Light Field Synthesis from a Monocular Image using Variable LDI

Junhyeong Bak and In Kyu Park
Department of Electrical and Computer Engineering, Inha University
Incheon 22212, Korea
{gladstone1840@gmail.com, pik@inha.ac.kr}

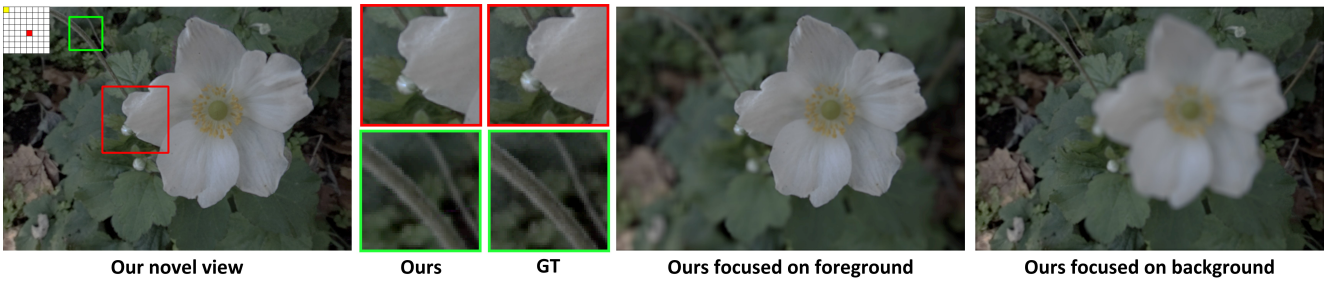

Figure 1. Novel view synthesis and refocusing result of the proposed method.

Abstract

Recent advancements in learning-based novel view synthesis enable users to synthesize light field from a monocular image without special equipment. Moreover, the state-of-the-art techniques including multiplane image (MPI) show outstanding performance in synthesizing accurate light field from a monocular image. In this study, we propose a new variable layered depth image (VLDI) representation to generate precise light field synthesis results using only a few layers. Our method exploits LDI representation built on a new two-stream halfway fusion network and transformation process. This framework has an efficient structure that directly generates the region that does not require network prediction from inputs. As a result, the proposed method allows us to acquire high-quality light field easily and quickly. Experimental results show that the proposed method outperforms the previous works quantitatively and qualitatively for diverse examples.

1. Introduction

Light field captures the direction and intensity of rays in space within a single image. It is a useful photographic technique because it can be post-processed, such as viewpoint change, refocusing, and depth estimation. Recently, light field has received huge attention from the industry for the possible use of immersive content creation, especially in augmented reality (AR) and virtual reality (VR). However, acquisition of light field remains problematic as it requires professional equipment such as a plenoptic camera or multiple camera array [12, 23].

Srinivasan *et al*. [18]'s pioneering work has become a dominant technique to synthesize a light field from a monocular image. Nevertheless, this method is trained only for a particular type of object and consequently poorly performs in the general scene. Subsequently, Li *et al*. [10] proposed a method that uses an additional monocular depth estimation model learned from various scenes for geometry estimation. This method can deal with occluded areas robustly by using layered representation, that is, multiplane image (MPI). For this purpose, it uses a method to learn the geometrical scale of dataset through the extended representation, that is, variable MPI (VMPI).

In this paper, we propose a novel method that utilizes an improved variable layered depth image (VLDI) for light field synthesis. Unlike conventional MPI and VMPI which have a single depth value per layer, our method exploits layered depth image (LDI) in which layer depth is encoded in a per-pixel manner. Thus, it can produce more accurate light field synthesis results using only a few layers. Fig. 1 shows an example of the novel view synthesis and refocusing result.

To build VLDI, we specially design a framework consisting of synthesis and transformation stages. It inputs a monocular image and its normalized depth image that a monocular depth estimation model predicts. Our synthesis network simultaneously produces RGB and depth channels

of VLDI, which have different characteristics by using a newly designed two-stream halfway fusion structure. Owing to the separated streams of the network, inter-channel interference caused by simultaneously estimating different types of channels might be minimized. Thus, the network only estimates the in-painted RGB and scaled layer depth. We demonstrate that the proposed method can quickly produce clearer and more accurate light field synthesis results through various comparisons with state-of-the-art methods. Our main contributions are summarized as follows.

- A novel VLDI scene representation that generates more natural motion parallax results with only a few layers.
- An efficient VLDI construction framework, which combines synthesis and transformation stages that minimizes network prediction.
- A two-stream network in the form of halfway fusion so that different types of channels configuring VLDI can be synthesized with minimal interference.

2. Related Works

Light Field Synthesis. Light field synthesis began with a study on an angular super-resolution (SR) method capable of generating dense light fields through a small number of input images. Wanner *et al*. [22] used depth as geometry information obtained by epipolar plane image (EPI) analysis. Zhang *et al*. [26] utilized a phase-based method that can use a tiny baseline stereo pair as input.

Kalantari *et al*. [8] pioneered a method of synthesizing novel 8×8 sub-aperture images (SAIs) in fronto-parallel baseline using a convolutional neural network that estimates the parallax of each SAI from the four corner SAIs. Wu *et al*. [24] proposed a method that uses EPI up-sampling and deblur from several input images.

