Generative Models for Generic Light Field Reconstruction

Paramanand Chandramouli*, Kanchana Vaishnavi Gandikota*, Andreas Goerlitz, Michael Moeller

E-mail: {paramanand.chandramouli, kanchana.gandikota, andreas.goerlitz, andreas.kolb, michael.moeller}@uni-siegen.de

Abstract—Recently deep generative models have achieved impressive progress in modeling the distribution of training data. In this work, we present for the first time generative models for 4D light field patches using variational autoencoders to capture the data distribution of light field patches. We develop two generative models, a model conditioned on the central view of the light field and an unconditional model. We incorporate our generative priors in an energy minimization framework to address diverse light field reconstruction tasks. While pure learning-based approaches do achieve excellent results on each instance of such a problem, their applicability is limited to the specific observation model they have been trained on. On the contrary, our trained light field generative models can be incorporated as a prior into any model-based optimization approach and therefore extend to diverse reconstruction tasks including light field view synthesis, spatial-angular super resolution and reconstruction from coded projections. Our proposed method demonstrates good reconstruction, with performance approaching end-to-end trained networks, while outperforming traditional model-based approaches on both synthetic and real scenes. Furthermore, we show that our approach enables reliable light field recovery despite distortions in the input.

1 Introduction

High quality light field (LF) images are vital for a wide range of applications, such as the precise free viewpoint rendering of a 3D scene or the estimation of geometries or materials of objects in a scene. Mathematically, light fields are represented using the plenoptic function that models the radiance of the scene in spatial and angular dimensions. Unfortunately, the acquisition of high quality light field data is commonly restricted by specific constraints imposed by the underlying camera hardware. Light field images can be acquired using exhaustive and expensive hardware setups comprising dozens of cameras in a camera-rig, or by using plenoptic cameras that utilize microlens arrays placed in front of the imager of a standard 2D camera [1]. While camera-rigs allow for larger baselines with rather sparse angular resolution, plenoptic cameras allow recording dense light fields with a rather small baseline. Plenoptic cameras have the advantage that they capture a full light field with a single exposure, but there is a trade-off between the spatial resolution of each sub-aperture image and the angular resolution of the micro images.

To address the trade-off between spatial and angular resolution optimally, researchers have proposed to linearly compress the angular or spatial dimension (or both), giving rise to the important problem of recovering a light field I from linear observations i related via

\[ i = \Phi I + n, \]

for a (problem dependent) linear operator \( \Phi \) and additive noise \( n \).

A classical approach to solve the ill-posed inverse problem (1) is by energy minimization methods. One designs a cost function \( H \) depending on the light field in such a way that low values of \( H(I) \) correspond to light fields \( I \) with desirable properties. Subsequently, the solution is determined by finding the argument that minimizes the energy \( H \), for example [2]. Alternate traditional approach is to estimate parameters such as depth map or disparity map which are subsequently used to synthesize light field [3].

Recent approaches have instead simulated large numbers of pairs \((i, I)\) and learned a mapping from \( i \) to \( I \) by a deep neural network, see [4], [5], [6], [7], [8]. While such approaches often improve the reconstruction quality in a specific application significantly, they lack the flexibility of classical methods and have to be retrained as soon as the observation model [1] changes.

To exploit the expressive power of neural networks without loosing the flexibility of energy minimization methods several hybrid methods have been proposed, e.g. by using neural networks as proximal operators (often also referred to as plug-and-play priors, see e.g. [9], [10]), using the parameterization of convolutional neural networks as a regularizer [11], or optimizing over the latent space of a generative model trained on representing the desired type of solutions, see e.g. [12], [13]. Interestingly, such approaches have not yet been exploited for light field reconstruction problems arising from [1], most likely due to the high complexity of light field data.

In this paper, we introduce for the first time, generative models for light field data for generic reconstruction. The key idea is to model the distribution of light fields using a class of generative autoencoders [14]. Once the training is complete, we use the generative models as priors in different light field reconstruction problems in an energy minimization framework. Due to the high complexity and variability of the light field data, generating light fields in a consistent fashion is highly challenging. In this paper,
we consider only small baseline light fields and we address this challenge by training generative models for light field patches instead of entire light fields. The advantage of our approach is that models learned on patches can readily generalize to a variety of scene classes, while being small enough to be amenable for training.

