Data-driven light field depth estimation using deep convolutional neural networks

Xing Sun[1], Zhimin Xu[2], Nan Meng[1], Edmund Y. Lam[1], and Hayden K.-H. So[1]

[1]Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong
[2]Tianjin Sharpnow Technology Co. Ltd., China

***Abstract*—This paper presents a data-driven approach to estimate the object depths from light field data using Convolutional Neural Networks (CNN). By exploring the relationship between the epipolar-plane images (EPI) and the corresponding depth map, we propose an enhanced EPI feature that encodes the depth information of each physical point in the light field and obtains the disparity map of the whole scene in a supervised manner. This work covers two major contributions, namely the extraction of the enhanced EPI features and the light field depth estimation with CNN. The proposed features augment the depth information of the corresponding points in the light field, and then our CNN architecture differentiates them into different depth layers. Forward propagation step of the CNN model allows rapid recognition of the disparity map of the test light field data. In the experiments, we apply our method on the HCI (Heidelberg Col-laboratory for Image Processing) benchmark dataset and demonstrate that it is significantly faster than the state-of-the-art light field depth estimation approaches while achieving satisfactory performance.**

I. Introduction

Recent advances in computational photography have enabled new features and functionalities in the imaging process, due to the powerful capacity of capturing more information beyond what could be obtained traditionally [1], [2]. As a specific example, conventional camera systems can merely provide a two-dimensional (2D) projection of a three-dimensional (3D) world, which is the integration of the radiance of light rays in the space. In contrast, with light field cameras, we have the ability to record both the radiance along the incoming light rays and their directions [3], [4].

Various methods have been proposed to extract depth information from light field data [5]–[10]. Wanner and Goldluecke [5], [6] estimated the depth map by extending the problem of stereo matching to the light field scenario. Other 3D cues, such as the defocus and correspondence information of an epipolar-plane image (EPI), are then used [7]. Later, Tao et al. [8] improved depth quality by further combining shading information and its angular coherence within the light-field data. Other methods, such as using the distortion of an EPI image [9] and using 3D reconstruction schemes [10], are also employed. All of these approaches have achieved relatively good depth labeling performances. However, high computational cost and difficulties in parameter tuning restrict these approaches from applications in broader scenarios (e.g., real-time processing).

Conventional 2D image depth estimation algorithms with data-driven approaches have been widely discussed in the computer vision community [11]–[13]. In this paper, we extend to the case of light field data. It has been pointed out that EPI images of light fields encode the depth information of the scene [6], [7]. That is, different slopes in the EPI images correspond to different depth layers. However, directly classifying EPI images is impossible to obtain a satisfactory depth map. Inspired by the deep learning input enhancement technique [14], we propose an EPI image enhancement algorithm, which increases the estimation accuracy.

In this paper, a data-driven light field depth estimation algorithm is presented, based on the traditional Convolutional Neural Networks (CNN) architecture. The major contributions and advantages of our method are as follows:

1) The proposed algorithm solves the light field depth estimation problem in a very computationally efficient way, with the help of the proposed EPI features and a well-trained CNN model.
2) Our method is adaptive for the light field data with various situations, such as different depth scaling and different number of sub-aperture views. When dealing with data capture using different camera systems, conventional depth estimation algorithms require careful parameter tuning for each camera setup. In contrast, our proposed approach is able to adapt different light field data via more training samples.

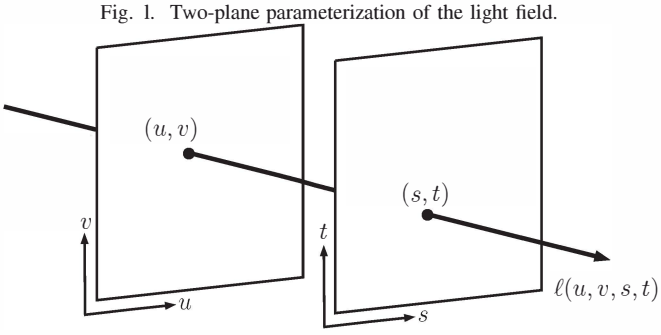
In the experiments, we show that our algorithm can achieve a satisfactory depth estimation performance on HCI dataset[1] and significant computational efficiency compared to the other state-of-the-art light field depth estimation algorithms [6], [7], [9].

This paper is organized as follows: Section II introduces the background knowledge and problem statements. In Section III, the proposed EPI image enhancement algorithm will be presented. Section IV presents our CNN architecture for depth estimation. Experimental results and comparisons are given in Section V, while Section VI draws several conclusions.

[1]HCI dataset can be downloaded at http://hciweb.iwr.uni-heidelberg.de/hci/softwares/light_field_analysis

978-1-5090-0620-5/16/$31.00

Fig. 1. Two-plane parameterization of the light field.



