Convolutional Networks for Shape from Light Field

Stefan Heber[1]
[1]Graz University of Technology
stefan.heber@icg.tugraz.at

Thomas Pock[1,2]
[2]Austrian Institute of Technology
pock@icg.tugraz.at

Abstract

Convolutional Neural Networks (CNNs) have recently been successfully applied to various Computer Vision (CV) applications. In this paper we utilize CNNs to predict depth information for given Light Field (LF) data. The proposed method learns an end-to-end mapping between the 4D light field and a representation of the corresponding 4D depth field in terms of 2D hyperplane orientations. The obtained prediction is then further refined in a post processing step by applying a higher-order regularization.

Existing LF datasets are not sufficient for the purpose of the training scheme tackled in this paper. This is mainly due to the fact that the ground truth depth of existing datasets is inaccurate and/or the datasets are limited to a small number of LFs. This made it necessary to generate a new synthetic LF dataset, which is based on the raytracing software POV-Ray. This new dataset provides floating point accurate ground truth depth fields, and due to a random scene generator the dataset can be scaled as required.

1. Introduction

A 4D light field [23, 14] provides information of all light rays, that are emitted from a scene and hit a predefined surface. Contrary to a traditional image, a LF contains not only intensity information, but also directional information. This additional directional information inherent in the LF implicitly defines the geometry of the observed scene.

It is common practice to use the so-called two-plane or light slab parametrization to describe the LF. This type of parametrization defines a ray by its intersection points with two planes, that are usually referred to as image plane $\Omega \subseteq \mathbb{R}^2$ and lens plane $\Pi \subseteq \mathbb{R}^2$. Hence the LF can be defined in mathematical terms as the mapping

$$L : \Omega \times \Pi \to \mathbb{R}, \qquad (\mathbf{p}, \mathbf{q}) \mapsto L(\mathbf{p}, \mathbf{q}) \,, \tag{1}$$

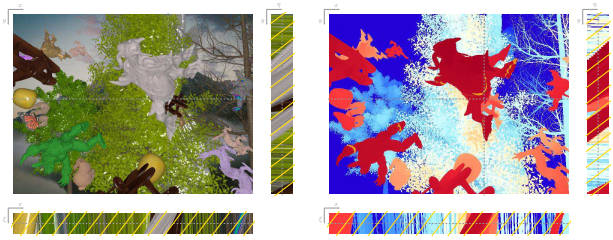
where $\mathbf{p} = (x, y)^\top \in \Omega$ and $\mathbf{q} = (\xi, \eta)^\top \in \Pi$ represent the spatial and directional coordinates.

There are different ways to visualize the 4D LF. One way of visualizing the LF is as a flat 2D array of 2D arrays, which can be arranged position major or direction major. The direction major representation can be interpreted as a set of pinhole views, where the viewpoints are arranged on a regular grid parallel to a common image plane (*c.f*. Figure 2). Those pinhole views are called sub-aperture images, and they clearly show that the LF provides information about the scene geometry. When considering Equation (1) a sub-aperture image is obtained by holding $\mathbf{q}$ fixed and by varying over all spatial coordinates $\mathbf{p}$. Another visualization of LF data is called Epipolar Plane Image (EPI) representation, which is a more abstract visualization, where one spatial coordinate and one directional coordinate is held constant. For example if we fix y and η, then we restricts the LF to a 2D function

$$\Sigma_{y,\eta} : \mathbb{R}^2 \to \mathbb{R}, \qquad (x, \xi) \mapsto L(x, y, \xi, \eta) \,, \tag{2}$$

that defines a 2D EPI. The EPI represents a 2D slice through the LF and it also shows that the LF space is largely linear, *i.e*. that a 3D scene point always maps to a 2D hyperplane in the LF space (*c.f*. Figure 1).

There are basically two ways of capturing a dynamic LF. First, there are camera arrays [34], that are bulky and ex-



(a) LF (b) depth field

Figure 1. Illustration of LF data. (a) shows a sub-aperture image with vertical and horizontal EPIs. The EPIs correspond to the positions indicated with dashed lines in the sub-aperture image. (b) shows the corresponding depth field, where red regions are close to the camera and blue regions are further away. In the EPIs a set of 2D hyperplanes is indicated with yellow lines, where corresponding scene points are highlighted with the same color in the sub-aperture representation in (b).

pensive, but allow to capture high resolution LFs. Second, more recent efforts in this field focus on plenoptic cameras [1, 24, 25], that are able to capture the LF in a single shot. Although LFs describe a powerful concept that is well-established in CV, commercially available LF capturing devices (*e.g*. Lytro, Raytrix, or Pelican) currently only fill a market niche, and are by far outweighed by traditional 2D cameras.

LF image processing is highly interlinked with the development of efficient and reliable shape extraction methods. Those methods are the foundation of all kinds of applications, like digital refocusing [18, 24], image segmentation [33], or super-resolution [4, 31], to name but a few. In the context of Shape from Light Field (SfLF) authors mainly focus on robustness w.r.t. depth discontinuities or occlusion boundaries. These occlusion effects occur when near objects hide parts of objects that are further away from the observer. In the case of binocular stereo occlusion handling is a tough problem, because it is basically impossible to establish correspondences between points that are observed in one image but occluded in the other image. In this case only prior knowledge about the scene can be used to resolve those problems. This prior knowledge is usually added in terms of a regularizer. In the case of multi-view stereo those occlusion ambiguities can be addressed by using the different viewpoints. This somehow suggests to select for each image position a subset of viewpoints that, when used for shape estimation, reduce the occlusion artifacts in the final result. In this paper we propose a novel method that implicitly learns the pixelwise viewpoint selection and thus allows to reduce occlusion artifacts in the final reconstruction.