We propose two ways of learning the representation of light field patches: (i) a variational autoencoder conditioned on the central view (CVAE) (ii) an unconditioned variational autoencoder (VAE) which can directly generate samples of light field patches. Fig. 1(d) shows the schematic of the CVAE. The CVAE, consists of an encoder \( E_1 \) that takes a patch of light field as input and returns a low-dimensional latent code \( z \). The generator \( G_1 \) maps this latent code back to the light field patch. A convolutional feature extractor Fig. 1(c) provides features of the central view of the light field patch as an additional input to both the encoder and generator of the CVAE. Consequently, both the encoder and the generator utilize the information from the central patch. In the reconstruction of the light field patch shown in Fig. 1(d), we observe that the generator can map the encoded latent variable along with the features of the central view to a light field patch which looks similar to the input patch. This indicates that the encoder has learned to encode properties such as disparity and occlusion in the latent space, such that the generator can reconstruct the light field patch just from this latent code and the central view features. In our unconditional VAE, the encoder and generator are trained without conditioning them on the central view.

We solve different LF reconstruction problems using our generative models namely, view synthesis, spatial angular super resolution and coded aperture to demonstrate the flexibility of our approach. We illustrate the efficacy of the CVAE in different LF reconstruction tasks when the central view is given. When the central view is unavailable, we show that using both the models is vital for good reconstruction. As previous hybrid methods are not straightforward to extend to the usage of multiple generative models, we propose a new energy minimization method that simultaneously exploits both of our generative autoencoders as regularizers. Experimental results indicate that our approach performs close to end-to-end trained networks trained for a specific LF reconstruction tasks, while retaining the flexibility to address different reconstruction tasks. Moreover, our approach can effectively handle different distortions and noise in inputs while learning-based approaches cannot handle such variations without retraining.

2 RELATED WORK

Light field reconstruction

Light field reconstruction has been performed from different observation models, i.e., different instances of (1), such as coded aperture [15], [16], [17], compressed sensing [2], [18], novel view synthesis and angular super-resolution [3], [19], [20], [21], spatial angular super-resolution aided by high resolution central view [22] and also light-field image inpainting and focal stack reconstruction in [23]. Since virtually all such observation models make the solution of (1) an ill-posed problem, a natural strategy is to consider regularized energy minimization methods, for example [2], [21]. Alternately, one could estimate depth maps [24], [25].
or disparity maps which could be subsequently used to synthesize light fields, see [3, 25] for examples. Recently, learning based approaches have also been applied in light field recovery for coded aperture in [6, 8], compressed sensing in [5], view synthesis and angular super-resolution in [4, 7, 27, 28, 29, 30], spatial and angular super-resolution in [31, 32] as well as view extrapolation for wide baseline light fields in [33, 34].

While neural network-based reconstruction schemes [4, 3, 6, 7, 8, 30, 32, 35] outperform traditional approaches to light field reconstruction by a large margin, they are applicable to specific observation models only, i.e., they are not flexible in adapting to modifications of the observation model. We note that [36] is a deep network based approach for compressive light field recovery, which also takes a mask as an input to the deep network, achieving flexibility with respect to different masks for compressive sensing.

Learning light field representations has been addressed previously since the data is high dimensional and contains redundant information. Representations based on sparse coding have been utilized to perform inference tasks such as disparity estimation [37, 38] and light field reconstruction [2]. Alperovich et al. [39] have shown that an autoencoder trained on stacks of epipolar-plane images (EPI) can learn useful light field representations which can be used for supervised training for disparity estimation and intrinsic decomposition. Recently, there have been efforts to synthesize a light field from a single image in [40, 41, 42]. Srinivasan et al. [40] train an end-to-end network which is based on depth estimation from single image and subsequent warping to render light field. CNN-based appearance flow estimation is used in [41], to accomplish LF synthesis from a single image. Chen et al. [42] synthesize a light field from single image without estimating any depth map using deep neural network employing GAN loss. Generating a light field from a single view can have several possible solutions. The approaches [40, 41, 42] output a fixed light field for a given input image. In contrast, our CVAE can generate different LF patches for the same input patch, by sampling in the latent distribution.

Generative models

Deep generative models starting from variational autoencoders [43] and GANs [44] have emerged as an important tool for learning data representations in an unsupervised way. These models have demonstrated an impressive ability in generating realistic new image samples from specific image classes [45]. However, training generative models which can synthesize class independent natural images remains difficult and often requires huge network architectures like [46]. Recently, generative models have also been proposed for videos [47, 48]. However, generative modeling has not been used in context of light fields so far.