II. Background and Problem Statement

A. Light field representation

Radiance of light rays emitted from every 3D scene point are commonly represented by a 5D plenoptic function [15], $P(x, y, z, \theta, \phi)$, parameterized by three spatial coordinates (x, y, z) and two angles (θ and ϕ). Subsequently, two equivalent representations of the plenoptic function were proposed in the form of the light field [16] and the lumigraph [17]. They handle the case where the radiance along a ray in free space is a constant. Hence the plenoptic function is reduced to four dimensions.

Rays in a light field can be parameterized in a variety of ways. The most common of these is the two-plane parameterization, shown in Fig. 1. Specifically, each ray is encoded by its intersections with two parallel planes, that is, $\ell(u, v, s, t)$.

B. Epipolar-plane image

While given (s, t) a uv slice of a light field $\ell(u, v, s, t)$ corresponds to a single 2D view of the scene, a us slice for a fixed (v, t) corresponds to a so-called epipolar-plane image (EPI) [18]. Likewise a vt slice for a fixed (u, s) is also an EPI.

There is an important property of light field representation, which is that the coordinates (u, v, s, t) of the light rays emitting from a given physical point located at (x, y) have the following relationships,

$$\begin{aligned} u &= ks + (1 - k)x, \\ v &= kt + (1 - k)y, \end{aligned} \tag{1}$$

where

$$k = \frac{d - d_0}{d}. \tag{2}$$

The parameter d is the object depth measured from the object plane to the st plane, and d_0 is the distance between the uv and st planes. The slope k encodes the scene's depth. Fig. 2 illustrates the 2D version of the relationships. Therefore, the EPI is constructed by lines of different slopes. In the following, we will show how to use such embedded information to infer the depth of a 3D scene from the EPIs.

Here, we develop a data-driven method based on the EPI image [7]. In Fig. 3, the blue and red rectangles indicate two EPI images $g(v, s|u^1), g(v, s|u^2) \in \mathbb{R}^2$, which correspond to the blue and red horizontal lines in the light field image

Fig. 2. Light ray transmission in a 2D scene.

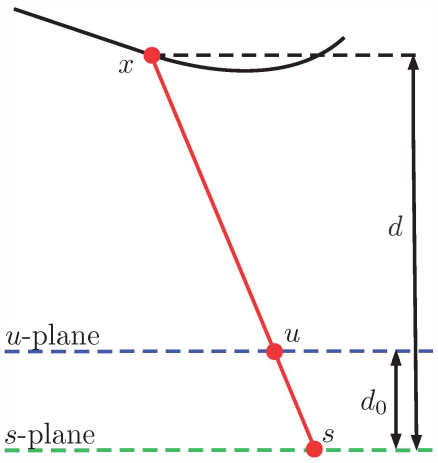
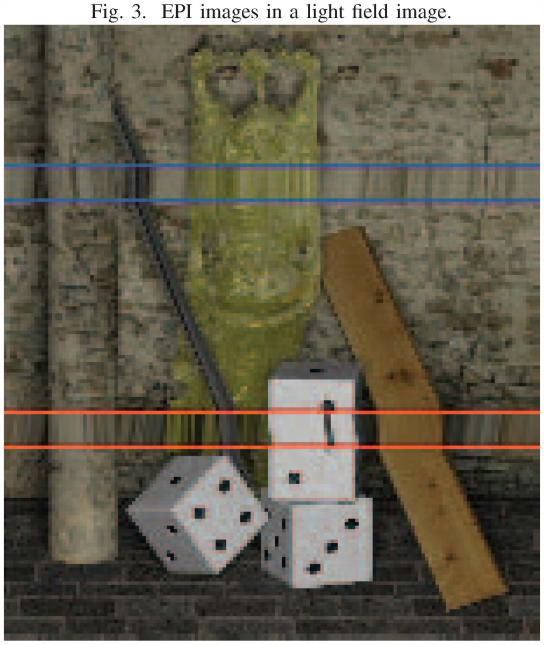


Fig. 3. EPI images in a light field image.



$f(v|u^1), f(v|u^2) \in \mathbb{R}$, where u, v, s, t are vertical, horizontal spatial coordinates and vertical, horizontal angular coordinates, and $f(\cdot)$ indicates the uv plane image, while $g(\cdot)$ indicates the EPI image on the us and vt planes.

III. Input EPI image enhancement

A. Edge detection

We focus on enhancing input EPI images for any physical point (x, y). In our CNN model, we use enhanced EPI images (i.e. $g(u, s|x, y)$) as input, which represents a single physical point (x, y). To enhance the input EPI images, we first apply the local contrast normalization [19], [20] and edge detection [21] to distinguish the slopes in the EPI images. Figure 4 demonstrates these EPI image enhancement steps. The first sub-figure shows an EPI image for the "Buddha"

Fig. 4. EPI image edge detection.