Contribution. The contribution of the presented work is twofold. First, we propose a novel method to estimate the shape of a given LF by utilizing deep learning strategies. More specifically, we propose a method that predicts for each imaged scene point the orientation of the corresponding 2D hyperplane in the domain of the LF. After the pointwise prediction, we use a 4D regularization step to overcome prediction errors in textureless or uniform regions, where we use a confidence measure to gauge the reliability of the estimate. For this purpose we formulated a convex optimization problem with higher-order regularization, that also uses a 4D anisotropic diffusion tensor to guide the regularization.

Second, we present a dataset of synthetic LFs, that provides highly accurate ground-truth depth fields, and where scenes can be randomly generated. On the one hand, the generated LFs are used to train a CNN for the hyperplane prediction. On the other hand, we use the generated data to analyze the results and compare to other SfLF algorithms. Our experiments show that the proposed method works for synthetic and real-world LF data.

2. Related Work

Extracting geometric information from LF data is one of the most important problems in LF image processing. We briefly review publications most relevant in this field. As already mentioned in the introduction, LF imaging can be seen as an extreme case of a multi-view stereo system, where a large amount of highly overlapping views are available. Hence, it is hardly surprising, that the increasing popularity of LFs renewed the interest on specialized multi-view reconstruction methods [2, 3, 17, 8]. For instance, in [2] Bishop and Favaro proposed a multi-view stereo method, that theoretically utilizes the information of all possible combinations of sub-aperture images. Anyhow, the paper mainly focuses on anti-aliasing filters, that are used as a pre-possessing step for the actual depth estimation. In a further work [3] they propose a method that performs the matching directly on the raw image of a plenoptic camera by using a specifically designed photoconsistency constraint. Heber *et al*. [17] proposed a variational multi-view stereo method, where they use a circular sampling scheme that is inspired by a technique called Active Wavefront Sampling (AWS) [11]. In [8] Chen *et al*. introduced a bilateral consistency metric on the surface camera to indicate the probability of occlusions. This occlusion probability is then used for LF stereo matching.

Another, more classical way of extracting the depth information from LF data is to analyze the line directions in the EPIs [30, 13, 16]. Wanner and Goldluecke [30, 13] for example applied the 2D structure tensor to measure the direction of each position in the EPIs. The estimated line directions are then fused using variational methods, where they incorporate additional global visibility constraints. Heber and Pock [16] recently proposed a method for SfLF, that shears the 4D light field by applying a low-rank assumption. The amount of shearing then allows to estimate the depth map of a predefined sub-aperture image.

Unlike all the above mentioned methods, we suggest to train a CNN that allows to predict for each imaged 3D scene point the corresponding 2D hyperplane orientation in the LF domain. This is achieved by extracting information from vertical and horizontal EPIs around a given position in the LF domain, and feeding this information to a CNN. In order to handle textureless regions a 4D regularization is applied to obtain the final result, where a confidence measure is used to gauge the reliability of the CNN prediction. Our approach incorporates higher-order regularization, which avoids surface flattening. Moreover, we also make use of a 4D anisotropic diffusion tensor, that is calculated based on the intensity information in the LF. This tensor weights and orients the gradient during the optimization process.

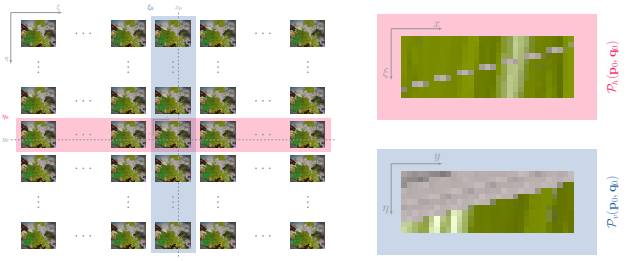


Figure 2. Illustration of the patch extraction. The figure to the left illustrates the LF as a direction major 2D array of 2D arrays, where the coordinate $(\mathbf{p}_0, \mathbf{q}_0)$ is marked. The corresponding vertical and horizontal patches at that location are shown to the right.

3. Methodology

In this section we describe the methodology of the proposed approach, that can be divided into three main areas: (1) Utilizing deep learning to predict 2D hyperplane orientations in the LF space (*c.f*. Section 3.1), (2) formulating a convex energy functional to refine the predicted orientations (*c.f*. Section 3.2), and (3) solving the resulting optimization problem using a first-order primal-dual algorithm (*c.f*. Section 3.3).

3.1. Hyperplane Prediction

The popularity of CNNs trained in a supervised manner via backpropagation [22] increased drastically after Krizhevsky *et al*. [21] utilized them effectively for the task of large-scale image classification. Inspired by the good performance on the image classification task, authors proposed numerous works, that apply CNNs to different CV problems including depth prediction [10], keypoint localization [15], edge detection [12], and image matching [35]. Zbontar and LeCun [35] for example proposed to train a CNN on pairs of small image patches, to predict stereo matching costs. Those costs were then refined using cross-based cost aggregation and semiglobal matching.