Image reconstruction using generative models

In addition to generating realistic samples of images [45, 49], generative models have also been used as priors in various image reconstruction [12, 13, 50] and image manipulation [51] tasks. Some of these algorithms involve an optimization in the latent space of the generative model with gradient descent based updates in [12, 13]. More sophisticated optimization schemes such as projected gradient descent, ADMM have also been used in conjunction with GAN priors for optimization in the latent space [52, 53, 54]. Alternatively, encoder-decoder based optimization has also been used with gradient based updates in [50] and with ADMM in [55]. Such methods have, however, not been exploited for light fields and have not been extended to multiple generative priors yet.

3 LIGHT FIELD MEASUREMENT MODELS

Continuous light fields are represented using the plenoptic function \( L(x, v) \) that represents the radiance of the scene emitted at the spatial position \( x \) and in the angular direction \( v \). For the discrete light field, we consider the angular resolution for each axis to be \( N_v \), and the spatial resolution of each view to be \( N_x \times N_y \). The discrete light field can be represented in vector form as \( I \in \mathbb{R}^k \) with \( k = N_v^2 \cdot N_x^2 \). In this work, we attempt to solve 3 different light field reconstruction problems utilizing generative priors: (i) LF view synthesis/ view upsampling (ii) Spatial-angular super-resolution aided by a central view, and (iii) LF recovery from coded aperture images. Among these 3 models, for LF view synthesis and spatial angular super-resolution, we assume that the central view is available. We now consider the specific measurement models for each of these reconstruction tasks.

View synthesis / Angular super-resolution

The task of view synthesis is to recover all sub-aperture images (SAIs) from a sparse subset of input views. The forward model can be considered to be a point-wise multiplication of the light field with a binary mask \( M \), whose value is 1 at the known views, and 0 at all other locations, leading to

\[
i(x, v) = L(x, v) \odot M(x, v).
\]

where \( \odot \) is the point-wise multiplication operator.

Spatial and angular super-resolution using central view

Here the task is to recover all SAIs from a sparse subset of spatially down-sampled input views. Furthermore, we assume that the central view is available in full resolution which aids in spatial upsampling of novel views. The corresponding measurement model can be written as

\[
i(x, v) = (L(x, v) \odot M(x, v))_{\downarrow_{2}(v)}.
\]

where \( M \) is a binary mask which is non-zero only at known views, and \( \downarrow_{2}(v) \) is the spatial down-sampling operation of the known views. However, the central view is available at full resolution, i.e \( \downarrow_{2}(v) \) is 1, for the central view.

Coded aperture

Coded aperture images are the result of optical multiplexing only along angular dimension. In a continuous setting, the coded aperture image formation model can be written as

\[
i(x) = \int L(x, v) M(v) dv
\]

where \( M \) represents the coded mask, which depends on the angles \( v \), but not on the spatial position.
Each of the forward models given in Eqs. (2), (3), (4), is a linear measurement model, which can be discretized and represented via (1). In the following, we develop generative models for light fields, which can be exploited for solving such general LF reconstruction problems.

4 Light Field Generative Models

Though light field data has high dimensionality, patches of light fields lie in a manifold of much lower dimension owing to their redundant structure [39]. Therefore, training generative models for light field patches instead of full light fields is a promising alternative. Moreover, the representation learned on the small light field patches can generalize to a wide variety of different light fields independent of any specific class of objects.

We introduce generative models for 4D light field patches based on a class of variational autoencoders known as Wasserstein autoencoders [14]. In addition to the autoencoder MSE loss between input and output, these models have a maximum mean discrepancy (MMD) penalty between the encoder distribution, and the prior latent distribution, instead of the Kullback-Leibler (KL) divergence penalty found in the traditional variational autoencoders. The loss function is given as

$$\text{Total loss} = \text{MSE loss} + \lambda \cdot \text{MMD loss}$$ (5)

We propose two generative models for light field patches (i) CVAE, a conditional VAE for light fields, conditioned on the central view, and (ii) an unconditional VAE for light fields. We trained both the models for LF patches of spatial resolution $25 \times 25$. The angular resolution of the LF patch is chosen to be same as the angular resolution of the light field to be reconstructed ($5 \times 5$ and $7 \times 7$ in our experiments).