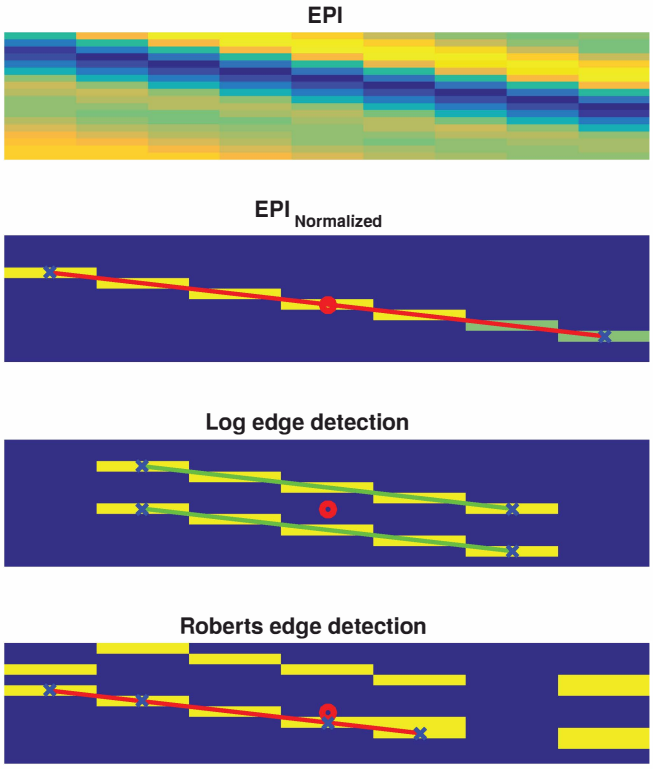
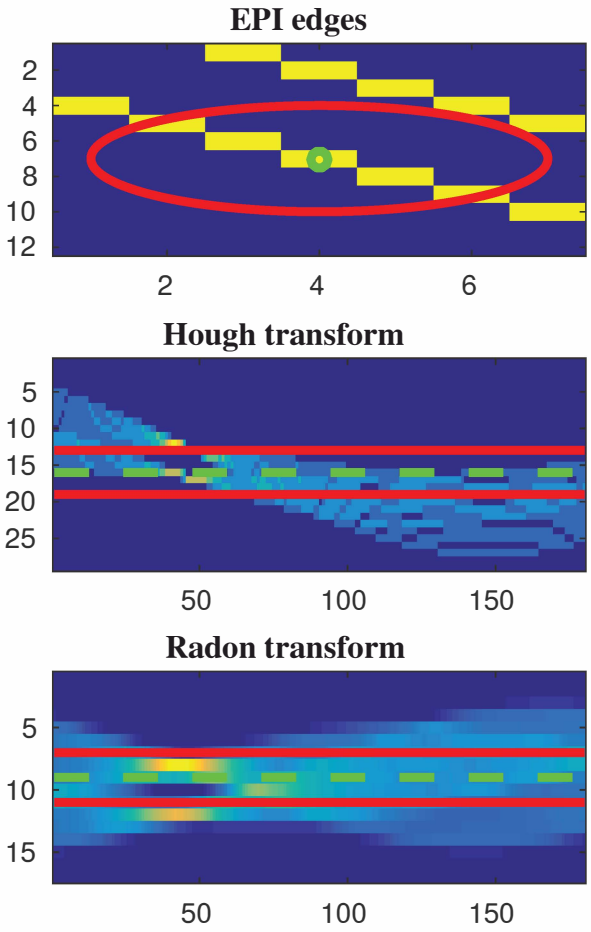


Fig. 5. Hough and Radon transforms for an EPI image.



dataset, while local contrast normalized EPI image is shown in the second sub-figure in Fig. 4 , with red point indicating the given physical location (x, y) where the slope through it becomes distinguished. The bottom two sub-figures show the edge detection features with log edge detection and Roberts edge detection algorithms, respectively.

B. Hough transform and Radon transform

The EPI images have been enhanced before inputting into the CNN, where the Hough transform [22] and the discrete Radon transform [23] have been applied on the EPI images. Figure 5 shows the details of the two transforms on the EPI image. Based on the relationship of us slice in Eq. (1), a polar representation can be derived where

$$s\cos\theta + u\sin\theta = \rho, \tag{3}$$

where $k\tan\theta = -1$ and ρ is proportional to the distance with given physical location x. Therefore, any slope θ also corresponds to a depth d. To better distinguish different slopes, as well as different depth layers, we want to enhance the input data in order to transfer the EPI image into the slope angular θ space. Here, we introduce two powerful transformation algorithms to enhance the input feature. First, we apply the Hough transform to transfer us slice into $\theta\rho$ space as shown in the second sub-figure. As the given physical location (x, y) is always at the center of the EPI image (i.e. green circle in the first sub-figure of Fig. 5), these lines can be the candidate slope only if they go through the closed region of the physical location (x, y) (i.e. red circle in the same sub-figure). The corresponding region in Hough transform is shown in a rectangle region (i.e. red rectangle in the second sub-figure of Fig. 5).