Srinivasan *et al*. [18] proposed a monocular-based light field synthesis. Notably, this method includes approximating the occluded and non-Lambertian regions. In a similar fashion, Ivan *et al*. [7] proposed a method of using geometric representation called appearance flow, and introduced a new loss function which can avoid a reflection on pixel brightness. In a recent study, Chen *et al*. [1] showed that the generative adversarial networks (GAN) approach could be applied in this work. Li *et al*. [10] demonstrated that the MPI representation could generate light field type results by learning the scale of geometry from dataset through a extended structure, that is, VMPI.

Multiplane Image. MPI originated as a photographic technique of film animation. It is a 3D scene representation that contains fronto-parallel multiple planes including occluded areas. Zhou *et al*. [27] proposed a pioneering work in generating MPI by using a learning-based method. The original MPI method can produce enlarged baseline results from a stereo pair with a small baseline. To construct MPI, the network leverages the scene geometry extracted from input and naturally handles occlusion areas by using differentiable layered representation. Subsequently, Srinivasan *et al*. [17] and Flynn *et al*. [3] proposed improvements to the above method by extrapolating it with an extensive baseline. These methods acquire the scene geometry required for synthesis by constructing a plane sweep cost volume from reference images.

However, pose information is necessary for constructing the precise plane sweep cost volume. Moreover, the model itself might be complex for handling. Furthermore, MPI-synthesis-based methods that utilize multiple input images might not be convenient for synthesis. Thus, Tucker *et al*. [19] proposed a monocular-based MPI synthesis method to solve the aforementioned problems. Specifically, this method utilizes a scale-invariant learning approach to overcome the scale ambiguity of the monocular-based scene geometry estimation.

Layered Depth Image. Shade *et al*. [14] proposed the original LDI as a 3D scene representation consisting of multiple layers containing occluded regions. LDI is similar to MPI, but there is a inherent difference that all layers have per-pixel depth value. Hereby, the layer is not a plane and may be expressed as a sprite. This form can naturally represent motion parallax, even when the used number of layers is less than MPI.

In fact, various methods can be used to divide layers in LDI. For example, Tulsiani *et al*. [20] designed a method of synthesizing two layers consisting of front and back. However, a limitation of this approach is the difficulty of coping with various occlusion on the basis of a few layers. On the other hand, Shih *et al*. [15] exploited a segmentation method that can construct layers adaptively and detect occlusion through depth edges. Interestingly, the occluded areas are inpainted by their method of catching contextual information. Although this method can generate extensive baseline results, it does not support differentiable rendering like MPI. Therefore, synthesizing non-Lambertian surfaces through this method remains challenging.

3. Proposed Method

Fig. 2 shows the overall structure of our network. Given an input monocular image I and target position p in u and v directions, the proposed method synthesizes $\widehat{I}$ for estimating the target I_t. The total network consists of a monocular depth estimation network that estimates scene geometry information and a VLDI synthesis network that performs inpainting on the occluded region and estimates the scaled depth of each layer. The VLDI synthesis network receives input of monocular image I and its normalized depth images I_D. The geometrical scale is fixed to the light field

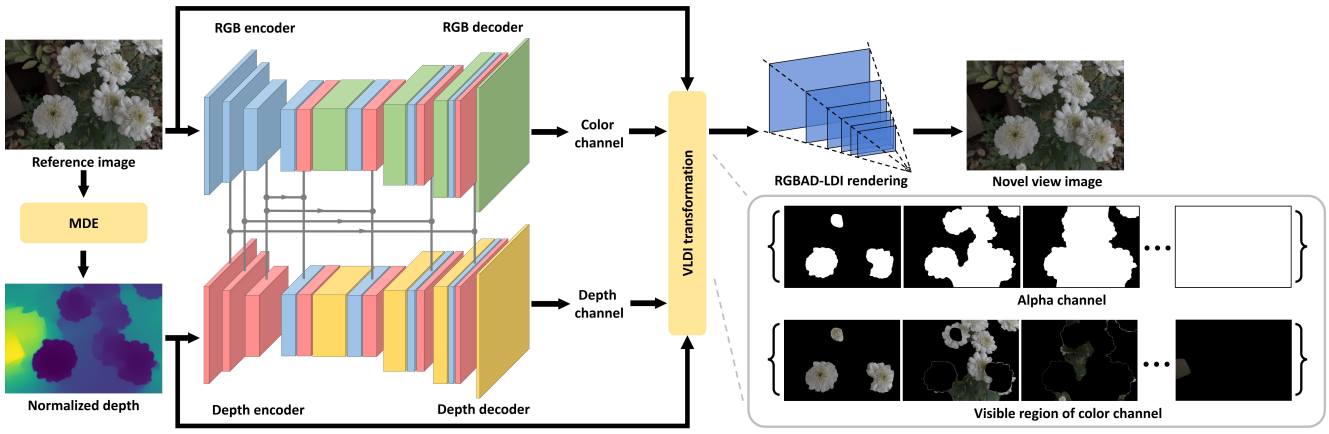

Figure 2. Our pipeline renders novel view using VLDI representation which is built on the synthesis network and transformation process. MDE denotes monocular depth estimation. The network has a two-stream halfway fusion structure to estimate the RGB and scaled layer depth simultaneously. The alpha channel which defines the layers and the visible region of RGB channel is generated through the transformation process without network prediction.

dataset through training in the same manner as VMPI [10]. Furthermore, our method does not require any network to generate the alpha channel defining the layers and the visible regions. We achieve such generation directly from inputs passed through our transformation process.