4.1 Conditional Generative Model

Although we restrict the spatial extent of a light field patch to $25 \times 25$ pixels, due to diverse possibilities of texture content, parallax effects and occlusion effects, representing any patch with a generative model would still be a difficult task. Therefore, we first simplify the task and develop a model which is conditioned on the patch corresponding to the central view. With the central patch being fed into the network as an additional input, the encoder only needs to encode the additional information to represent the parallax and occlusion effects in the light field. The decoder learns to utilize the information from the central view to map the latent variable to the light field.

The schematic of the CVAE with its main components is illustrated in Fig. 2. Features of central view are extracted from a convolutional feature extractor at different layers (CVF1 and CVF2), which are together referred to here as the central view features (CVF). These are simultaneously fed to both encoder and generator. The feature extractor is jointly trained along with the encoder and generator. We employ 3D and 2D convolutions in our architecture as an alternative to computationally expensive 4D convolutions. To realize this, the encoder blocks Enc1 and Enc2 in $E_1$ (Fig. 2 (d)) take the input 4D LF patch as a set of 3D LF patches by splitting them along the horizontal and vertical view dimensions, respectively. The outputs of these encoder blocks are together fed into a common encoder Enc3, along with a set of central view features CVF1. This encoder’s output together with central view features CVF2 are further encoded by fully connected layers to output latent code $z$. The generator $G_1$, takes in the latent code and central view features CVF2 which first pass through linear fully connected layers to output horizontal and vertical view dimensions, respectively. The outputs of these encoder blocks are together fed into a common decoder Dec1, along with a set of central view features CVF1, which itself is input to a final 4D convolutional layer. Further details of CVAE network architecture for both the conditional models are provided in the appendix.

Fig. 2. (a) Schematic of CVAE. (b) Central view feature (CVF) extraction. (c) Architecture of feature extractor, CVF = \{CVF1, CVF2\}. (d) Schematic of encoder $E_1$ of CVAE. (e) Schematic of generator $G_1$ of CVAE.
4.2 Unconditional Generative Model

In many applications of light field recovery, the central view may not be available. Therefore, we also need to learn a representation of light field patches without assuming that the central view information is available. To handle the difficulty of data diversity, we design an unconditional VAE consisting of two stages. The schematic of the unconditional VAE with its main components is illustrated in Fig. 3. First, a VAE is trained to recover an input that has been spatially down-sampled by a factor of 2. The corresponding generator is $G_{2,0}$. The architecture of this first stage generator is illustrated in Fig. 3 (d). In the second stage, we use the trained $G_{2,0}$ as a part of VAE generator $G_2$ as shown in Fig. 3 (c). The output of the fully connected layers in the generator is divided into $z_0$ and $z_1$. $z_0$ is fed into the trained $G_{2,0}$, and $z_1$ into the residual components of the generator $G_2$. Finally, the system is trained such that the sum of their outputs recovers the input light field. Similar to the CVAE, the VAE architecture mainly employs 3D and 2D convolutions. The architecture in both the stages is very similar to the CVAE with partial encoders/decoders for rows and columns, and common encoders/decoders. The only difference is that we do not have additional CV features, instead we have two-stage training. Further details of the VAE network architecture are provided in the appendix.

4.3 Reconstruction from Generative Models

To motivate our use of generative models CVAE and VAE for LF recovery, we show here sample reconstructions (encoding and generation) from our trained generative models. This is to show the strengths of each model and also weaknesses which can be overcome by combining both the models together. Fig. 4 shows sample reconstructions from both our generative models for 4 light field patches. We handle colored light field inputs by reconstructing each color channel separately.

We observe that the CVAE can reconstruct the input LF patches quite accurately, see Fig. 4 (b), and is even able to realistically estimate pixel values that are not present in the central view due to the parallax. Despite capturing the gross structure well, the reconstructions in Fig. 4 (c) from the unconditional VAE look more blurry, showing the difficulty of the generator in accurately reproducing light field just from the latent code. This is in line with prior research results, i.e. autoencoder based generative models [14], [56] tend to produce blurry looking outputs.

