End-to-End View Synthesis for Light Field Imaging with Pseudo 4DCNN

Yunlong Wang[1,2], Fei Liu[2], Zilei Wang[1], Guangqi Hou[2], Zhenan Sun[2(✉)], and Tieniu Tan[1,2]

[1] University of Science and Technology of China, Hefei, China
zlwang@ustc.edu.cn
[2] Center for Research on Intelligent Perception and Computing, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China
{yunlong.wang,fei.liu}@cripac.ia.ac.cn, {gqhou,znsun,tnt}@nlpr.ia.ac.cn

Abstract. Limited angular resolution has become the main bottleneck of microlens-based plenoptic cameras towards practical vision applications. Existing view synthesis methods mainly break the task into two steps, *i.e.* depth estimating and view warping, which are usually inefficient and produce artifacts over depth ambiguities. In this paper, an end-to-end deep learning framework is proposed to solve these problems by exploring Pseudo 4DCNN. Specifically, 2D strided convolutions operated on stacked EPIs and detail-restoration 3D CNNs connected with angular conversion are assembled to build the Pseudo 4DCNN. The key advantage is to efficiently synthesize dense 4D light fields from a sparse set of input views. The learning framework is well formulated as an entirely trainable problem, and all the weights can be recursively updated with standard backpropagation. The proposed framework is compared with state-of-the-art approaches on both genuine and synthetic light field databases, which achieves significant improvements of both image quality (*+2* dB *higher*) and computational efficiency (*over 10X faster*). Furthermore, the proposed framework shows good performances in real-world applications such as biometrics and depth estimation.

Keywords: View synthesis · Light Field · End-to-end Pseudo 4DCNN

1 Introduction

As a revolutionary imaging technology, Light Field (LF) imaging [1,11,14,23] has attracted extensive attention from both academia and industry, especially with the emergence of commercial plenoptic cameras [16] and recent dedication in the field of Virtual Reality (VR) and Augmented Reality (AR) [8]. With additional optical components like the microlens array inserted between the main lens and the image sensor, plenoptic cameras are capable of capturing both intensity and direction information of rays from real-world scenes, which enables

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11206, pp. 340–355, 2018.
https://doi.org/10.1007/978-3-030-01216-8_21

applications such as refocusing and 3D display. However, the inherent tradeoff between angular and spatial resolution is inevitable due to the limited sensor resolution, which restricts LF imaging in many practical vision applications.

One possible solution to this problem is view synthesis, which synthesizes novel views from a sparse set of input views. Inspired by traditional view synthesis approaches and recent success of data-driven methods, Kalantari *et al.* [9] break down the goal of view synthesis into the disparity estimator and color predictor modeled by convolutional neural network (CNN). Since disparities are implicitly inferred from the first CNN, it obtains better results than other state-of-the-art (SOTA) methods [12,21,22] that require explicit depth[1] information as priors for view warping. However, this framework is quite limited in reconstructing challenging LF scenes, such as occluded regions, non-Lambertian surfaces, *etc.* Actually, depth-dependent view synthesis methods inevitably rely on the accuracy of depth information, which tends to produce artifacts where inaccurate depth estimation usually happens. Moreover, they mostly generate a single novel view so that it is rather inefficient to synthesize all the in-between views.

Recently, Wu *et al.* [24] firstly model view synthesis as learning-based angular detail restoration on 2D Epipolar Plane Images (EPIs). They propose a "blur-restoration-deblur" framework without estimating the geometry of the scene. It achieves superior results than Kalantari *et al.* [9] on a variety of scenes, even in the occluded regions, non-Lambertian surfaces and transparent regions. However, their framework still has some shortcomings. Firstly, the full LF data is underused since EPIs are just 2D slices of 4D LF. Secondly, it is quite time-consuming because the operations of "blur-restoration-deblur" on EPIs loop numerous times before all the in-between views are synthesized.

In fact, 4D LF data are highly correlated in ray space, which record abundant information of the scene. The key insight of view synthesis for light field imaging is to make full use of the input views. Unlike 2D array or 3D volume, it is proved to be a tough problem working on the high dimensional data with CNN currently. Therefore, there scarcely exist approaches that address the problem of view synthesis in this way. In this paper, we propose an end-to-end learning framework that efficiently synthesizes dense 4D LF from sparse input views. Specifically, the learnable interpolation using 2D strided convolutions is applied on stacked EPIs to initially upsample 3D volumes extracted from LF data. Then, 3D CNNs are employed to recover high-frequency details of volumes in the row or column pattern. The angular conversion is introduced as the joint component to transform from receiving output of row network to giving input of column network. Moreover, a prior sensitive loss function is proposed to measure the errors of synthesized views according to the level of received prior knowledge. The learning framework is well formulated as an entirely trainable problem and all the weights can be recursively updated with standard backpropagation.

Experimental results on a variety of challenging scenes, including depth variations, complex light conditions, severe occlusions, non-Lambertian surfaces and

[1] depth and disparity are used interchangeably throughout the paper, since they are closely related in structured light fields.

so on, demonstrate that the proposed framework significantly outperforms other SOTA approaches with higher numerical quality and better visual effect. By directly operating on 4D LF data, the proposed framework also greatly accelerates the process of view synthesis, over one order of magnitude faster than other SOTA methods.