In the case of LF data the depth information of an imaged scene point is encoded in the orientation of the corresponding 2D hyperplane in the LF domain. In order to be able to predict this orientation we extract information from a predefined neighborhood of a given point $(\mathbf{p}, \mathbf{q}) \in \Omega \times \Pi$. More specifically, a training example comprises two image patches of size 31×11 centered at $(\mathbf{p}, \mathbf{q})$, where the first patch $\mathcal{P}_v(\mathbf{p}, \mathbf{q}) \subseteq \Sigma_{x,\xi}$ is extracted from the vertical EPI, and the second patch $\mathcal{P}_h(\mathbf{p}, \mathbf{q}) \subseteq \Sigma_{y,\eta}$ is extracted from the horizontal EPI. Note, that values outside the domain of the LF are set to zero. Figure 2 illustrates this patch extraction step. The figure shows a pair of horizontal and vertical patches, where the orientation of the line that intersects the center of the patch defines the orientation of the 2D hyperplane.

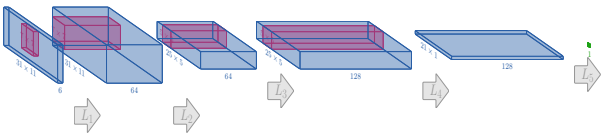


Figure 3. Illustration of the network architecture.

(a) 3D view (b) sub-aperture image

Figure 4. Illustration of a rendered LF. (a) provides a 3D view of a randomly generated scene, where foreground, midground, and image plane are highlighted in green, blue and purple. (b) shows a sub-aperture image of the obtained LF.

Network Architecture. The used network architecture is depicted in Figure 3. The network consists of five layers, denoted as L_i, $i \in [5]$ [1]. The first four layers are convolutional layers, followed by one fully-connected layer. Each convolutional layer is followed by a Rectified Linear Unit (ReLU) nonlinearity. The first and third layer is padded such that the width and height between input and output is not changing. The kernel size of the convolutional layers decreases towards deeper layers. More precisely, we us kernels of size 7×7 for the first two layers, and kernels of size 5×5 for the layers three and four. The number of feature maps also increases towards deeper layers, *i.e*. we use 64 feature maps for the first two layers and double them for the following two layers. Note, that there is no pooling involved in the used network architecture, and the inputs are two RGB image patches of size 31×11.

POV-Ray Dataset. Despite the success of CNNs, there are also some drawbacks. One main drawback is the need of huge labeled datasets, that can be used for the supervised training. In order to fulfill this requirement we generated a synthetic LF dataset using POV-Ray [28]. Compared to the widely used Light Field Benchmark Dataset (LFBD) [32], which is generated with Blender [5], POV-Ray allows to calculate floating point accurate ground truth depth maps without discretization artifacts. In order to be able to increase the dataset as required, we also implemented a random scene generator. This scene generator divides the entire scene in foreground, midground, and background, as illustrated in Figure 4(a). The foreground and midground

[1] Notation: $[N] := \{1, \dots, N\}$

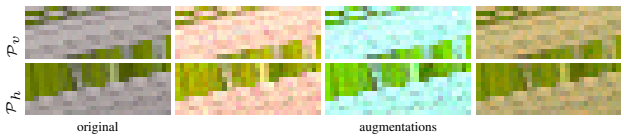


Figure 5. Illustration of data augmentation. The figure shows the original patches to the left and three different augmentation results to the right.

regions are randomly filled with comparatively small and large objects, respectively. Those objects are heavily occluding each other. The resulting occlusion and disocclusion effects lead to a high degree of hyperplane intersections in the LF domain. The used 3D objects for the foreground and midground are obtained from the Stanford 3D scanning repository [9], and from the Oyonale dataset [26]. We use around 20 different 3D objects, where about half of them come with random textures from categories like for instance *stone*, *wood*, or *metal*. Moreover, we also use random finish properties. Among other things those finish properties define the non-Lambertian reflectance characteristics of the different surfaces. The backgrounds of the scenes are represented by images downloaded from Google image search, that are labeled for reuse. We use background images with various resolutions from the categories *city*, *landscape*, *mountain*, and *street*.

After creating a random scene we render it from various viewpoints, where those viewpoints are placed on a regular grid (*c.f.* Figure 2 left). All rendered images use the same image plane, and the optical axes converge at a predefined point, that is chosen at random somewhere between the image plane and the background. Note, that due to the non-parallel viewing directions this results in non-perpendicular camera vectors, which is intended. Using this procedure we generate 25 LFs, where we use 20 to extract patches for training and 5 LFs are used for testing. The spatial resolution of the rendered LFs is set to 640×480, and the directional resolution is set to 11×11, which results in 121 sub-aperture images per LF.

Data Augmentation. Data augmentation is a widely used strategy to generalize neural networks [21, 10]. Although we could simply increase the dataset, augmentation during training seems to be important to avoid overfitting. The used augmentation includes changes in brightness, and color, as well as additive Gaussian noise. More specifically, the additive Gaussian noise has a sigma of 0.05. The multiplicative and additive color changes for each RGB channel are randomly sampled from the interval $[0.5, 2]$ and $[-0.15, 0.15]$, respectively. Figure 5 provides some augmentation examples.

Network Training. In order to train the CNN we make use of the caffe framework [19], where we use Adam [20] as the optimization method to minimize the Euclidean loss. From the 20 LFs rendered for training we extract 8e6 training examples, which are doubled using data augmentation. We pre-process each patch by subtracting the mean and dividing by the standard deviation of the pixel intensities. In order to monitor overfitting we use a test set of 2e6 examples. The results presented in this paper are obtained after 150k iterations of backpropagation.

