38767-38775, 2020, doi: 10.1109/ACCESS.2020.2975209.
- [7] Heidelberg Collaboratory for Image Processing. *4D Light Field Dataset*. [Online]. Available: <http://hci-lightfield.iwr.uni-heidelberg.de/> 2018.
- [8] Computer Graphics Laboratory, Stanford University. *The (New) Stanford Light Field Archive*. [Online]. Available: <http://lightfield.stanford.edu> 2018.