1.1 Depth-Dependent View Synthesis

Generally, depth-dependent view synthesis approaches synthesize novel views of a scene in a two-step process [3,5], *i.e.* estimating disparities of the input views and warping to the novel views based on the disparities, then combining warped images in a specific way (*e.g.* weighted summation) to obtain the final novel views.

Wanner and Goldluecke [22] propose the optimization framework to synthesize novel views with explicit geometry information, which only performs well for synthetic scenes with ground truth disparities, but produces significant artifacts for real-world scenes. The phase-based approach by Zhang *et al.* [28] reconstructs LF from a micro-baseline stereo pair. However, it is quite time-consuming for refining the disparity iteratively. The patch-based synthesis method by Zhang *et al.* [27] decomposes the disparity map into different layers and requires user interactions for various LF editing goals. Note that even state-of-the-art LF depth estimation methods are not specifically designed to be suitable for pixel warping. Thus view synthesis approaches that take explicit depth as priors usually fail to reconstruct plausible results for real-world scenes.

To alleviate the need of explicit depth information for view-warping, another strategy aims to synthesize novel views along with implicitly estimating the geometry of the scene. Kalantari *et al.* [9] propose the first deep learning system for view synthesis. Inspired by aforementioned methods, they factorize view synthesis into the disparity estimator and color predictor modeled by CNNs. Both networks are trained simultaneously by minimizing the error between the synthesized view and the ground truth. Thus, the disparity for view-warping is implicitly produced by the first CNN, which is more suitable for view synthesis application. However, this method is quite limited in reconstructing challenging LF scenes due to the insufficient information of warped images. Srinivasan *et al.* [18] build on the pipeline similar to Kalantari *et al.* [9], and synthesize a 4D RGBD LF from a single 2D RGB image. Overall, depth-dependent view synthesis strongly depends on the depth information. For challenging scenes that contain significant depth variations, complex lighting conditions, occlusions, non-Lambertian surfaces, *etc*, where inaccurate depth estimation usually happens, these methods tend to fail since the warped images are not able to provide sufficient information to synthesize high-quality views.

1.2 Depth-Independent View Synthesis

Alternative approaches for view synthesis are to upsample the angular dimensions without any geometry information of the scene. Some depth-independent

methods are designed to process the input LF sampled in specific patterns. For example, Levin and Durand [10] exploit dimensionality gap priors to synthesize novel views from a set of images sampled in a circular pattern. Shi *et al.* [17] sample a small number of 1D viewpoint trajectories formed by a box and two diagonals to recover 4D LF. To capture input views in such specific pattern is rather difficult, and thus these methods are still far from practical applications.

Many learning-based methods working on angular SR of LF have been proposed recently. Yoon *et al.* [26] propose a deep learning framework called LFCNN, in which two adjacent views are employed to generate the in-between view. In the successive work [25], some modifications are applied to the network structure but with the same input organization strategy as [26]. These methods can not make full use of the angular domain as only a couple of sub-aperture images around the novel view are fed into the network. Besides, it can only generate novel views at 2X upsampling factor.

Wu *et al.* [24] model view synthesis as learning-based angular detail restoration on 2D EPIs. A "blur-restoration-deblur" framework is presented that consists of three steps: firstly, the input EPI is convolved with a predefined blur kernel; secondly, a CNN is applied to restore the angular detail of the EPI damaged by the undersampling; finally, a non-blind deconvolution is operated to recover the spatial detail suppressed by the EPI blur. It achieves promising results on a variety of scenes, but there are still some shortcomings: the potential of the full LF data is underused; the operations of "blur-restoration-deblur" loop numerous times before all the in-between views are synthesized.

To sum up, the key insight of view synthesis is to make full use of the input views. To reduce the difficulty of collecting data, it is appropriate that the input views are regularly spaced on a grid. Besides, it is rather difficult to work on the high dimensional data with current CNN frameworks. In this paper, an end-to-end framework called Pseudo 4DCNN is proposed to efficiently synthesize novel views of densely sampled LF from sparse input views.

2 Methodology

2.1 Problem Formulation

In this paper, 4D LF data are denoted as $L(x, y, s, t)$ decoded from the LF raw image as depicted in Fig. 1. Each light ray is illustrated by the interactions with two parallel planes, travelling from the angular coordinate (s, t) on the main lens plane to the spatial coordinate (x, y) on the microlens array plane.

Given $n \times n$ sparse input views on a grid at the spatial resolution of $H \times W$, the goal of view synthesis for LF imaging is to restore a more densely sampled LF at the resolution of (H, W, N, N), where $N = f \times (n - 1) + 1$ and f is the upsampling factor in the angular dimension.

As shown in Fig. 1, EPI is a 2D slice of 4D LF by fixing one angular dimension and one spatial dimension. The framework proposed by Wu *et al.* [24] is based on restoration of 2D EPIs, enhancing one angular dimension s or t. 3D volume from 4D LF like $V_{t^*}(x, y, s)$ can be extracted by fixing one angular dimension $(t = t^*)$,

which consists of stacked 2D EPIs. To directly process 4D LF, we assemble 2D strided convolutions on stacked EPIs and sequential 3D CNNs connected with angular conversion to build Pseudo 4DCNN. The proposed framework is well formulated to be entirely differentiable, which makes the learning process more tractable. In the next section, the proposed framework is described in detail.