3.2. Refinement Model

In order to refine the predicted orientations we utilize variational techniques and formulate the following optimization problem

$$\underset{\mathbf{u}}{\text{minimize}} \quad \mu\, \mathcal{D}(\mathbf{u}, \mathbf{f}) + \mathcal{R}(\mathbf{u})\,, \tag{3}$$

where $\mathbf{f}$ and $\mathbf{u}$ denote tensors of order four. The objective function in Equation (3) is a combination of the data term $\mathcal{D}(\mathbf{u}, \mathbf{f})$, that measures the fidelity of the argument $\mathbf{u}$ to the predicted measurements $\mathbf{f}$, and the regularization term $\mathcal{R}(\mathbf{u})$ that incorporates prior-knowledge about the solution. The scalar μ is used to balance the influence of the data term w.r.t. the regularization term.

The data term in our model ensures that the final solution is close to the predicted measurements of the CNN and is thus defined as

$$\mathcal{D}(\mathbf{u}, \mathbf{f}) = \frac{1}{2} \left\| \mathbf{c} \odot (\mathbf{u} - \mathbf{f}) \right\|_2^2 , \tag{4}$$

where $\mathbf{c}$ is a confidence measure (*c.f.* (8)), and $\odot$ denotes the Hadamard product.

The regularization term has to meet the challenges of removing artifacts and noise and simultaneously preserving sharp discontinuities in the sub-aperture images and in the EPIs as well. Common regularization terms are based on the first-order smoothness assumption. A famous example is the Total Variation (TV) semi-norm [29] given as $\mathrm{TV}(\mathbf{u}) = \|\nabla \mathbf{u}\|_1$. This type of regularization favors piecewise constant solutions an is thus well suited for intensity image denoising. However, when used for range data this property of the solution to be piecewise constant results in piecewise fronto-parallel depth reconstructions, which is not desirable. In order to avoid this effect in the spatial domain of the reconstruction we use a generalization of TV called Total Generalized Variation (TGV) introduced by Bredies *et al.* [6]. TGV of order k introduces higher order derivatives to incorporate smoothness from the first up to the k^{th} derivative. In other words, TGV of order k favors piecewise polynomial solutions of order $k-1$. For our purpose TGV of second order is sufficient, since most objects can be well approximated by piecewise affine surfaces. The

primal form of TGV of second order is given as

$$\mathrm{TGV}^2_\alpha(\mathbf{z}) = \min_{\mathbf{w}} \left\{ \alpha_1 \left\| \nabla \mathbf{z} - \mathbf{w} \right\|_1 + \alpha_0 \left\| \mathcal{E}\mathbf{w} \right\|_1 \right\}, \quad (5)$$

where $\mathcal{E}\mathbf{w}$ is the distributional symmetrized derivative of $\mathbf{w}$, and α_i, $i \in \{0, 1\}$, are weighting factors. The objective function in Equation (5) has the following intuitive interpretation. Before the TV of $\mathbf{z}$ is measured a vector field $\mathbf{w}$ of low variation is subtracted from the gradient. We choose to apply the TGV regularization w.r.t. the spatial coordinates $\mathbf{p}$ of the LF, and use a TV regularization w.r.t. to the directional coordinates $\mathbf{q}$ of the LF, which results in the following regularization term

$$\mathcal{R}(\mathbf{u}) = \mathrm{TGV}^2_\alpha(\mathbf{u}|_{x,y}) + \beta\, \mathrm{TV}(\mathbf{u}|_{\xi,\eta}), \quad (6)$$

where β is a scalar that allows to weight the TV component, and $\mathbf{u}|_{x,y}$ and $\mathbf{u}|_{\xi,\eta}$ denote the restrictions of $\mathbf{u}$ to the coordinates (x, y) and (ξ, η), respectively.

Assuming that intensity discontinuities in the LF correspond to depth discontinuities, we will make use of the intensity information to guide the regularization. More specifically, we will include an anisotropic diffusion tensor $\mathbf{\Gamma}$, that is calculated by analyzing the 4D structure tensor at each point $(\mathbf{p}, \mathbf{q})$ in the discrete domain of the LF. Therefore we will first calculate the eigenvalues λ_i and eigenvectors $\mathbf{v}_i$, $i \in [4]$, of the 4D structure tensor at position $(\mathbf{p}, \mathbf{q})$. Assuming that the eigenvalues are given in ascending order, $\lambda_1 \leqslant \ldots \leqslant \lambda_4$, the anisotropic diffusion tensor $\mathbf{\Gamma}(\mathbf{p}, \mathbf{q})$ is given as

$$\sum_{i \in [2]} \mathbf{v}_i \mathbf{v}_i^\top + \sum_{j \in [4] \setminus [2]} \exp(-\gamma \left\| \nabla L(\mathbf{p}, \mathbf{q}) \right\|^\delta) \mathbf{v}_j \mathbf{v}_j^\top, \quad (7)$$

where γ and δ adjust the magnitude and the sharpness of the tensor. $\mathbf{\Gamma}$ will orientate and weight the gradient direction during the optimization process, which leads to sharp depth transition at regions with high intensity differences. Note that a similar strategy was used in [17] to regularize the depth map of the 2D center view of the LF. The confidence measure $\mathbf{c}$ is also calculated based on the information derived from the structure tensor, *i.e.* the confidence at position $(\mathbf{p}, \mathbf{q})$ is calculated as

$$\mathbf{c}(\mathbf{p}, \mathbf{q}) = \sum_{i \in [3]} \sum_{j \in [4] \setminus [i]} (\lambda_i - \lambda_j)^2. \quad (8)$$