Second, the Radon transform is also applied on the EPI image $g(u, s)$ to present another slope θ space $R(\rho, \theta)$. The Radon transform is [23]

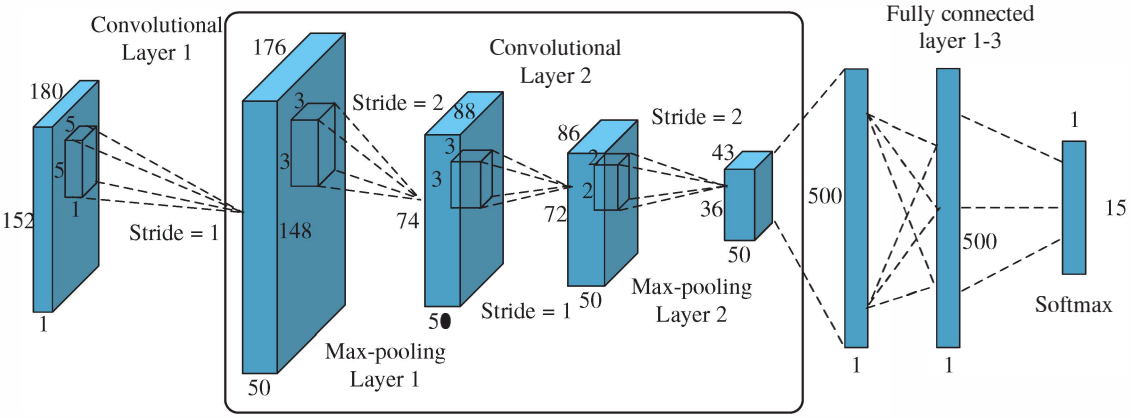
$$R(\rho,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(u,s)\delta(\rho - s\cos\theta - u\sin\theta)\,\mathrm{d}s\,\mathrm{d}u, \tag{4}$$

where $\delta(\cdot)$ is the delta function. Similarly, the candidate slope region in the Radon transform is also shown in a rectangle region (i.e. red rectangle in the bottom sub-figure of Fig. 5).

IV. Convolutional Neural Networks for Depth Estimation

Since the success of Krizhevsky et al. [24], CNNs have been applied on a variety of recognition and classification tasks [24], [25]. Traditionally, CNN is viewed as a multilayer neural network composed of one or more convolutional layers (often with a sub-sampling layer behind), followed by one or more fully connected layers. It has the advantages of being shift, scale, and distortion invariant [26]. These properties

Fig. 6. Convolutional neural networks framework.



are powerful for object recognition problems, which often require identifying different objects with various positions. Nevertheless, there is not much work on using such a model to estimate the depth of images. Directly feeding the EPI images into CNN is not feasible. With no reliable depth cues and difficulty in finding good representations of depth, CNN can hardly extract useful features for classification. Some previous work focuses on using geometric priors or additional manual features. Our proposed model is motivated by LeNet [26], which does not require any knowledge of viewing geometry.

A. The Architecture of CNN model

The CNN model, inspired by LeCun et al. [26] (LeNet-5), contains two convolutional layers and followed by three fully-connected layers. In general, the convolutional layer with a kernel size of k can be represented by

$$x_j^l = f(\sum_{i \in M_j} x_i^{l-1} * a_{ij}^l + b_j^l), \tag{5}$$

where M_j represents a selection of input layers, l is the layer index, and $\{a, b\}$ represent the weight and bias, respectively. The variable x is a patch of size $k \times k \times N$ (where k is the size of the convolutional kernel and N is the number of channels for input layer and the number of filters for intermediate layers). The function above shows an intuitive explanation for the property of shift and some degree of distortion invariance, which is helpful to distinguishably recognize random transform images. Most of the time, there is a sub-sampling layer, which can be described by

$$y_j^l = f(\beta_j^l * \text{DownSample}(x_j^{l-1}) + b_j^l), \tag{6}$$

where y is the sampled value within this layer, and β stands for the weight of the sub-sampling layer. The function DownSample$(\cdot)$ is the commonly used max-pooling function which can be explained as p-norm sub-sampling for $p \to \infty$ [27]. We also introduce the pooling layer mainly for two reasons:

1) To perform spatial dimension reduction to lower computational complexity.
2) To make the representation acquired from the CNN invariant for different training light fields. There are roughly four training light fields, and each training set contains over $200,000$ combined images generated by the method stated above while the test dataset contains $50,000$ combined images.