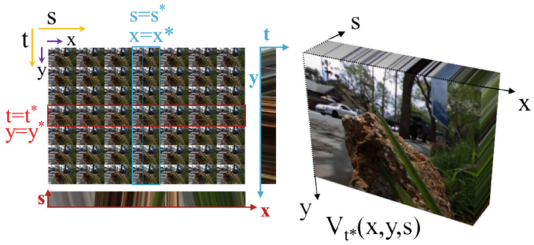

Fig. 1. 4D light fields $L(x, y, s, t)$. A horizontal EPI is a 2D (x, s) slice $L(x, y^*, s, t^*)$ by setting $y = y^*$ and $t = t^*$, and a vertical EPI (y, t) by setting $x = x^*$ and $s = s^*$. By analogy, 3D volume $V_{t^*}(x, y, s)$ can be extracted by setting $t = t^*$.

2.2 Proposed Framework

Overview. Given input sparse views $L_0(x, y, s, t)$ with the resolution of (H, W, n, n) depicted as Fig. 2, we fix one angular dimension $t = t^*, t^* \in \{1, 2, \ldots, n\}$ to extract 3D volume with the resolution of (H, W, n) as

$$V_{t^*}(x, y, s) = L_0(x, y, s, t^*) \tag{1}$$

$V_{t^*}(x, y, s)$ are interpolated as $V_{t^*}(x, y, s) \uparrow$ to the desired resolution (H, W, N) given the upsampling factor. The high-frequency details of $V_{t^*}(x, y, s) \uparrow$ are restored by the row network modeling as $F_r(\cdot)$, and then form the intermediate LF as

$$L_{inter}(x, y, s, t^*) = F_r(V_{t^*}(x, y, s) \uparrow) \tag{2}$$

Next, we perform angular conversion to transform from the angular dimension t to dimension s. By fixing $s = s^*, s^* \in \{1, 2, \ldots, N\}$, $V_{s^*}(x, y, t)$ are extracted from $L_{inter}(x, y, s^*, t)$ as

$$V_{s^*}(x, y, t) = L_{inter}(x, y, s^*, t) \tag{3}$$

with the resolution of (H, W, n), which is also interpolated to $V_{s^*}(x, y, t) \uparrow$ at the same resolution as $V_{t^*}(x, y, s) \uparrow$. The column network is then employed to recover details of $V_{s^*}(x, y, t) \uparrow$, modeling as $F_c(\cdot)$. Finally, the output $L_{out}(x, y, s, t)$ with the resolution of (H, W, N, N) are formed as

$$L_{out}(x, y, s^*, t) = F_c(V_{s^*}(x, y, t) \uparrow) \tag{4}$$

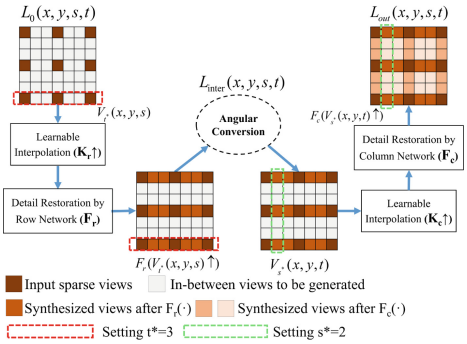

Fig. 2. Overview of the proposed framework Pseudo 4DCNN. Take reconstructing 7×7 LF data from 3×3 sparse views for example ($n = 3, N = 7, t^* = 3, s^* = 2$).

Learnable Interpolation on 3D Volumes. Volumes $V_{t^*}(x, y, s)$ and $V_{s^*}(x, y, t)$ consist of two spatial dimensions and one angular dimension as Fig. 1. Take $V_{t^*}(x, y, s)$ as an example, they can be regarded as n sub-aperture images with the resolution of (H, W), and are also composed of W stacked EPIs with the resolution of (H, n). Long *et al.* [13] state that upsampling can be performed using fractionally strided convolution. By reversing the forward and backward passes of convolution, the interpolation kernel for upsampling can be learned through end-to-end training with backpropagation. Rather than fixed interpolation on a single EPI, we introduce the learnable interpolation on stacked EPIs in 3D volumes using a deconvolutional layer as

$$V_{t^*}(x, y^*, s) \uparrow = deconv(V_{t^*}(x, y^*, s), f, K_r) \tag{5}$$

where $V_{t^*}(x, y^*, s)$ is a 2D EPI slice inside the 3D volume $V_{t^*}(x, y, s)$ by fixing $y = y^*$, f is the desired upsampling factor and K_r is the learnable kernel.

Another deconvolutional layer is employed to upsample $V_{s^*}(x, y, t)$ as

$$V_{s^*}(x^*, y, t) \uparrow = deconv(V_{s^*}(x^*, y, t), f, K_c) \tag{6}$$

As deconvolutional layers are differentiable, the learnable interpolation enables the proposed framework to be trained in an end-to-end strategy.

Detail Restoration Using 3D CNNs. 3D Convolutional Neural networks [15,19] are mostly applied to extract spatio-temporal features among frames for video analysis. Instead, we employ 3D CNNs and the residual learning [6] to recover high-frequency details of 3D volumes extracted from 4D LF.