The final energy term combines the data term (4), the confidence measure (8), the regularization term (6), and the anisotropic diffusion tensor (7), and is given as

$$\min_{\mathbf{u},\mathbf{w}} \left\{ \frac{\mu}{2} \left\| \mathbf{c} \odot (\mathbf{u} - \mathbf{f}) \right\|_2^2 + \left\| \mathbf{\Gamma} \begin{bmatrix} \alpha_1 (\nabla \mathbf{u}|_{x,y} - \mathbf{w}) \\ \beta \quad \nabla \mathbf{u}|_{\xi,\eta} \end{bmatrix} \right\|_1 + \alpha_0 \left\| \mathcal{E}\mathbf{w} \right\|_1 \right\}. \quad (9)$$

3.3. Optimization

The optimization problem in Equation (9) is convex but non-smooth. In order to find a global optimal solution we will utilize the primal-dual algorithm proposed by Chambolle and Pock [7]. Therefore we reformulate (9) as the following convex-concave saddle-point problem

$$\min_{\mathbf{u},\mathbf{v}} \max_{\mathbf{d}_u, \mathbf{d}_w} \left\{ \frac{\mu}{2} \left\| \mathbf{c} \odot (\mathbf{u} - \mathbf{f}) \right\|_2^2 + \left\langle \mathbf{\Gamma} \begin{bmatrix} \alpha_1 (\nabla \mathbf{u}|_{x,y} - \mathbf{w}) \\ \beta \quad \nabla \mathbf{u}|_{\xi,\eta} \end{bmatrix}, \mathbf{d}_u \right\rangle - \chi_{B_\infty(0,1)}(\mathbf{d}_u|_{x,y}) - \chi_{B_\infty(0,1)}(\mathbf{d}_u|_{\xi,\eta}) + \langle \alpha_0\, \mathcal{E}\mathbf{w}, \mathbf{d}_w \rangle - \chi_{B_\infty(0,1)}(\mathbf{d}_w) \right\}, \quad (10)$$

where we introduced the dual variables $\mathbf{d}_u$ and $\mathbf{d}_w$. Moreover, we denote by $B_\infty(0, 1)$ the $\ell_{2,\infty}$ norm ball centered at zero with radius one, and χ_A denotes the characteristic function of a set A. The saddle-point formulation in Equation (10) allows to directly apply the primal-dual algorithm. Moreover, using adequate symmetric and positive definite preconditioning matrices as suggested in [27] the convergence speed of the algorithm can be further improved. Note however that the diagonal preconditioning results in dimension-dependent step lengths, instead of global step lengths, *i.e.* the global complexity of the algorithm does not change. The final algorithm is iterated for a fixed number of iterations or till a suitable convergence criterion is fulfilled. The involved gradient and divergence operators are approximated using forward/backward differences with Neumann and Dirichlet boundary conditions, respectively.

4. Experiments

In this section we will evaluate the proposed method on synthetic and real world LF data. For the synthetic evaluation we will use the test set of the generated POV-Ray dataset. For the real world evaluation we use the Stanford Light Field Archive (SLFA), that includes LFs captured with a multi-camera array [34], where each LF contains 289 sub-aperture images on a 17×17 grid.

Synthetic Evaluation. We start with the synthetic evaluation. When considering Figure 6, that provides network predictions and refinement results for two examples of the POV-Ray test set, we see that the CNN is able to predict reasonable 2D hyperplane orientations. The predicted orientations are accurate in well textured regions, but degrade in regions with less texture. Note that predicted orientations are barely effected by depth discontinuities. When comparing the network predictions with the refinement results, we see that the additional refinement model allows to reduce the errors in textureless regions and simultaneously preserves sharp depth discontinuities, as expected.