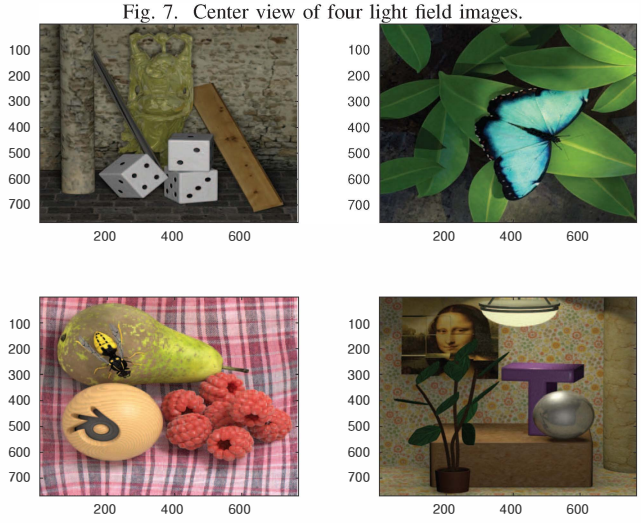
B. Algorithm procedure

We can see that an EPI image $g(u, s|x, y)$ has correlation among all points with spatial coordinate u and angular coordinate s. To reduce the redundancy, we crop with a window size $W_s \times W_a$ for spatial and angular coordinates, which is shown in Fig. 4. In our experiments, we set $W_s = 12$ and $W_a = 9$. In order to increase the differences among "features" of various depths, we extract the center region of the Radon transform domain. We focus on the range within $90°$ angles on both sides of the center point in Radon transform domain. Thus, the angular size of the input image is $180°$.

We use concatenated EPI images $g(u, s|x, y)$ and $g(v, t|x, y)$ as input image for the physical point (x, y). After applying the enhancement steps, the Hough transform of EPI image is 180×27 and the Radon transform of EPI image is 180×17. As shown in Fig. 5, we extract the candidate region of the Hough transform with size 180×11, and the Radon transform with size 180×5. With concatenating the Hough transform of the local contrast normalization, the log edge detection, the Roberts edge detection, and the Radon transform of the local contrast normalization, the local feature can be obtained with size 180×38. To achieve a global depth estimation, a mixture feature from physical point neighborhood around (x, y) with the mask

$$\mathcal{M} = \frac{1}{12} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \tag{7}$$

Fig. 7. Center view of four light field images.



Then, combining the local feature and mixture neighborhood feature, we obtain a 180×76 for us EPI image. The input image size 180×152, which is shown in Fig. 6, is obtained after concatenating us EPI and vt EPI features. Lastly, we apply the bilateral filter [28], [29] to smooth the output from the CNN, as shown in Fig. 6.

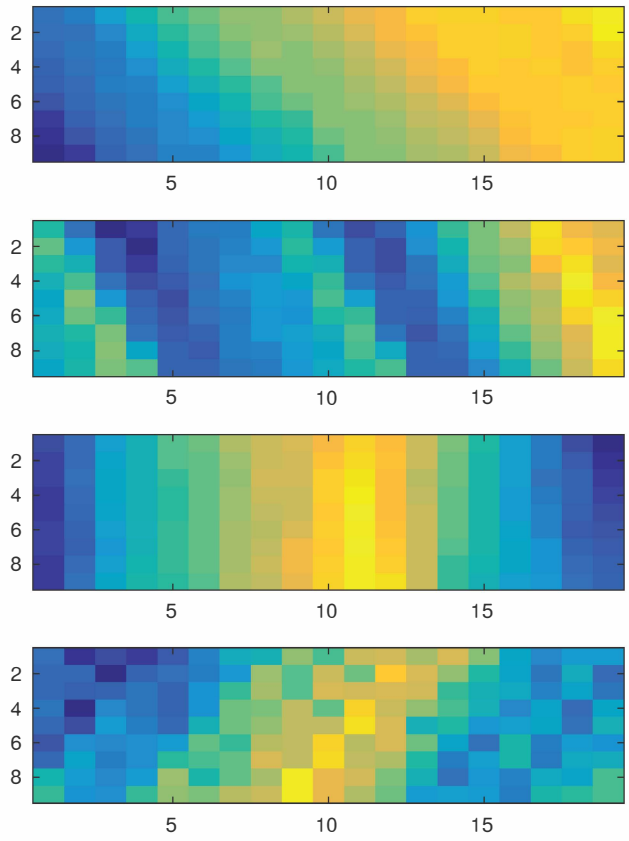
V. Experimental Results

We conduct experiments on synthetic HCI light field datasets, four of them are shown in Fig. 7. In each dataset, there are 9×9 views. The results from two representative datasets, Mona and Buddha, are illustrated in comparison with the ones using the method in [6].