In order to ensure efficiency, the proposed network $F_r(\cdot)$ and $F_c(\cdot)$ are of the same structure, which is lightweight and simple. As depicted in Fig. 3, both networks consist of two hidden layers followed by the sum of the predicted residual $\Re(V)$ and the input volume V as $F(V) = V + \Re(V)$. The first 3D convolutional layer comprises 64 channels with the kernel $5 \times 5 \times 3$, where each kernel operates

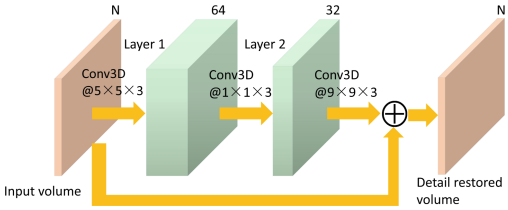

Fig. 3. Structure of the network for recovering details of 3D volumes. Layer 1 and layer 2 are followed by a rectified linear unit (ReLU). The final detail restored volume is the sum of the predicted residual and the input.

on 5×5 spatial region across 3 adjacent views inside V. Therefore, the size of filters W_1 in the first layer is $64 \times N \times 5 \times 5 \times 3$ and the size of bias b_1 is 64. Similarly, the second 3D convolutional layer comprises 32 channels with the kernel $1 \times 1 \times 3$, where each kernel operates on 1×1 spatial region (*i.e.* single pixel) across 3 adjacent slices of feature maps of the first layer. The size of filters W_2 in the second layer is $32 \times 64 \times 1 \times 1 \times 3$ and the size of bias b_2 is 32. The output layer of the network for the residual prediction utilizes $9 \times 9 \times 3$ filter, and thus the size of filters W_o in this layer is $N \times 32 \times 9 \times 9 \times 3$ and the size of bias b_o is N. Note that the first and second layers are activated by the rectified linear unit (ReLU), *i.e.* $\sigma(x) = \max(0, x)$, while the output layer is not followed by any activation layer. The residual prediction is formulated as

$$\Re(V) = W_o * \sigma(W_2 * \sigma(W_1 * V + b_1) + b_2) + b_o \tag{7}$$

where $*$ denotes the 3D convolution operation. To avoid border effects, we appropriately pad the input and feature maps before every convolution operations to maintain the input and output at the same size.

Prior Sensitive Loss Function. The proposed framework is designed to directly reconstruct the desired 4D LF. Rather than minimizing the L_2 distance between a pair of synthesized and ground truth images in [9] or between a pair of detail restored and ground truth EPIs in [24], the prior sensitive loss function is specifically formulated as follows:

$$E = \frac{1}{2N^2} \sum_{s^*=1,t^*=1}^{N} w_{s^*t^*} \| L_{gt}(s^*, t^*) - L_{out}(s^*, t^*) \|^2 \tag{8}$$

where the loss E is a weighted average over the entire mean squared errors (MSE) between the reconstructed L_{out} and ground truth L_{gt}.

Novel views generated in the later stage of the pipeline receive less prior information from the sparse input views as shown in Fig. 2. For instance, synthesized views after the row network $F_r(\cdot)$ are inferred from the input views, while a portion of those views synthesized after the column network $F_c(\cdot)$ only receive prior information propagated from earlier synthesized views. Hence, we design a prior sensitive scheme which pays more attention to the errors of the later synthesized

views by using larger weights. According to the order that views are generated and the level of received prior knowledge, all the synthesized views are divided into four groups and their MSE against the ground truth are summed up with corresponding weights. The weighting coefficient $w_{s^*t^*}$ for the synthesized view at (s^*, t^*) is particularly set as

$$w_{s^*t^*} = \begin{cases} \lambda_1 \ s^* \in [1:f:N], t^* \in [1:f:N] \\ \lambda_2 \ s^* \in [1:f:N], t^* \notin [1:f:N] \\ \lambda_3 \ s^* \notin [1:f:N], t^* \in [1:f:N] \\ \lambda_4 \ s^* \notin [1:f:N], t^* \notin [1:f:N] \end{cases} \tag{9}$$

Empirically, $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are set to 0.1, 1, 1 and 2 relatively.

Derivation on End-to-End Training. As the proposed system comprises 2D strided convolutions and detail-restoration 3D CNNs connected with angular conversion, it is non-trivial to train the networks with standard backpropagation. We analyze in detail that the proposed framework is entirely differentiable and all the weights can be recursively updated with standard backpropagation.

Firstly, we calculate the partial derivative of the loss E with respect to the intermediate LF $L_{inter}(x, y, s, t)$ using chain rule as

$$\frac{\partial E}{\partial L_{inter}(x, y, s, t)} = \frac{\partial E}{\partial L_{out}(x, y, s, t)} \cdot \frac{\partial L_{out}(x, y, s, t)}{\partial L_{inter}(x, y, s, t)} \tag{10}$$

according to Eq. 8, the first term on the right-hand side of Eq. 10 is derivable. The second term can be derived as

$$\frac{\partial L_{out}(x, y, s, t)}{\partial L_{inter}(x, y, s, t)} = \sum_{s^*=1}^{N} \frac{\partial L_{out}(x, y, s^*, t)}{\partial V_{s^*}(x, y, t) \uparrow} \cdot \frac{\partial V_{s^*}(x, y, t) \uparrow}{\partial V_{s^*}(x, y, t)} \tag{11}$$

The partial derivative of $L_{out}(x, y, s, t)$ with respect to $L_{inter}(x, y, s, t)$ is the sum of N partial derivatives of $L_{out}(x, y, s^*, t)$ with respect to $V_{s^*}(x, y, t)$. The first term on the right-hand side of Eq. 11 is apparently differentiable since it is the partial derivative of the output of the column network $F_c(\cdot)$ with respect to its input as Eq. 4. Further, the second term can be derived as

$$\frac{\partial V_{s^*}(x, y, t) \uparrow}{\partial V_{s^*}(x, y, t)} = \sum_{x^*=1}^{H} \frac{\partial V_{s^*}(x^*, y, t) \uparrow}{\partial V_{s^*}(x^*, y, t)} \tag{12}$$

the term on the right-hand side attributes its partial derivative to upsampling on EPIs using the learnable interpolation as Eq. 6. At this point, we have proved that the term on the left-hand side of Eq. 10 is differentiable.