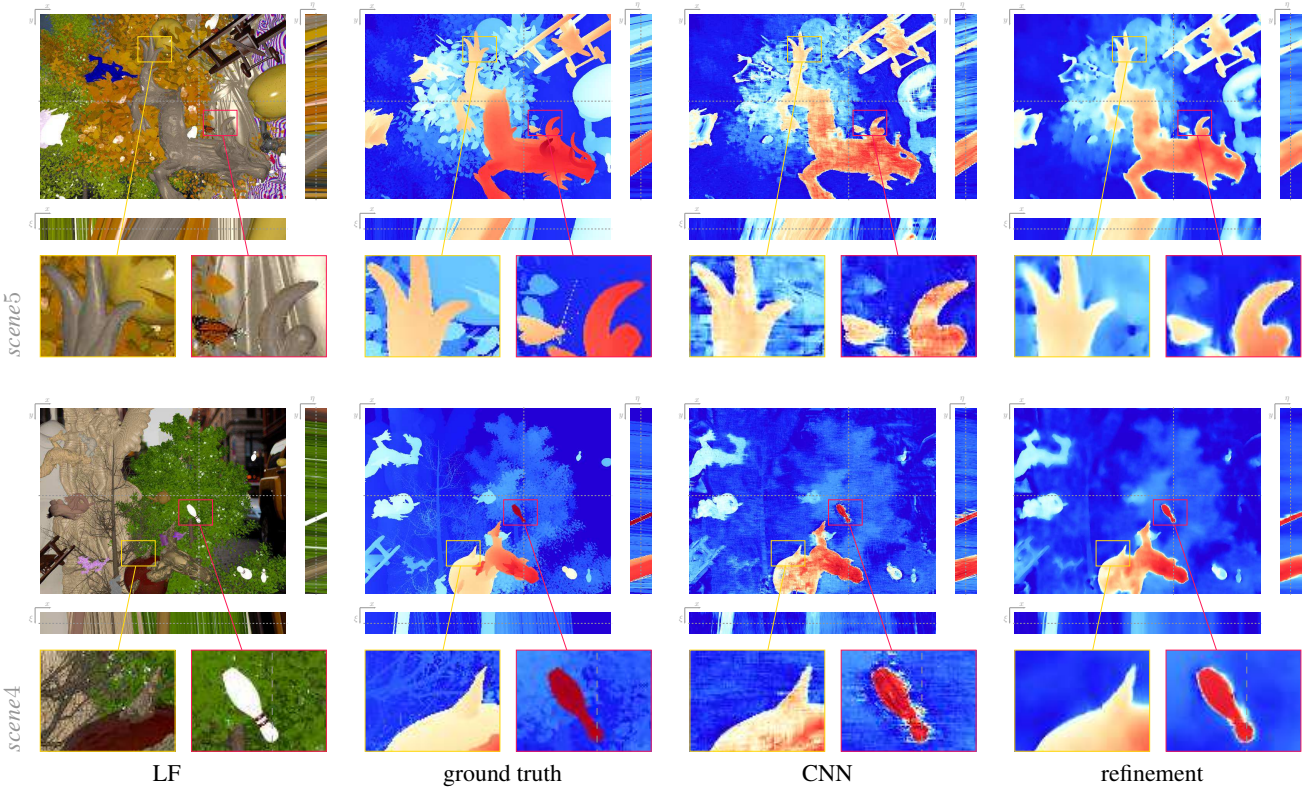


Figure 6. Illustration of reconstruction results for example scenes from the POV-ray dataset. The figure shows, from left to right, the LF data, the color-coded ground truth, the CNN prediction, and the refinement result.

In Figure 7 we compare our method to the works of Wanner and Goldluecke [30] and Heber and Pock [16]. The method by Wanner and Goldluecke [30] makes use of the EPI representation of the LF and calculates a globally consistent depth labeling. Heber and Pock [16] proposed a variational multi-view stereo model based on low rank minimization, where they use ideas from Robust Principal Component Analysis (RPCA), to define an all vs. all matching term. Compared to the method by Wanner and Goldluecke [30] we observe that the proposed method provides a more accurate reconstruction. This is mainly due to the fact that the proposed method provides continuous estimates and the method of Wanner and Goldluecke only provides a discrete depth labeling. Also note that the method by Wanner and Goldluecke fails if the hyperplane orientations are too close to the orientation of the xy plane. In this case the lines in the EPIs disconnect and the 2D structure tensor fails to estimate the correct orientation of the line. This is for example the reason for the large reconstruction errors of this method in scene1 (*c.f.* Figure 7). Compared to the variational model by Heber and Pock [16] the proposed method provides more details and sharper depth discontinuities for objects close to the camera. Hence the proposed method is especially useful in areas with severe occlusion effects. The method by Heber and Pock [16] suffers from a lose of detail mainly due to the required coarse to fine warping scheme.

Table 1. Quantitative results for the POV-Ray test set. The table shows the RMSE scaled by a factor of 100 for the different synthetic scenes shown in Figure 6 and Figure 7. Note, that the results for the methods proposed by Wanner and Goldluecke [30] and Heber and Pock [16] are obtained by running the source code provided by the authors.

#	Wanner and Goldluecke [30]	Heber and Pock [16]	CNN	proposed
1	2.1309	0.2501	0.2593	0.2575
2	0.6334	0.7610	0.5577	0.5202
3	0.2574	0.2094	0.2027	0.1847
4	0.9546	0.1760	0.1829	0.1408
5	0.6080	0.4903	0.4110	0.4018
	0.9168	0.3774	0.3227	0.3010

Quantitative results in terms of the root mean squared error (RMSE) are presented in Table 1 for the entire test set. When considering the results of the individual scenes we see that the proposed method is able to outperform the