3.1. Monocular Depth Estimation (MDE)

The depth information required for light field synthesis can use all kinds of normalized depth images. However, we utilize a general monocular depth estimation model to obtain the depth from an input monocular image in the same manner as VMPI [10]. In fact, the depth estimation model can increase generalization performance by means of LeRes [25] model, which is pretrained on a wider variety of scenes and larger amounts of data than light field datasets. Such depth estimation model provides a state-of-the-art performance compared with recently released monocular-based models. Hence, we can obtain sophisticated scene geometry information. However, the input image obtained from the light field dataset generally has a dark tendency, it is input to the model by improving brightness through gamma correction. Finally, the depth estimation result is normalized to remove inaccurate scale information estimated by the monocular-based model.

3.2. Representation

The proposed VLDI representation designed by applying VMPI [10] structure has N fixed number of layers. Each layer of VLDI is combined with RGB channel C and transparent map, alpha channel A, which can be expressed by $L_i(x,y) = \{C_i(x,y), A_i(x,y)\}$. Here, x and y denote pixel position. In addition, VLDI estimates depth channel D together. Unlike the VMPI method, ours does not make the depth of each layer as a single value through average. Therefore, it can be expressed as $D(x,y) = \{D_i(x,y), ..., D_N(x,y)\}$. To render a VLDI for generating a targeted view, each layer is warped using a relative pose and depth. Given that the depth channel is actually learned as disparity and not as depth, warping can be expressed as follows.

$$\hat{L}_i(x,y) = L_i(x + p_u D_i(x,y), y + p_v D_i(x,y)) \quad (1)$$

Generating a final target image by compositing the warped layers can be expressed as follows, as in VMPI [10].

$$\hat{I}_i = \left(1 - \hat{A}_i\right) \hat{I}_{i-1} + \hat{A}_i \hat{C}_i \, , where \; \hat{I}_1 = \hat{C}_1 \quad (2)$$

3.3. Transformation

The existing MPI synthesis model estimates the alpha channel that determines the layers through a network prediction. Moreover, the final pixel value is determined from values of several layers through compositing. However, the proposed LDI-based method directly generates a binarized alpha channel without network from the inputs. To perform our transformation process, first, the mask $\mathcal{V}$ for the visible region of LDI is obtained as shown in Eq. (3) through a quantization of the input depth.

$$\mathcal{V}_i(x,y) = \begin{cases} 1, \text{ if } i = N - \text{round}\left((N-1) I_D(x,y)\right) \\ 0, \text{ otherwise} \end{cases} \quad (3)$$

where round$(\cdot)$ is a rounding function. The final alpha channel generated from $\mathcal{V}$ can be obtained as follows.

$$A_i = \mathcal{V}_i + \sum_{j=i+1}^{N} \mathcal{V}_j, \text{where } \; A_N = \mathcal{V}_N \quad (4)$$

The transformation alpha channel makes the pixel value of the RGB channel determined by only one layer, even when the composition is performed. Then, the visible area for the RGB channel of the LDI is obtained, as shown in Eq. (5), to replace it with an input image. We denote the RGB channel estimated through the network as $C^{'}$ to distinguish it from the final RGB channel. Although copying input images is similar to the approach of [19], our approach has a substantial difference. We obtain results that are inpainted by layer rather than a single background, and the alpha channel is not estimated over the network. This functionality can be formulated as

$$C_i = I\mathcal{V}_i + C_i^{'}\left(1 - \mathcal{V}_i\right). \tag{5}$$

3.4. Synthesis Network

The network receives RGB image and normalized depth image as inputs and simultaneously estimates depth channels and the occluded region of the RGB channel by the two-stream architecture. However, given that RGB image and depth image have different characteristics, interference occurs between channels when the network has early-fusion structure. Therefore, we design a novel LDI synthesis network in the form of halfway fusion in which the features of the two inputs can be appropriately fused in the middle and decoders are separated to generate two different outputs. This network consists of two encoders and two decoders, but we do not use weight sharing between encoder-encoder and decoder-decoder. Instead, we establish skip connections between encoders and decoders for transferring meaningful features. In architectural level, our network modules have similar structure presented in original VMPI [10], but we replace ReLU and Tanh activations with leaky ReLU and HardTanh activations respectively.