To demonstrate the efficacy of the CVAE latent code in encapsulating different properties of a light field patch, Fig. 4 (d) shows the CVAE reconstruction we obtain when replacing the true central view by those of some other $25 \times 25$ texture. As we can see, the result is a new LF patch with disparity values similar to the input LF patch in Fig. 4 (a), indicating that latent vector indeed encodes an understanding of the geometry of the scene.

Effect of distorted inputs:

In Fig. 5 we illustrate the effect of distortions in the input on reconstructions by the CVAE and the VAE models. We consider different scenarios: i) When input is an undistorted LF patch Fig. 5 (a), ii) when some views input other than the central patch are distorted to varying degrees Fig. 5 (b) & (c), iii) when the central patch is also distorted Fig. 5 (d). The specific distortions are described in the caption of Fig. 5. In all the scenarios, we can see that the unconditional generative model can roughly reproduce the overall structure of the true underlying LF patch, see third row in Fig. 5. This is due to the property of the light field VAE to always output a patch with light field structure. Therefore, despite distortions, VAE maps the input to the closest LF patch in its representation. Though the disparity information in the gross structure is captured in these outputs, the finer details are missing and the outputs appear blurred, as the VAE has not enough constraints to reconstruct sharp output.
light fields. In contrast to the unconditional model, the conditional model can accurately recover the input including fine textural details when input is a clean LF patch, see second row in Fig. 5. However, it fails in reconstruction when central view is missing. In the case where the central view is undistorted, and the other views are distorted, the CVAE still outputs a LF patch with high similarity to the ground truth patch. In the following, we develop light field recovery techniques which exploit the strength of both generative models, the CVAE and the VAE.

5 Generic Light Field Recovery

Light field recovery from measurements as seen in Sec. 3 is an inherently ill-posed problem, and needs strong priors to obtain acceptable solutions. We consider two scenarios: i) The central view is available, and ii) the central view is not available. We now proceed to solve the LF reconstruction problems in both the cases using our generative models developed in Sec. 4.

Central view available

In some LF recovery applications such as view synthesis, or spatial angular super-resolution, one can assume that the central view is known. For such scenarios, we utilize our CVAE model for reconstruction. Given the central view, the generator of CVAE is trained to always map a latent code to a light field patch. Therefore, we optimize over the latent space similar to [12, 13]. However, unlike [12, 13], we use a conditioned generative model. More specifically, we solve

$$\min_{\hat{z}} \|i - \Phi G_1(\hat{z})\|_2^2$$

where $G_1$ is the generator of CVAE and $\Phi$ is the operator corresponding to measurement from angular subsampled views or from spatial and angular subsampled views, assuming central view is present. We minimize (6) locally using Adam [56], a gradient-based optimization algorithm. After finding a local minimum $\hat{z}$ of (6), $G_1(\hat{z})$ is considered to be our final light field estimate.

Central view not available

In LF recovery applications such as recovery from coded aperture, the central view is not available. In such a case, we cannot use our CVAE model alone for recovery. Instead, we constrain the solution space using both our generative models. The generators of both VAE and CVAE are trained to always output a light field patch. Consequently, the reconstruction error would be low only when the input to the encoder is similar to a light field patch.
As discussed in Sec. 4.3, the reconstructions from unconditional VAE model capture well the gross structure, but appear blurred. CVAE gives sharper reconstructions given the central view, which may not be available in general. Therefore, we use both the models jointly as this allows us to guide the reconstruction to realistic and sharp solutions. Rather than optimizing over the latent space of the generator, we directly optimize over the light field patch and utilize both CVAE and VAE as regularizers which penalize reconstruction error. The conditional model gets the central view from the current estimate of \( I \). The encoder \( E_1 \) and generator \( G_1 \) of the CVAE, as well as the encoder \( E_2 \) and generator \( G_2 \) of the VAE are employed as regularization terms in our energy minimization approach by solving

\[
\min_{\mathbf{I}} \| \mathbf{I} - \Phi \mathbf{I} \|_2^2 + \alpha_1 \| \mathbf{I} - G_1(E_1(\mathbf{I})) \|_2^2 + \alpha_2 \| \mathbf{I} - G_2(E_2(\mathbf{I})) \|_2^2
\]

(7)

where \( \Phi \) is the operator corresponding to coded aperture measurements. We solve this problem using Adam optimizer. We’d like to point out that the use of both generative models with complementary strength is crucial for the success of the approach. We observed experimentally that using only one of the two generative models does not lead to satisfactory performance when central view is not available.