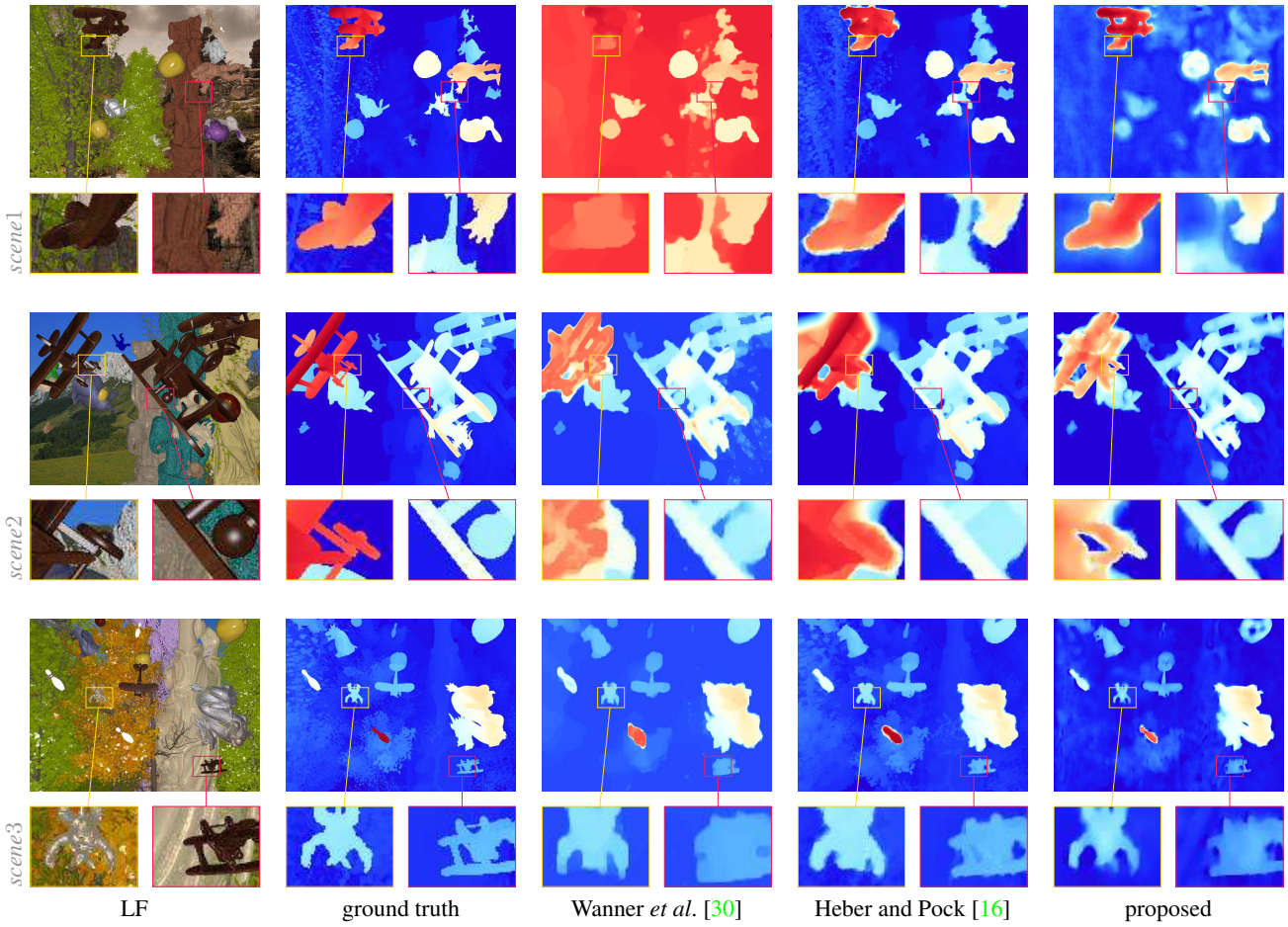


Figure 7. Comparison to state-of-the-art methods on the synthetic POV-Ray dataset. The figure shows, from left to right, the center view of the LF, the color-coded ground truth, the results for two state-of-the-art SfLF methods [30, 16], followed by the refinement result of the proposed method.

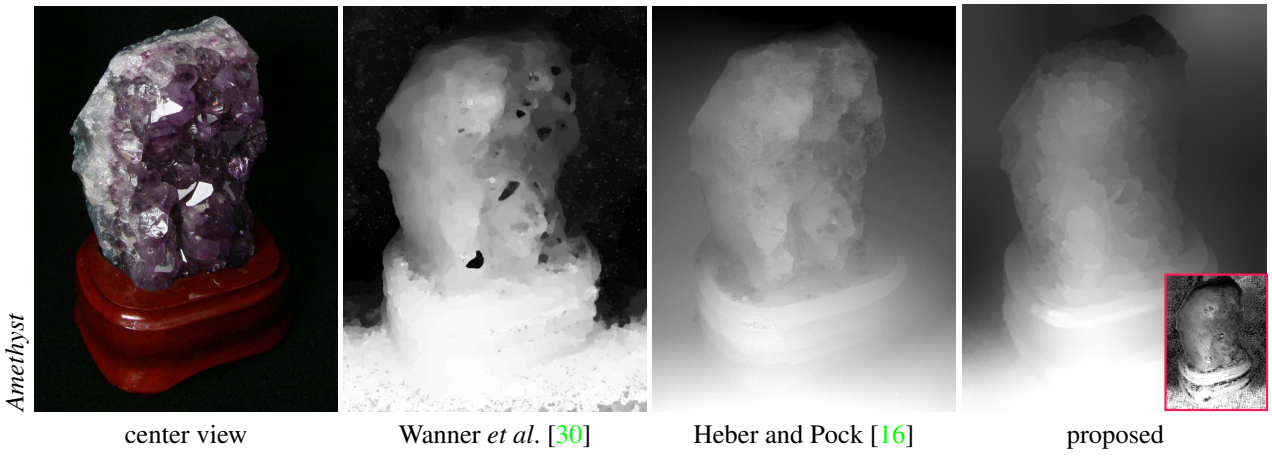


Figure 8. Qualitative comparison for a LF from the SLFA. The figure shows, from left to right, the center view of the LF, the results for two state-of-the-art SfLF methods [30, 16], and the refinement result of the proposed method, where we also show the network prediction in the purple sub-window.

competing methods on all but one scene. Furthermore, on average the proposed method is able to clearly outperform the other state-of-the-art methods. However, it should be emphasized that a main drawback of the presented method is to apply the CNN in a sliding window fashion, which results in considerable high computational costs.