3.5. Training

Our model renders VLDI to induce learning in a direction where the difference between the estimated target SAI and ground truth is minimal. As the loss function, we adapt L1 loss for areas where occlusion does not occur, and straightforwardly calculate it using Eq. (6). The soft visibility mask M [10] is used to mask the occluded part that is revealed after rendering.

$$\mathcal{L}_{LAD} = \left\| M(I_t - \hat{I}) \right\|_1 \tag{6}$$

Losses for the entire area including the occluded part use SSIM-based [21] loss to increase structural similarity and VGG-based [16] loss to increase perception similarity of occluded areas. The SSIM loss can be expressed as follows,

$$\mathcal{L}_{SSIM} = 1 - \text{SSIM}\left(I_t, \hat{I}\right) \tag{7}$$

Method	PSNR [dB] ↑		SSIM ↑	
	8×8	15×15	8×8	15×15
Flower dataset				
Srinivasan *et al.* [18]	37.568	N/A	0.920	N/A
Ivan *et al.* [7]	37.271	N/A	0.918	N/A
Li *et al.* [10]	36.784	35.180	0.909	0.855
Ours	**38.196**	**36.155**	**0.933**	**0.884**
Stanford dataset				
Srinivasan *et al.* [18]	36.803	N/A	0.883	N/A
Ivan *et al.* [7]	35.857	N/A	0.854	N/A
Li *et al.* [10]	36.683	35.695	0.897	**0.854**
Ours	**37.360**	**36.001**	**0.902**	0.851
Kalantari dataset				
Srinivasan *et al.* [18]	34.641	N/A	0.829	N/A
Ivan *et al.* [7]	34.273	N/A	0.828	N/A
Li *et al.* [10]	34.620	33.650	0.852	0.783
Ours	**35.699**	**34.364**	**0.876**	**0.799**

Table 1. Quantitative comparison. We evaluate the PSNR and SSIM on three datasets by synthesizing 8×8 and 15×15 light field. All models are trained using only the Flower dataset.

The VGG-based loss uses the method proposed in [27], which compares $\hat{\lambda}_i$-weighted VGG outputs ϕ^i_{VGG} of multiple layers by

$$\mathcal{L}_{VGG} = \sum_{i=1}^{5} \hat{\lambda}_i \left\| \phi^i_{VGG}\left(I_t\right) - \phi^i_{VGG}(\hat{I}) \right\|_1 \tag{8}$$

As a result, the total loss can be expressed as follows,

$$\mathcal{L} = \lambda_1 \mathcal{L}_{LAD} + \lambda_2 \mathcal{L}_{SSIM} + \lambda_3 \mathcal{L}_{VGG} \tag{9}$$

where λ_1, λ_2 and λ_3 are the weights to balance the output of each loss.

4. Experimental Results

4.1. Setting

We train our model using Adam optimizer [9] with hyperparameters, learning rate $2e-4$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$, and batch size 12. The loss function weights in Eq. (9) are set to $\lambda_1 = 2.5$, $\lambda_2 = 1$ and $\lambda_3 = 2.5$. The number of layer N is 8. Training and testing are performed on Intel i7-9700K CPU with 32GB RAM and NVIDIA RTX 2080Ti GPU with 11GB VRAM. The dataset for training is the flower light field dataset [18] taken using a Lytro Illum camera. The total number of samples in the dataset is 3,243 from which 100 are separated for the test. In the training procedure, the sample SAIs are randomly selected and cropped to 256×256.

The evaluation is performed through comparison with state-of-the-art models. Models that cannot produce light field-type result or are difficult to compare quantitatively