The angular conversion operates on $L_{inter}(x, y, s, t)$ to transform from receiving output of the row network $F_r(V_{t^*}(x, y, s) \uparrow)$ to giving input of the column network $V_{s^*}(x, y, t)$, and thus there are no parameters in this component. Next,

we calculate the partial derivative of $L_{inter}(x, y, s, t)$ with respect to the input sparse LF $L_0(x, y, s, t)$ as

$$\frac{\partial L_{inter}(x, y, s, t)}{\partial L_0(x, y, s, t)} = \sum_{t^*=1}^{n} \frac{\partial L_{inter}(x, y, s, t^*)}{\partial V_{t^*}(x, y, s) \uparrow} \cdot \frac{\partial V_{t^*}(x, y, s) \uparrow}{\partial V_{t^*}(x, y, s)} \tag{13}$$

Similarly, it can be deduced that the first term on the right-hand side of Eq. 13 is differentiable because the numerator and the denominator of this term correspond exactly to the output and input of the row network $F_r(\cdot)$ as Eq. 2. The second term can be further derived as

$$\frac{\partial V_{t^*}(x, y, s) \uparrow}{\partial V_{t^*}(x, y, s)} = \sum_{y^*=1}^{W} \frac{\partial V_{t^*}(x, y^*, s) \uparrow}{\partial V_{t^*}(x, y^*, s)} \tag{14}$$

the term on the right-hand side of Eq. 14 is also differentiable since $V_{t^*}(x, y^*, s)$ is upsampled to $V_{t^*}(x, y^*, s) \uparrow$ by the learnable interpolation as Eq. 5. Overall, the partial derivative of the loss E with respect to the input $L_0(x, y, s, t)$ is derived as

$$\frac{\partial E}{\partial L_0(x, y, s, t)} = \frac{\partial E}{\partial L_{inter}(x, y, s, t)} \cdot \frac{\partial L_{inter}(x, y, s, t)}{\partial L_0(x, y, s, t)} \tag{15}$$

Considering Eqs. 10 and 13, it can be concluded that the proposed framework is entirely differentiable. Due to space limitations, formulations about the gradients of upsampling kernels and weights in the row or column network are not presented here.

2.3 Training Details

To train the proposed framework, we take over 300 LF samples with various lighting conditions, texture properties and depth variations through Lytro Illum and the lab-developed LF camera under indoor and outdoor environment. The LF raw images are decoded via Light Field Toolbox v0.4 [4]. LF images captured by Lytro Illum are with the spatial resolution 625×434 and angular resolution 9×9, while 729×452 and 11×11 respectively by the lab-developed LF camera.

Specifically, small patches in the same position of each view are extracted to formulate the training LF data. The spatial patch size is 48×48 and the stride is 20. If the angular upsampling factor is $3X$, we remove the border views and crop the original LF data to 7×7 views as ground truth, and then downsample to 3×3 views as the input. For $2X$ angular upsampling, the original LF data are just downsampled to 5×5 views. We cascade the proposed framework trained for $2X$ to deal with $4X$ angular upsampling.

In total, over 10^5 training samples are collected. Similar to other SR methods, we only process the luminance Y channel in the YCrcb color space. Since the proposed framework comprises two detail-restoration 3D CNNs connected with angular conversion, operating firstly in row or column pattern is prone to influence the accuracy of the final output. To alleviate such effect, we double the

training datasets by adding a copy of each LF sample with permuted angular dimensions.

The optimization of end-to-end training is conducted by the mini-batch momentum Stochastic Gradient Descent (SGD) method with a batch size of 64, momentum of 0.9 and weight decay of 0.001. The kernels of the learnable interpolation are initialized exactly like the bilinear upsampling. The filters of 3D CNNs are initialized from a zero-mean Gaussian distribution with standard deviation 0.01 and all the bias are initialized to zero. The learning rate is initially set to 10^{-4} and then decreased by a factor of 0.1 every 10 epochs until the validation loss converges. The proposed framework is implemented using Theano package [2] and proceeded on a workstation with an Intel 3.6 GHz CPU and a TiTan X GPU. Training takes within 8 h to converge.

3 Experimental Results and Applications

To validate the efficiency and effectiveness of the proposed framework, we compare with two recent state-of-the-art approaches, *i.e.* the depth-dependent method by Kalantari *et al.* [9] and depth-independent method by Wu *et al.* [24]. Experiments are carried out on various datasets for evaluating the robustness, including real-world scenes, synthetic scenes and biometrics data. The peak signal-to-noise ratio (PSNR), the gray-scale structural similarity (SSIM) and elapsed time per synthesized view are utilized to evaluate the algorithms numerically.