A. Experimental setting

We make use of the HCI dataset for training the CNN model,which is composed of thirteen high quality densely sampled light fields [5]. Since all of the light field images contain ground truth, they can be used to evaluate the results of CNN. The whole dataset is divided into two categories, one is rendered synthetic dataset, the other one is real-world dataset sampled using single moving camera. In our work, a subset has been used to train the CNN model with leave-one-out cross validation policy. The selected subset contains 6 images of synthetic category, namely Buddha, Horses, Papillon, StillLife, Medieval and MonasRoom. Unlike real-world category images, these images shared similar depth ranges. We also drop the Buddha2 image to avoid the similarity of training and testing dataset. In our experiment, the CNN model is used to extract features from the input images. The whole model is shown in Fig. 6. For training the CNN network, we first randomly pick an image as test dataset, and then train the CNN with the rest 5 images. For instance, iteratively select 4 of remaining 5 images as training set and the left one as validation set, and feed the training set into CNN, then calculate the average result.

Fig. 8. EPI Feature extracted from the "Buddha" dataset with five depth layers



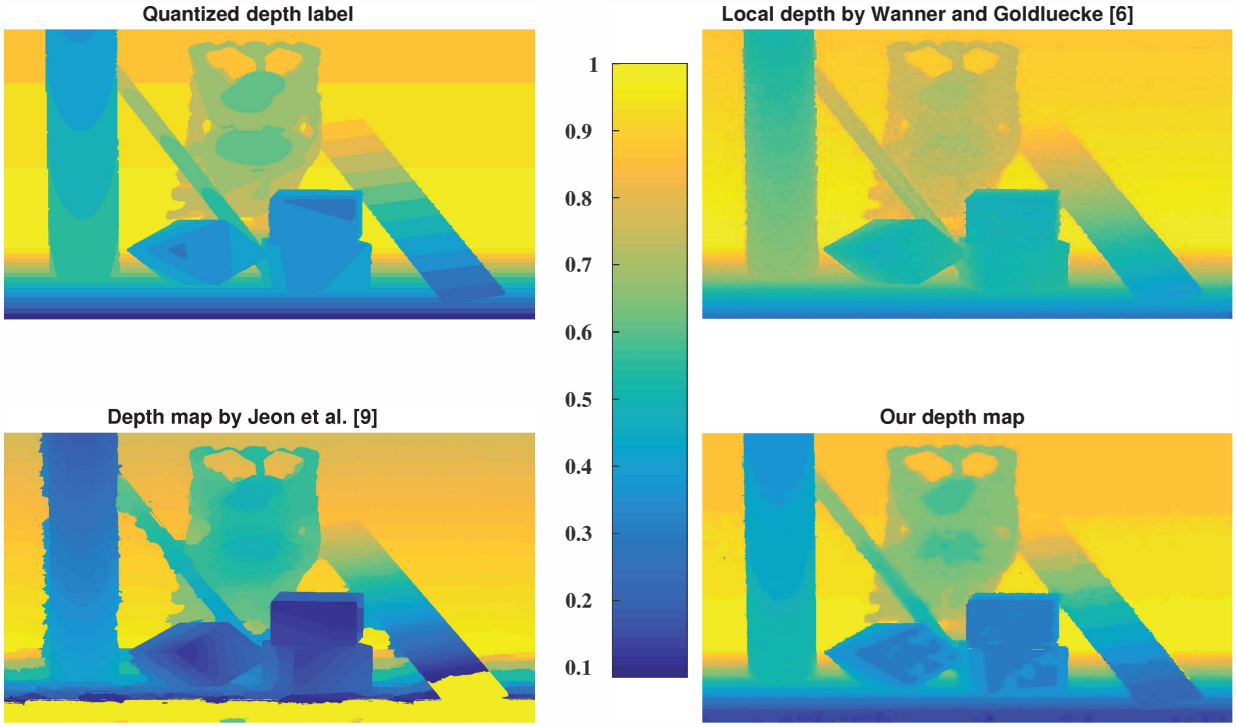
B. Buddha dataset

Figure 8 shows the extracted EPI feature for five different depth layers from "Buddha" dataset. Different depth layers correspond to different EPI image pattern, Fig. 8 illustrates that depth layer classification based on the input EPI feature is reasonable. Figure 9 demonstrates normalized depth estimation results of the center view on the Buddha dataset. The first sub-figure shows the quantized depth label with 15 different layers, while the second sub-figure demonstrates the estimated local depth map by Wanner and Goldluecke's method [6]. The fine-tuned depth map by Jeon et al. [9] is shown in the third sub-figure, and our estimated result is presented in the fourth sub-figure. Our data-driven depth estimation algorithm has achieved a satisfactory performance compared with other methods, although it also suffers some quantization problems, i.e. the depth transition of the right pillar is not quite smooth.