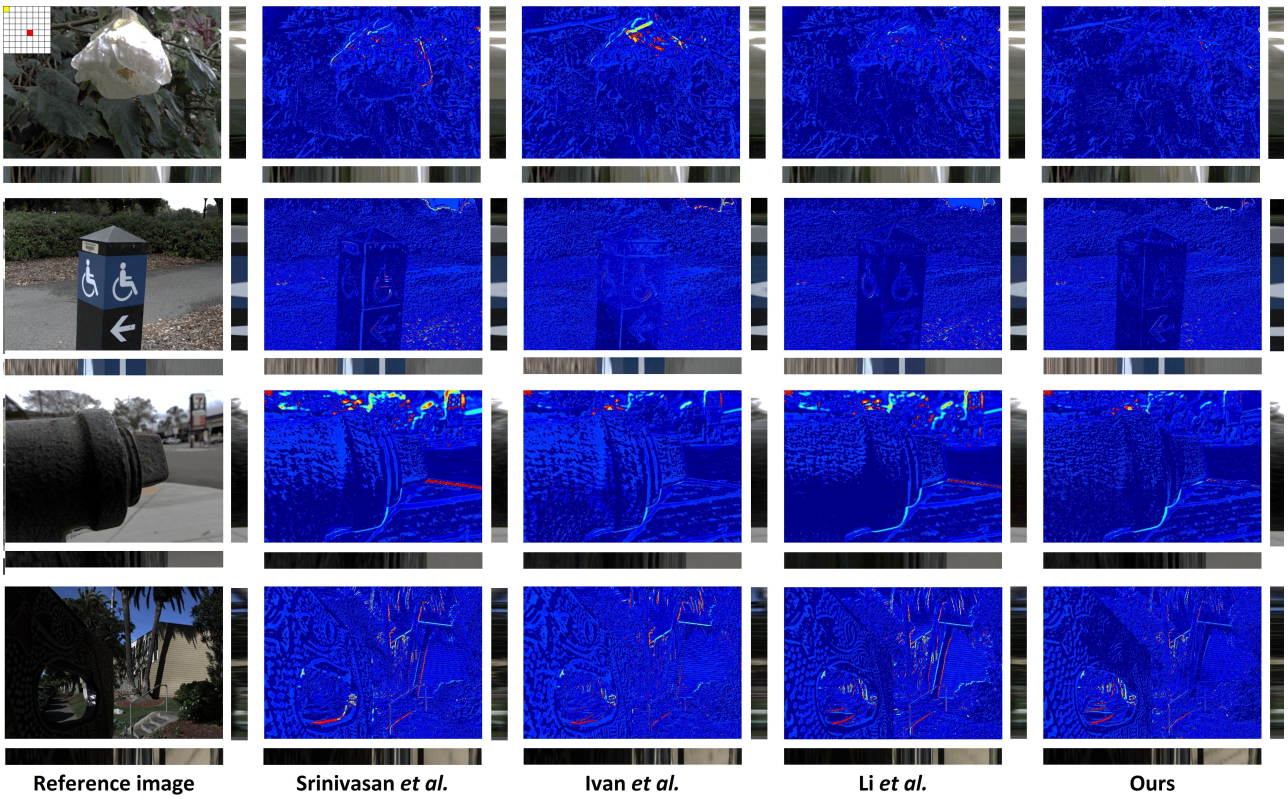

Figure 3. Qualitative evaluation using error map and EPI slicing for different dataset (Flower [18], Stanford [2], Kalantari [8]). The error map shows the overall difference of the predicted result and the ground truth. The prediction target is the corner SAI estimated from the center. Our model shows results with little difference. The EPI shows the horizontal and vertical motion parallax of the estimated light field. Likewise, our model is most similar to the ground truth.

due to scale-ambiguity are excluded from the evaluation. During the test phase, we utilize the Stanford dataset [2] and Kalantari dataset [8] consisting of various types of objects and scenes. Similar to the Flower dataset [18], the number of samples set to 100 and the models are not finetuned.

4.2. Novel View Synthesis

We perform a quantitative evaluation using PSNR [6] and SSIM [21] metrics by estimating and comparing 8×8 and 15×15 light fields. Here, 8×8 means the light field acquired by inputting the center SAI whereas 15×15 means the wide-baseline result acquired by inputting the top-left corner SAI. We exclude the model of Srinivasan *et al*. from the 15×15 evaluation because it could only be estimated from the center.

Table 1 shows the result for each test set. Our approach outperforms in most cases. Unlike the method of Srinivasan *et al*. and Ivan *et al*., using a separately learned depth estimation model has excellent results in all test sets, which confirms better generalization. The result is similar to that of Li *et al*., but our model outperforms on the flower dataset. Thus, our proposed model can improve scene geometry estimation better. In addition, unlike the method of Li *et al*., our method shows better performance with only a single LDI synthesis network. This result verifies that the proposed network can be easily optimized.

As a qualitative evaluation, an error map and an EPI slicing comparison are generated as shown Fig. 3. The previous methods show noticeable error on occlusion boundaries, visual edges, and letters; whereas the proposed method shows significantly less error. The EPI comparison results visually confirm that the proposed method is the closest to the ground truth.

Fig. 4 shows the comparison of the undesired blur effect on the occlusion area. Although minor artifacts appear, the proposed method produces much clearer pixels than others. Srinivasan *et al*.'s result shows the boundary is dragged around. It is observed that the proposed method generates significantly less blur enough to recognize the letters.

Although there appears little artifact around the boundary, sharpness and unpleasing dragging are significantly improved and they affect the quality of light field rendering more. Boundary artifacts come from mismatched boundary location between the input and the estimated depth from