3.1 Real-World Scenes

As to experiments on real-world scenes, we follow the protocols in [24] to reconstruct 7×7 LF from 3×3 sparse views on *30 Scenes* captured by Lytro Illum in [9]. The performances of comparative methods [9,24] are obtained via implementing source codes released by respective authors, and the parameters are carefully tuned to maximize performances. For fair comparisons, all methods run in the GPU mode and are proceeded on the same workstation.

Figure 4 depicts quantitative comparisons of the average PSNR and elapsed time on *30 scenes* [9]. It is easy to find out that the proposed framework performs significantly better than other approaches: (1) greatly accelerates the process of view synthesis (0.28 s), (2) strongly improves the image quality of reconstructed 4D LF (43.28 dB). As demonstrated by numerical results, the proposed framework gains huge advantages in terms of both efficiency and effectiveness. Moreover, we conduct ablation experiments on *30 Scenes* with variants of Pseudo 4DCNN. As shown in Table 1, if F_c equals to F_r, the results decrease 0.38dB on average. Also, the results decrease 0.94 dB on average without prior sensitive loss. It can be demonstrated that each component of Pseudo 4DCNN definitely contributes to improving the performance.

For qualitative comparisons, we select two challenging outdoor scenes (*Rock, Flower*) containing complex depth variations and occlusions as shown in Fig. 5.

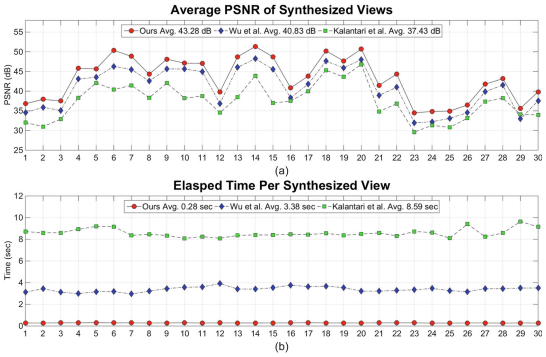

Fig. 4. Quantitative comparisons on real-world scenes (*30 scenes* [9]). The lateral axis represents scene No. from 1 to 30. (a) Average PSNR statistics. The proposed framework achieves **43.28** dB per LF scene on average, **5.85** dB higher than Kalantari *et al.* [9] (**37.43** dB) and **2.45** dB higher than Wu *et al.* [24] (**40.83** dB). (b) Elapsed time statistics. The proposed framework takes **0.28** s per synthesized view on average to reconstruct 7 × 7 LF from 3 × 3 views (angular upsampling factor $3X$) at the spatial resolution of 625 × 434, nearly **30X** faster than Kalantari *et al.* [9] (**8.59** s) and **12X** faster than Wu *et al.* [24] (**3.38** s). So the proposed framework greatly improves the accuracy and accelerates the process of view synthesis for LF imaging.

The depth-dependent method in [9] is not sensitive to depth changes in small areas, leading to large errors around object boundaries and depth discontinuities. The "blur-restore-deblur" scheme in [24] fails to reconstruct plausible details for small objects at a far distance, *e.g.* the white car in *Rock*, the background tree in *Flower*. As shown in the last column, our results are closer to ground truth. **Enlarge and view these figures on screen for better comparisons. See more comparisons in the supplement.**

Table 1. Quantitative comparisons on *30 Scenes* with variants of Pseudo 4DCNN.

	PSNR (dB)	SSIM
Pseudo-4DCNN full	**43.28**	**0.9916**
Same sub-networks ($F_r = F_c$)	42.90	0.9907
Without prior sensitive loss	42.34	0.9901
Without 2D learnable interpolation	40.15	0.9885
Without 3D detail-restoration CNN	39.01	0.9876

3.2 Synthetic Scenes

The synthetic experimental results are shown in Table 2, including two challenging scenes *Kitchen* and *Museum* from the LF datasets by Honauer *et al.* [7]. The spatial resolution is 512 × 512 and angular resolution 9 × 9. The central 7 × 7 views are extracted as ground truth, and 3 × 3 sparse views are taken as input.

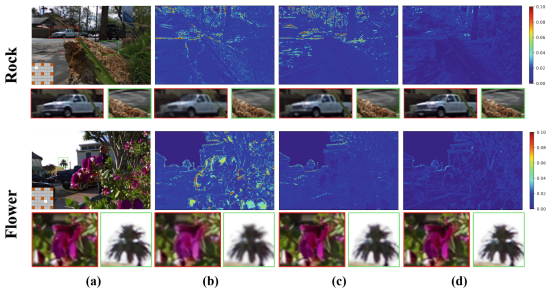

Fig. 5. Qualitative comparisons on *30 scenes* [9]. The ground truth view, error maps in the Y channel, and close-ups of image patches are presented. (a) Ground Truth. (b) Kalantari *et al.* [9] (c) Wu *et al.* [24] (d) Ours.

Transparent glasses in *Museum* and highlights in *Kitchen* are extremely difficult for view synthesis. The disparity estimator network in [9] fails to estimate reasonable disparities for non-Lambertian surfaces, especially at the boundaries. Significant artifacts are produced at the boundaries and geometry structures of transparent surfaces can not be preserved (Fig. 6). The method in [24] reconstructs better photo-realistic details than [9]. The proposed framework achieves the best performance, which is quite robust to specular reflection properties.