Real World Evaluation. We continue with a short real world evaluation. Figure 8 provides a qualitative comparison of different SfLF methods, where a LF from the SLFA is used as input. Note that the scene is quite challenging because of the high degree of specularity. The result of the proposed method basically shows that the trained model can be applied to reconstruct real world LF data.

5. Conclusion

In this paper we proposed a novel method for SfLF. Our method is a combination of deep learning and variational techniques. We trained a CNN to predict 2D hyperplane orientations in the LF domain. Knowing these orientations allows to reconstruct the geometry of the scene. In addition to the learning approach we formulated a global energy optimization problem with a higher-order regularization to refine the network predictions. For numerical optimization of the variational model we use a first-order primal-dual algorithm. Overall the presented method demonstrates the possibility to use deep learning strategies for the task of shape estimation in the LF setting.

In order to provide enough data to train the network we generated a synthetic dataset by using the raytracing software POV-Ray. To generate an arbitrary amount of scenes we also implemented a random scene generator. The generated data was not just used to train the CNN, but also to provide quantitative and qualitative comparisons to existing SfLF methods. The qualitative evaluation of reconstruction results of synthetic and real world LF data showed that the proposed method is able to provide accurate reconstructions with sharp depth discontinuities. Moreover, our quantitative experiments showed that the proposed method is able to outperform existing methods on the POV-Ray test set in terms of the RMSE.

Acknowledgment. This work was supported by the FWF-START project *Bilevel optimization for Computer Vision*, No. Y729 and the Vision+ project *Integrating visual information with independent knowledge*, No. 836630.

References

[1] E. H. Adelson and J. Y. A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, 1992. 2

[2] T. Bishop and P. Favaro. Plenoptic depth estimation from multiple aliased views. In *12th International Conference on Computer Vision Workshops*, pages 1622–1629. IEEE, 2009. 2

[3] T. E. Bishop and P. Favaro. Full-resolution depth map estimation from an aliased plenoptic light field. In *Proceedings of the 10th Asian Conference on Computer Vision - Volume Part II*, pages 186–200, Berlin, Heidelberg, 2011. Springer-Verlag. 2

[4] T. E. Bishop and P. Favaro. The light field camera: Extended depth of field, aliasing, and superresolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):972–986, 2012. 2

[5] Blender. https://www.blender.org. 3

[6] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010. 4

[7] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011. 5

[8] C. Chen, H. Lin, Z. Yu, S. Bing Kang, and J. Yu. Light field stereo matching using bilateral statistics of surface cameras. June 2014. 2

[9] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 303–312, New York, NY, USA, 1996. ACM. 4

[10] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2366–2374. Curran Associates, Inc., 2014. 3, 4

[11] F. Frigerio. *3-dimensional Surface Imaging Using Active Wavefront Sampling*. PhD thesis, Massachusetts Institute of Technology, 2006. 2

[12] Y. Ganin and V. Lempitsky. n^4-fields: Neural network nearest neighbor fields for image transforms. In D. Cremers, I. Reid, H. Saito, and M.-H. Yang, editors, *Computer Vision – ACCV 2014*, volume 9004 of *Lecture Notes in Computer Science*, pages 536–551. Springer International Publishing, 2015. 3

[13] B. Goldluecke and S. Wanner. The variational structure of disparity and regularization of 4d light fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[14] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *SIGGRAPH*, pages 43–54, 1996. 1

[15] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localiza-

tion. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[16] S. Heber and T. Pock. Shape from light field meets robust PCA. In *Proceedings of the 13th European Conference on Computer Vision*, 2014. 2, 6, 7

[17] S. Heber, R. Ranftl, and T. Pock. Variational Shape from Light Field. In *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2013. 2, 5

[18] A. Isaksen, L. McMillan, and S. J. Gortler. Dynamically reparameterized light fields. In *SIGGRAPH*, pages 297–306, 2000. 2

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *CoRR*, abs/1408.5093, 2014. 4

[20] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 4

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 3, 4

[22] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Comput.*, 1(4):541–551, Dec. 1989. 3

[23] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 31–42, New York, NY, USA, 1996. ACM. 1

[24] R. Ng. *Digital Light Field Photography*. Phd thesis, Stanford University, 2006. 2

[25] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a hand-held plenoptic camera. Technical report, Stanford University, 2005. 2

[26] Oyonale. http://www.oyonale.co. 4

[27] T. Pock and A. Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *International Conference on Computer Vision (ICCV 2011)*, 2011. 5

[28] Pov-ray. http://www.povray.org. 3

[29] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, Nov. 1992. 4

[30] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4D lightfields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 2, 6, 7

[31] S. Wanner and B. Goldluecke. Spatial and angular variational super-resolution of 4d light fields. In *European Conference on Computer Vision (ECCV)*, 2012. 2

[32] S. Wanner, S. Meister, and B. Goldluecke. Datasets and benchmarks for densely sampled 4d light fields. In *Vision, Modelling and Visualization (VMV)*, 2013. 3

[33] S. Wanner, C. Straehle, and B. Goldluecke. Globally consistent multi-label assignment on the ray space of 4d light fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[34] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *ACM Trans. Graph.*, 24(3):765–776, July 2005. 1, 5

[35] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. *CoRR*, abs/1409.4326, 2014. 3