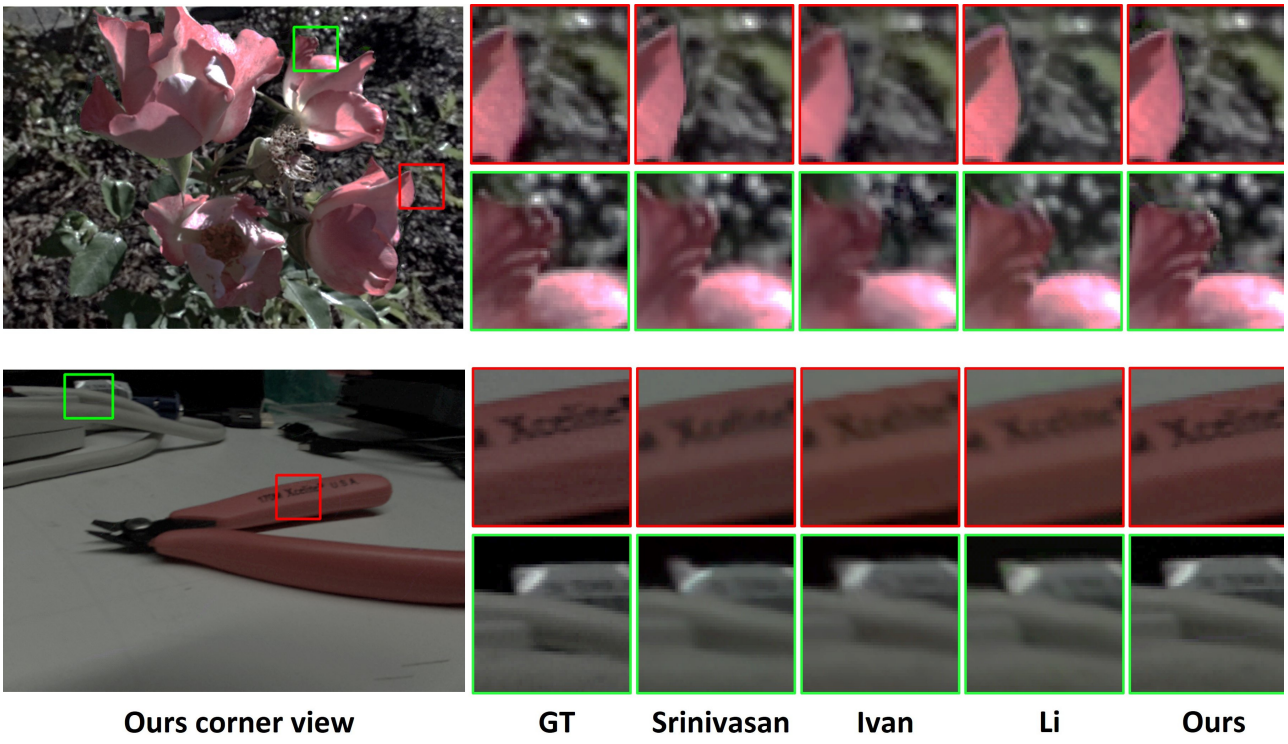

Figure 4. Qualitative comparison with other approaches through zoom and crop. In our method, the image rarely blurs and the boundary is not dragged around.

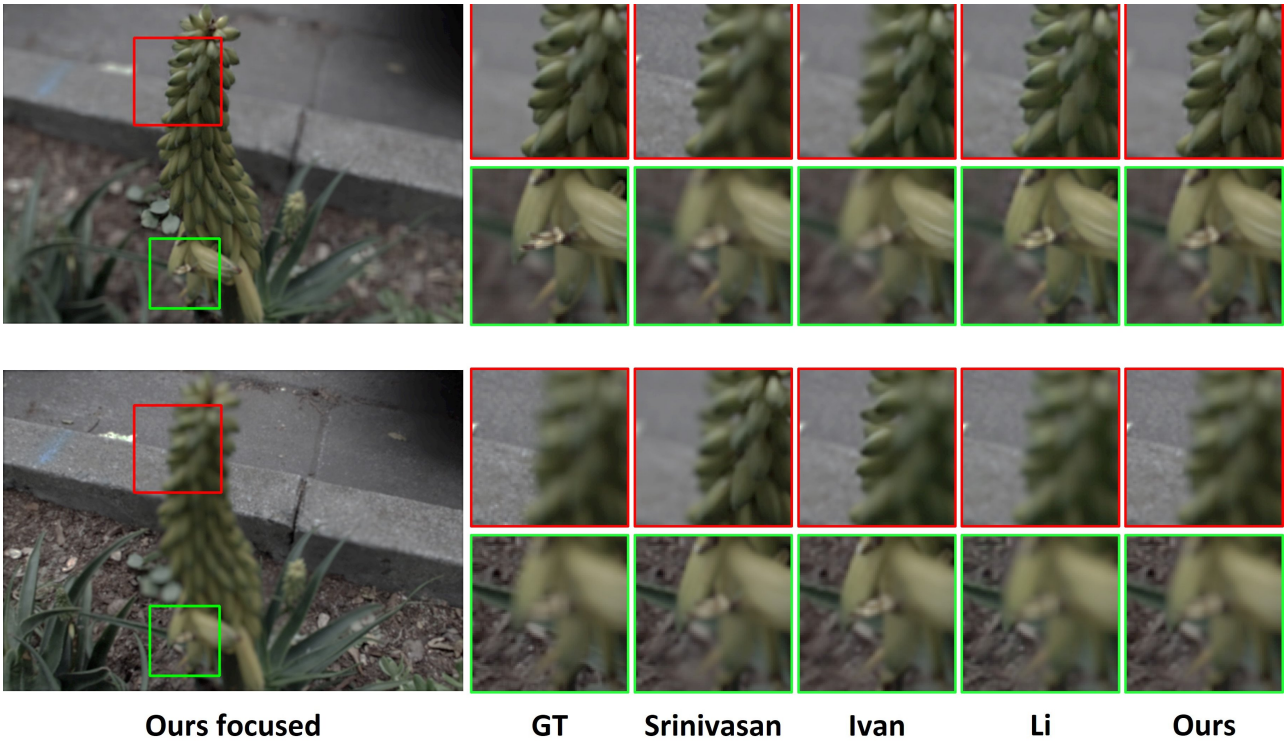

Figure 5. Qualitative comparison of refocusing. Foreground and background focus are compared. Our method is most similar to the ground truth.