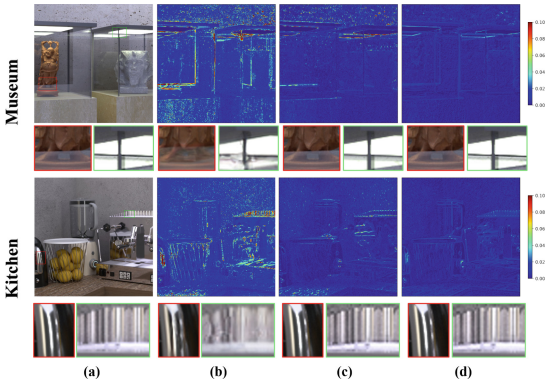

Fig. 6. Qualitative Comparisons on synthetic scenes. (*Kitchen* and *Museum*) (a) Ground Truth. (b) Kalantari *et al.* [9] (c) Wu *et al.* [24] (d) Ours.

Table 2. Quantitative comparisons on synthetic scenes (*Kitchen* and *Museum*).

	Kitchen		Museum	
	PSNR	SSIM	PSNR	SSIM
Kalantari *et al.* [9]	32.13	0.9156	30.45	0.9097
Wu *et al.* [24]	35.57	0.9360	34.98	0.9344
Ours	**38.12**	**0.9621**	**37.92**	**0.9559**

3.3 Application for Biometrics LF Data

For potential applications on biometrics, we capture a midsize LF dataset with over 200 face scenes (*Face*) and 100 iris scenes (*Iris*). *Face* is captured using Lytro Illum under natural lighting. Each scene contains 3 persons standing 0.2 m, 1 m, 3 m, and the faces are roughly 200 × 200, 100 × 100, 60 × 60 pixels on each sub-aperture image. *Iris* is captured under near infrared lighting with our lab-developed LF camera. Our LF camera is microlens-based and equipped with *240* um/*f4* microlens and a *135* mm/*f5.6* main lens. By setting the capturing distance 0.8 m, the iris on each sub-aperture image is around 90 pixels. We reconstruct 9 × 9 light fields from 5 × 5 sparse views on these biometrics LF data.

The bottleneck of view synthesis on *Face* is that severe defocus blur arises on the face outside depth of field (DOF). In Fig. 7, large errors around the eyes are occurred by [9] and the eyelid area are over-smoothed by [24]. Although LF cameras gain advantages in extending DOF that is beneficial for iris recognition, to obtain a LF iris image is greatly influenced by the specular reflection of the cornea region. Besides, the depth range of iris varies relatively small while the textures are very rich. As a consequence, [9] produces over-smoothed results without enough texture details on iris, and [24] recovers better textures of the iris but can not preserve details on the glossy area of the face. In contrast, the proposed framework obtains superior performances in both terms of efficiency and effectiveness (Table 3).

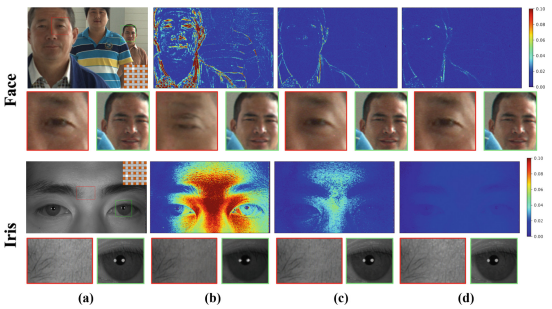

Fig. 7. Qualitative comparisons on biometrics LF data (*Face* and *Iris*). (a) Ground Truth. (b) Kalantari *et al.* [9] (c) Wu *et al.* [24] (d) Ours.

Table 3. Quantitative comparisons on biometrics LF data (*Face* and *Iris*).

	Face			Iris		
	PSNR	SSIM	Time (sec)	PSNR	SSIM	Time (sec)
Kalantari *et al.* [9]	29.50	0.8660	724.48	25.17	0.8235	904.02
Wu *et al.* [24]	40.04	0.9624	262.38	34.98	0.9344	339.71
Ours	**42.36**	**0.9869**	**23.49**	**40.14**	**0.9851**	**30.42**

3.4 Application for Depth Enhancement

We evaluate the accuracy and robustness of the proposed framework for depth enhancement. Table 4 shows quantitative comparisons on various scenes as well as challenging parts. It is observed that the depth maps with our reconstructed LF are roughly the same as depth maps with ground truth LF. The proposed framework will effectively contribute to depth enhancement of LF imaging.

Table 4. MSE statistics of depth estimation on the 4D LF benchmark [7] using the algorithm of Wang *et al.* [20].

Scenes		Wu *et al.* [24]	GT LF	**Ours**
Backgammon	Overall	0.1471	0.1307	**0.1181**
	Foreground fattening	0.1947	0.1680	**0.1601**
Pyramids	Overall	0.0214	**0.0191**	0.0193
	Pyramids	0.0117	0.0116	**0.0111**
Boxes	Overall	0.0512	**0.0497**	0.0507
	Fine surrounding	0.0312	**0.0287**	0.0303
Dino	Overall	0.0186	**0.0159**	**0.0159**
	Discontinuities	0.0174	**0.0161**	0.0163

4 Conclusion

In this paper, an end-to-end learning framework is proposed to directly synthesize novel views of dense 4D LF from sparse input views. To directly process high dimensional LF data, we assemble 2D strided convolutions operated on stacked EPIs and two detail-restoration 3D CNNs connected with angular conversion to build Pseudo 4DCNN. The proposed framework is well formulated to be entirely differentiable that can be trained with standard backpropagation. The proposed framework outperforms other SOTA approaches on various LF scenes. What's more, it greatly accelerates the process of view synthesis for LF imaging.