C. Mona dataset

Similar results on the Mona dataset are given in Fig. 10. In comparison with the depth result shown in the third sub-figure, our approach has a better performance on the complex details. For instance, our estimated depth is much clearer in the region around the flowerpot. In particular, the three stems are much clearer in our result, which in contrast are merged at the bottom in the third sub-figure. However, because the

Fig. 9. Our depth estimation result on the Buddha dataset compared with quantized depth map, local depth map by Wanner and Goldluecke [6] and the fine-tuned depth map by Jeon et al. [9]. The smaller value in the color bar shows near range and larger value indicates far range.



dependency of different pixels is quite weak in our approach, the background of our estimated depth map is a little noisy compared to other approaches. In summary, our approach achieved a comparable performance compared to the global depth estimation algorithms.

D. Estimation computational cost

Here, we compare the computational cost of our algorithm with other conventional light field depth estimation methods. To achieve a satisfactory depth estimation result, iterations and carefully tuning are required by conventional algorithms. For the training part of our CNN model, sufficient iteration steps are also demanded, a fine tuned CNN model can be trained for 30 hours with four training dataset. Fortunately, the testing procedure of the CNN model is a single forward step, which can be significant fast. We perform the experiments with the same computer, i.e., Intel (R) Core (TM) i7 CPU 920 @ 2.67GHz, 16 GB memory, and NVIDIA GeForce GTX 760. All the algorithms are tested on Matlab 2015b. We conduct all synthetic HCI dataset except the Buddha2 image, this is because Buddha2 is quite similar to Buddha. From results shown in Table I, both comparison methods take a lot of time to obtain the final results. In contrast, for the "Buddha" dataset, our algorithm only spends $68.233s$ to achieve a comparable depth estimation performance, which is 84 times faster than the algorithm in [9] and 10 times faster than the one in [7]. Moreover, the performance of our approach can potentially be improved by enlarging the training datasets, as the CNN forward step can likely be accelerated with better GPUs.

VI. Conclusions and Future Work

This paper presents an approach to estimate the depth map of the light field in a supervised manner. First, we propose an EPI feature enhancement algorithm to emphasize the depth information encoded in the EPI image, which can significantly improve the estimation accuracy. Furthermore, a well-designed CNN model rapidly distinguishes different EPI features with corresponding depth labels. Experimental results have shown that our data-driven approach provides satisfactory performance in estimating the light field depth map with much lower computational cost. As for future work, two major aspects of improvements can be done. First, we will increase the number of training samples and the number of depth layers to further increase the accuracy and the robustness of the model. Second, we will develop more a sophisticated network architecture to fit a larger amount of data.

References

[1] E. Y. Lam, "Computational photography: Advances and challenges," in *Tribute to Joseph W. Goodman*, ser. Proceedings of the SPIE, vol. 8122, August 2011, p. 81220O.

[2] ——, "Computational photography with plenoptic camera and light field capture: Tutorial," *Journal of the ●ptical Society of America A*, vol. 32, no. 11, pp. 2021–2032, November 2015.

Fig. 10. Our depth estimation result on the Mona dataset compared with quantized depth map, local depth map by Wanner and Goldluecke [6] and the fine-tuned depth map by Jeon et al. [9]. The smaller value in the color bar shows near range and larger value indicates far range.