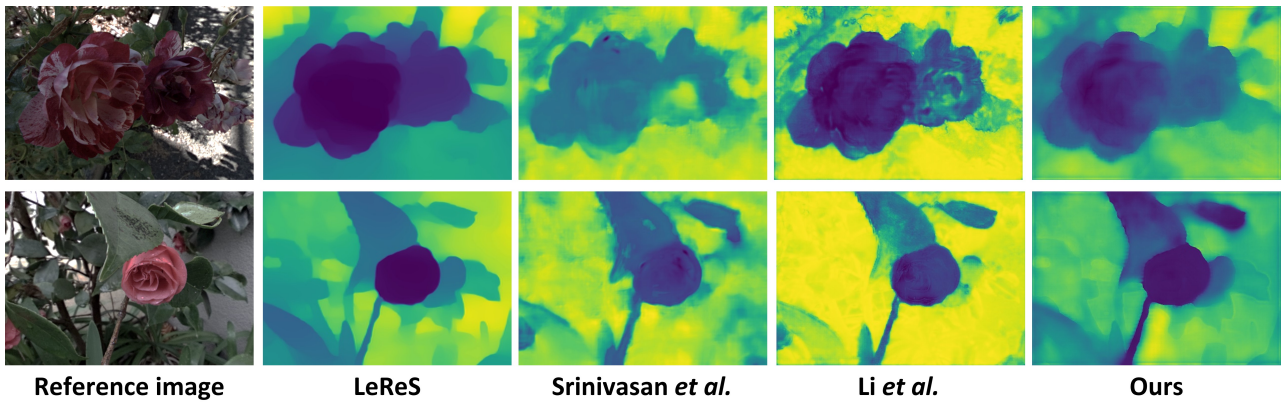

Figure 6. We compare the estimated depth map of each model. Given that no ground truth is available, we show the depth of LeReS as input. The comparison shows that our method has the least outliers and is similar to the LeReS results, but scales and details produce well-adjusted results.

MDE module, *i.e*. it appears less as the depth quality improves. To reduce the artifact, we can consider adopting preprocessing like bilateral median filter that sharpens the unclear depth boundary [11]. Another feasible method is to remedy those regions through depth completion with boundary consistency [5].

4.3. Refocusing

If the light field synthesis is performed well, then the refocusing results should also be natural. To verify it, we perform a qualitative evaluation on refocused results. The evaluation uses a method of generating a result of focusing on the foreground as well as a result of focusing on the background, and comparing it by cropping and zooming it. Fig. 5 shows the results. Notably, the proposed method demonstrates natural focusing results in most cases. In particular, more clear results are shown even when objects with different depths overlap. By contrast, the other methods produce results in which the focus is not well-matched in some areas.

4.4. Depth Evaluation

The proposed method does not use the input depth as scene geometry as it is. In fact, it helps to create alpha channels by simply grasping the relation between the near and far, and provides coarse scene geometry without scale. Therefore, the model is able to adjust the scale through learning and generate more accurate depth. Fig. 6 demonstrates the results of checking and comparing different methods. Given the absence of true depth information for the light fields, the monocular-based depth estimation results used as input and the depth results of several models are compared. Ivan *et al*.'s model is exempted because it only produces flow-type result, not depth. In the case of Srinivasan *et al*.'s method, severe outliers can be seen in several places. These outliers use to cause outliers in the synthesis results too. In the case of Li *et al*. method, the details of the background area are extremely damaged. However, our method is similar to the input depth where the details are increased, and the scale is well-adjusted. Given that this accurate scene geometry can be estimated, motion parallax of the proposed method appears accurately.

Method	Model size ↓	Processing Time ↓
Srinivasan *et al*. [18]	**3**	**0.751**
Ivan *et al*. [7]	59	0.865
Li *et al*. [10]	34	2.791
Ours	30	1.888

Table 2. Processing time (seconds) and model size (MB) to synthesize 8×8 light field from a monocular image.

4.5. Processing Time

The proposed method uses fewer layers and a single LDI synthesis network; hence, such elements are advantageous in terms of processing speed. To confirm this statement, we measure and compare the processing time taken to synthesize the light fields by different methods. Table 2 presents the experimental results. Note that the model sizes of Li *et al*. and ours do not include the size of the depth estimation model. On the contrary, the processing time includes the inference time of depth estimation model. The experiment shows that the method of Srinivasan *et al*., which has simple networks and does not use layered representation, is the fastest method. Our model is slightly slower than this method. However, ours takes much less processing time than Li *et al*.'s method does, which uses a similar layered presentation.

Method	PSNR [dB] ↑		SSIM ↑	
	8×8	15×15	8×8	15×15
Structure of layered scene representation				
MPI	37.265	35.397	0.908	0.845
N = 16	37.089	35.248	0.905	0.840
N = 32	37.542	35.607	0.917	0.858
Structure of network				
w/o skip-connection	37.424	35.450	0.912	0.848
one-stream input fusion	37.639	35.631	0.920	0.859
two-stream input fusion	36.324	34.788	0.875	0.809
Loss function composition				
w/o SSIM loss	36.638	34.913	0.890	0.820
w/o VGG loss	38.023	36.003	0.929	0.877
w/o SSIM and VGG loss	37.749	35.859	0.922	0.869
Monocular depth estimation				
VLDI + Godard *et al*. [4]	37.538	35.652	0.918	0.861
VLDI + Ranftl *et al*. [13]	37.421	35.361	0.912	0.847
Ours	**38.200**	**36.155**	**0.933**	**0.884**

Table 3. Ablation study with different representation, structure, loss function configuration and monocular depth estimation model.

4.6. Ablation Study

4.6.1 Different Network Configurations

To verify the effectiveness of our proposed method, we perform an ablation study on its different configurations by considering structure of layered representation and structure of network. We add models trained with different loss function and Table 3 presents the result. It demonstrates that our representation with eight layers performs better than all other approaches. Structural similarity is reduced significantly unless the LDI representation is employed. The models without proposed network structure also show relatively poor performance. Through the ablation study, we can see that our method encodes the 3D scene structure in a correct manner and it is mainly learned with SSIM loss. Moreover, it has an efficient network structure that reduces interference when fusing inputs. Furthermore, increasing the number of layers hardly improves performance. Considering the memory consumption and the processing time, the small number of layers (8) is enough in the proposed method.