In principle, we can use Eq. 7 even when central view is available, by setting \( \alpha_2 \) to 0. We observed that this gives similar reconstruction performance as optimizing in the latent space of the conditional generator Eq. 6. However, we use Eq. 6 for reconstruction when central view is available, as it involves only generator in optimization and hence faster.

6 Experiments

To be able to compare with recent network based approaches on small baseline light fields, we evaluate view synthesis from sparsely sampled views for LFs with angular resolution \( 7 \times 7 \). We evaluate LF recovery for view synthesis, spatial-angular super-resolution and coded aperture for LFs with angular resolution \( 5 \times 5 \).

6.1 Experimental Setup

**Baselines:**

We obtain the performance references for the reconstruction tasks using both model based approaches and network based approaches for comparisons. For \( 7 \times 7 \) view synthesis,
we compare with the recent neural network based technique of [28]. For comparison with a traditional approach, we report the performance of the depth-based approach from [28].

The dictionary based approach of Marwah et al. [2] is a flexible technique, which can be used with any observation model. We use their open sourced code which is available for LFs of angular resolution $5 \times 5$. We use this as a reference for model-based approaches on all the $5 \times 5$ LF recovery tasks. For the best performance of [2], we always compute their result obtained by averaging over overlapping patches with stride 1. Additionally, for coded aperture, we also compare with neural network approach of [6].

Datasets:

For training the generative models, we used the following datasets: i) The training set used by Kalantari et al. [4]. ii) the training set used in CNN based depth estimation for light fields by Heber et al. [57], and iii) the training set used in encoder-decoder based light field intrinsics [39]. These datasets contain a significant number of samples with effects such as occlusions and specular reflections. We create a training set by randomly cropping $250K$ LF patches of resolution $5 \times 5 \times 25 \times 25$ in gray scale from these datasets and use them for training the CVAE and VAE with angular resolution $5 \times 5$. Similarly, a training set of $250K$ LF patches of resolution $7 \times 7 \times 25 \times 25$ was created to train the CVAE with angular resolution $7 \times 7$. The dataset from [39] has high disparity, therefore we down-scale those light fields spatially by a factor of 1.4 before extracting patches from this data.

We evaluate the light field recovery on synthetic and real datasets. Specifically, for LFs of angular resolution $5 \times 5$, we evaluate the recovery from all the tasks on the light fields "Dino", "Kitchen", "Medieval 2" and "Tower" from the synthetic New HCI dataset [58]. Furthermore, we evaluate coded aperture reconstruction on the real light field from [6]. We evaluate view synthesis for LFs of angular resolution $7 \times 7$ on the test set of [4] which contains 30 real light fields captured by Lytro Illum.

Generative model training:

We used Pytorch 1.1.0 for all our experiments. For training CVAE and VAE, we use mini-batches of size 128 and trained both the models for 150 epochs. We used Adam optimizer [56], with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. We set the initial learning rate to $10^{-3}$, which is decreased by a factor of 2 after 30 epochs, further by a factor of 5 after first 50 epochs and finally by a factor of 10 after 100 epochs. For all our models, we choose the factor $\lambda$ in eq. (5) to be 100.

LF recovery:

Since our generative models are trained on gray scale patches, we divide the input into patches of suitable dimensions and use our generative models on all color channels separately. We solve the LF reconstruction tasks using Adam optimizer as discussed in Sec. 5. For view synthesis and spatial-angular super-resolution, we also perform reconstruction using overlapping patches with stride 5.

6.2 Results

We now evaluate the efficacy of our approach on different LF recovery tasks. We perform quantitative evaluation in terms of PSNR and also qualitative evaluation by comparing light field views of our approach with ground truth and baseline methods and show the corresponding error maps.
6.2.1 Central View Available

View synthesis $7 \times 7$:

We compare our approach with recent CNN based technique of Wu et al. [28] for LF reconstruction from sparsely sampled input views. We consider upsampling the angular resolution from $3 \times 3$ to $7 \times 7$. Since central view is available for this task, our approach uses CVAE for reconstruction. We use the publicly available trained model of [28] for evaluating their approach. We also report the performance of a traditional depth estimation-based approach from [28] for this task, where the depth is estimated using the approach of Jeon et al. [24], followed by a novel view synthesis by warping the input views following [25]. To demonstrate the flexibility of our approach, we also show $7 \times 7$ LF reconstruction from 5 input views including the central view. The mask used for selecting the 5 input views is provided in the inset of Fig. 6 (a). Since view extrapolations cannot be handled by Wu et al. [28], we show visual comparison only with the ground truth for this task.