Acknowledgement. This work is funded by National Natural Science Foundation of China (Grant No. 61427811, 61573360) and National Key Research and Development Program of China (Grant No. 2016YFB1001000, No. 2017YFB0801900).

References

1. Adelson, E.H., Bergen, J.R.: The plenoptic function and the elements of early vision. In: Computational Models of Visual Processing, pp. 3–20. MIT Press (1991)
2. Al-Rfou, R., et al.: Theano: a Python framework for fast computation of mathematical expressions. arXiv preprint (2016)

3. Chaurasia, G., Duchene, S., Sorkine-Hornung, O., Drettakis, G.: Depth synthesis and local warps for plausible image-based navigation. ACM Trans. Graph. (TOG) **32**(3), 30 (2013)
4. Dansereau, D.G., Pizarro, O., Williams, S.B.: Decoding, calibration and rectification for lenselet-based plenoptic cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1027–1034 (2013)
5. Eisemann, M., et al.: Floating textures. In: Computer Graphics Forum, vol. 27, pp. 409–418. Wiley Online Library (2008)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
7. Honauer, K., Johannsen, O., Kondermann, D., Goldluecke, B.: A dataset and evaluation methodology for depth estimation on 4D light fields. In: Lai, S.-H., Lepetit, V., Nishino, K., Sato, Y. (eds.) ACCV 2016. LNCS, vol. 10113, pp. 19–34. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54187-7_2
8. Huang, F.C., Chen, K., Wetzstein, G.: The light field stereoscope: immersive computer graphics via factored near-eye light field displays with focus cues. ACM Trans. Graph. (TOG) **34**(4), 60 (2015)
9. Kalantari, N.K., Wang, T.C., Ramamoorthi, R.: Learning-based view synthesis for light field cameras. ACM Trans. Graph. (TOG) **35**(6), 193 (2016)
10. Levin, A., Durand, F.: Linear view synthesis using a dimensionality gap light field prior. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 1831–1838. IEEE (2010)
11. Levoy, M., Hanrahan, P.: Light field rendering. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, pp. 31–42 (1996)
12. Liu, F., Hou, G., Sun, Z., Tan, T.: High quality depth map estimation of object surface from light-field images. Neurocomputing **252**, 3–16 (2017)
13. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
14. Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P.: Light field photography with a hand-held plenoptic camera. Computer Science Technical report CSTR, vol. 2, no. 11, pp. 1–11 (2005)
15. Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view CNNs for object classification on 3D data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5648–5656 (2016)
16. Raytrix: 3D light field camera technology. http://www.raytrix.de/
17. Shi, L., Hassanieh, H., Davis, A., Katabi, D., Durand, F.: Light field reconstruction using sparsity in the continuous Fourier domain. ACM Trans. Graph. (TOG) **34**(1), 12 (2014)
18. Srinivasan, P.P., Wang, T., Sreelal, A., Ramamoorthi, R., Ng, R.: Learning to synthesize a 4D RGBD light field from a single image. In: International Conference on Computer Vision, ICCV, pp. 2262–2270 (2017)
19. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4489–4497 (2015)
20. Wang, T.C., Efros, A., Ramamoorthi, R.: Occlusion-aware depth estimation using light-field cameras. In: IEEE International Conference on Computer Vision, ICCV, pp. 3487–3495 (2015)

21. Wang, T.C., Efros, A.A., Ramamoorthi, R.: Depth estimation with occlusion modeling using light-field cameras. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) **38**(11), 2170–2181 (2016)
22. Wanner, S., Goldluecke, B.: Variational light field analysis for disparity estimation and super-resolution. IEEE Trans. Pattern Anal. Mach. Intell. **36**(3), 606–619 (2014)
23. Wu, G., et al.: Light field image processing: an overview. IEEE J. Sel. Top. Signal Process. (JSTSP) **11**, 926–954 (2017)
24. Wu, G., Zhao, M., Wang, L., Dai, Q., Chai, T., Liu, Y.: Light field reconstruction using deep convolutional network on EPI. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 1638–1646 (2017)
25. Yoon, Y., Jeon, H.G., Yoo, D., Lee, J.Y., Kweon, I.S.: Light-field image super-resolution using convolutional neural network. IEEE Signal Process. Lett. **24**(6), 848–852 (2017)
26. Yoon, Y., Jeon, H.G., Yoo, D., Lee, J.Y., So Kweon, I.: Learning a deep convolutional network for light-field image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 24–32 (2015)
27. Zhang, F.L., Wang, J., Shechtman, E., Zhou, Z.Y., Shi, J.X., Hu, S.M.: Plenopatch: patch-based plenoptic image manipulation. IEEE Trans. Vis. Comput. Graph. **23**(5), 1561–1573 (2017)
28. Zhang, Z., Liu, Y., Dai, Q.: Light field from micro-baseline image pair. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3800–3809 (2015)