4.6.2 Different Monocular Depth Estimation Modules

Our method relies on a pretrained monocular depth estimation model to acquire input 3D information. To check the effect of depth estimation model, we perform an additional ablation study on the different models as shown in the bottom section of Table 3. The compared models are the model of Godard *et al*. [4] trained by self-supervised manner and the model of Ranftl *et al*. [13] trained by supervised manner for various scenes. Note that both are well known as SOTA monocular depth estimation models. However, the model that we choose still shows better results because it has been trained with more diverse and accurate depth data. This means that using a better depth estimation model is a more reasonable strategy as expected.

5. Conclusion

In this paper, we proposed a monocular-based light field synthesis method, dubbed VLDI, which utilized only a few layers as an extended LDI representation. We designed the VLDI construction framework consisting of a synthesis network and a transformation process. The network was devised to reduce interference between channels by means of the proposed two-stream halfway fusion structure. The framework avoids additional network prediction through transformation. Furthermore, extensive experiments validated that our model can generate significantly improved results over various test images.

Acknowledgement

This work was partly supported by Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-IT1702-54. This work was partly supported by Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2022-00155915, Artificial Intelligence Convergence Innovation Human Resources Development (Inha University) and No.2021-0-02068, Artificial Intelligence Innovation Hub).

References

[1] Bin Chen, Lingyan Ruan, and Miu-Ling Lam. LFGAN: 4D light field synthesis from a single RGB image. *ACM Trans. on Multimedia Computing, Communications, and Applications*, 16(1):1–20, 2020. 2

[2] Donald G Dansereau, Bernd Girod, and Gordon Wetzstein. Liff: Light field features in scale and depth. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8042–8051, 2019. 5

[3] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019. 2

[4] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel Brostow. Digging into selfsupervised monocular depth estimation. In *Proc. IEEE/CVF International Conference on Computer Vision*. 8

[5] Yu-Kai Huang, Tsung-Han Wu, Yueh-Cheng Liu, and Winston H. Hsu. Indoor depth completion with boundary consistency and self-attention. In *Proc. IEEE/CVF International Conference on Computer Vision Workshop*, 2019. 7

[6] Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics Letters*, 44(13):800–801, 2008. 5

[7] Andre Ivan and In Kyu Park. Joint light field spatial and angular super-resolution from a single image. *IEEE Access*, 8:112562–112573, 2020. 2, 4, 7

[8] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM Trans. on Graphics*, 35(6):193:1–193:10, 2016. 2, 5

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[10] Qinbo Li and Nima Khademi Kalantari. Synthesizing light field from a single image with variable MPI and two network fusion. *ACM Trans. on Graphics*, 39(6):229–1, 2020. 1, 2, 3, 4, 7

[11] Ziyang Ma, Kaiming He, Yichen Wei, Jian Sun, and Enhua Wu. Constant time weighted median filtering for stereo matching and beyond. In *Proc. IEEE International Conference on Computer Vision*, 2013. 7

[12] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light field photography with a hand-held plenoptic camera. *Stanford University Computer Science Tech Report CSTR 2005-02*, April 2005. 1

[13] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, volume 44. 8

[14] Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski. Layered depth images. In *Proc. SIGGRAPH*, pages 231–242, 1998. 2

[15] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3D photography using context-aware layered depth inpainting. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8028–8038, 2020. 2

[16] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[17] Pratul Srinivasan, Richard Tucker, Jonathan Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely. Pushing the boundaries of view extrapolation with multiplane images. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 175–184, 2019. 2

[18] Pratul P Srinivasan, Tongzhou Wang, Ashwin Sreelal, Ravi Ramamoorthi, and Ren Ng. Learning to synthesize a 4D RGBD light field from a single image. In *Proc. IEEE International Conference on Computer Vision*, pages 2243–2251, 2017. 1, 2, 4, 5, 7

[19] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 551–560, 2020. 2, 4

[20] Shubham Tulsiani, Richard Tucker, and Noah Snavely. Layer-structured 3D scene inference via view synthesis. In *Proc. European Conference on Computer Vision*, pages 302–317, 2018. 2

[21] Zhou Wang, Alan Bovik, Hamid Sheikh, and Eero Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. on Image Processing*, 13(4):600–612, 2004. 4, 5

[22] Sven Wanner and Bastian Goldluecke. Variational light field analysis for disparity estimation and super-resolution. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 36(3):606–619, 2013. 2

[23] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. *ACM Trans. Graphics*, 24(3):765–776, 2005. 1

[24] Gaochang Wu, Mandan Zhao, Liangyong Wang, Qionghai Dai, Tianyou Chai, and Yebin Liu. Light field reconstruction using deep convolutional network on EPI. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 6319–6327, 2017. 2

[25] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3D scene shape from a single image. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 204–213, 2021. 3

[26] Zhoutong Zhang, Yebin Liu, and Qionghai Dai. Light field from micro-baseline image pair. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3800–3809, 2015. 2

[27] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 2, 4