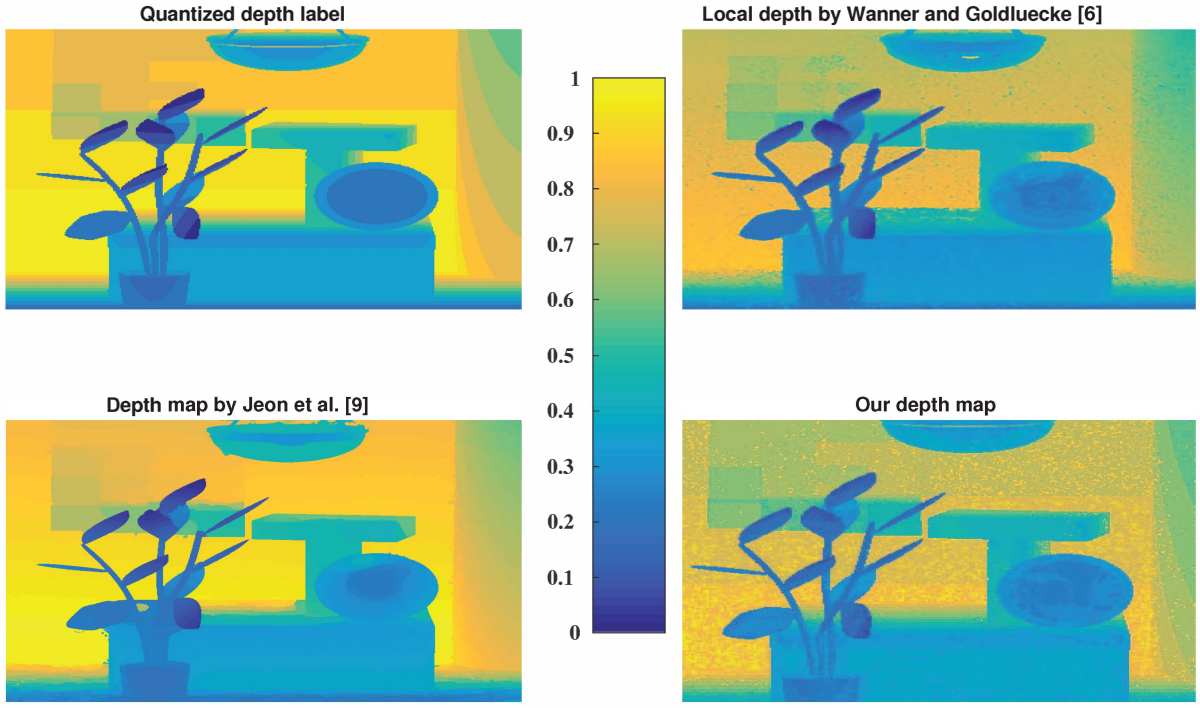


TABLE I
TIME CONSUMPTION COMPARISON

	Buddha	Pyramide	Horses	Papillon	StillLife	Mona
CNN forwarding test	**68.2s**	**67.7s**	**62.0s**	**67.2s**	**67.2s**	**65.4s**
Depth estimation [7]	629.1s	629.3s	628.3s	632.2s	629.0s	631.7s
Accurate estimation [9]	5713.1s	5625.7s	5627.4s	5661.0s	5621.6s	5774.7s

[3] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR*, vol. 2, no. 11, 2005.

[4] Z. Xu, J. Ke, and E. Y. Lam, "High-resolution lightfield photography using two masks," *Optics Express*, vol. 20, no. 10, pp. 10971–10983, 2012.

[5] S. Wanner and B. Goldluecke, "Variational light field analysis for disparity estimation and super-resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 606–619, 2014.

[6] ——, "Globally consistent depth labeling of 4D light fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 41–48.

[7] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras," in *IEEE International Conference on Computer Vision*, 2013, pp. 673–680.

[8] M. W. Tao, P. P. Srinivasan, J. Malik, S. Rusinkiewicz, and R. Ramamoorthi, "Depth from shading, defocus, and correspondence using light-field angular coherence," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1940–1948.

[9] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and I. S. Kweon, "Accurate depth map estimation from a lenslet light field camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1547–1555.

[10] S. Im, H.-G. Jeon, H. Ha, and I. S. Kweon, "Depth estimation from light field cameras," in *International Conference on Ubiquitous Robots and Ambient Intelligence*, 2015, pp. 190–191.

[11] H. Kwon, Y.-W. Tai, and S. Lin, "Data-driven depth map refinement via multi-scale sparse representation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 159–167.

[12] W. Zhuo, M. Salzmann, X. He, and M. Liu, "Indoor scene structure analysis for single image depth estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 614–622.

[13] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in Neural Information Processing Systems*, 2014, pp. 2366–2374.

[14] Z.-H. Ling, S.-Y. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. M. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, 2015.

[15] L. McMillan and G. Bishop, "Plenoptic modeling: An image-based rendering system," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1995, pp. 39–46.

[16] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1996, pp. 31–42.
[17] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1996, pp. 43–54.
[18] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision*, vol. 1, no. 1, pp. 7–55, 1987.
[19] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *IEEE International Conference on Computer Vision*, 2009, pp. 2146–2153.
[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.
[21] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.
[22] J. Illingworth and J. Kittle, "The adaptive Hough transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 5, pp. 690–698, 1987.
[23] G. Beylkin, "Discrete Radon transform," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 35, no. 2, pp. 162–172, 1987.
[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
[25] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *European Conference on Computer Vision*, 2014, pp. 512–528.
[26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[27] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.
[28] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1141–1151, 2002.
[29] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE International Conference on Computer Vision*, 1998, pp. 839–846.