Results of our quantitative evaluation on 30 real LFs of Kalantari [4] test set are provided in Tab. 1. ‘Ours’ indicates our reconstruction using overlapping patches with stride 5. Following Wu et al. [28], we show the result of average PSNR of the luminance component of novel synthesized views. Due to limited space we report only average PSNR of all the 30 LFs. Quantitative comparison for individual LFs are provided in the appendix. For the task of view upsampling from $3 \times 3$ to $7 \times 7$, we compute the average PSNR of the 40 novel views. For this task, we find that our performance is approaching the CNN based method of [28], with PSNR reduction of only $1.4$ dB when we use overlapping patches, and $2.6$ dB when non-overlapping patches are used. Our approach also outperforms the depth based approach using the method of Jeon et al. [24] by a large margin. Even when the number of known views is reduced to 5, our average PSNR of 44 novel views is 39.57 dB, with a reduction of only $0.2$ dB demonstrating the strength of our approach.

A qualitative comparison of the synthesized views for the task of $7 \times 7$ view synthesis is provided in Fig. 6 for the LF ‘Cars’ from the 30 scenes test set. Novel synthesized views at angular location (6, 6) (depicted by gray location in the inset) are shown. The first row of Fig. 6 (a)–(c) gives a visual comparison of the results of our approach with the ground truth when 5 input views are used. Visually, it can be seen that our approach provides a reasonable reconstruction quality even when using limited number of input views. The second row of Fig. 6 (d)–(f) compares our method with the approach of Wu et al. [28], for the task of $3 \times 3 \to 7 \times 7$ angular super resolution. In terms of reconstruction quality, our approach performs slightly worse than [28]. However, this is to be expected as [28] uses network specifically trained for this task. In contrast, we obtain a comparable reconstruction quality with flexible input views. It can be noticed from the error maps and zoomed in patches that our approach preserves the details fairly well. Further, we can observe that there are errors at the patch boundaries when non-overlapping are used. These errors are reduced due to averaging effect when overlapping patches are used. Visual

---

**Table 1**

| Corrupt. | [28] | Ours | Ours* |
|---|---|---|---|
| None | 36.02 | 31.74 | 33.45 |
| Gaussian noise $\sigma = 0.05$ | 33.34 | 31.75 | 33.47 |
| Gaussian noise $\sigma = 0.1$ | 29.95 | 31.67 | 33.41 |
| Salt&Pepper noise | 25.02 | 31.66 | 33.35 |
| 50% Pixel drop | 13.60 | 31.68 | 33.39 |

$3 \times 3 \to 7 \times 7$ view synthesis result on the LF ‘Cars’, under varying corruptions: PSNR values in dB

**Table 2**

| Corrupt. | [28] | Ours | Ours* |
|---|---|---|---|
| None | 36.02 | 31.74 | 33.45 |
| Gaussian noise $\sigma = 0.05$ | 33.34 | 31.75 | 33.47 |
| Gaussian noise $\sigma = 0.1$ | 29.95 | 31.67 | 33.41 |
| Salt&Pepper noise | 25.02 | 31.66 | 33.35 |
| 50% Pixel drop | 13.60 | 31.68 | 33.39 |

$3 \times 3 \to 7 \times 7$ view synthesis on 30 real scenes dataset of [4]. * indicates PSNR values for [24] as reported from the paper [28].

---

Gaussian noise $\sigma = 0.05$, Ours

Salt&Pepper, [28] Salt&Pepper, Ours

50% pixels, [28], [28] Ours

50% pixels, Ours

50% pixels, Ours*
comparisons for more LFs and videos of the reconstructed LFs are provided in the appendix.

To further demonstrate our flexibility vis-a-vis end to end trained networks, we consider the task of $3 \times 3 \rightarrow 7 \times 7$ angular super resolution and compare our reconstruction with Wu et al. [28], when inputs are corrupted. We assume that central view is clean and consider that the remaining 8 views are corrupted by different distortions. The qualitative and quantitative comparison of our reconstructions with the approach of Wu et al. [28], with corrupted input views is provided in Fig. 7 and in Tab. 2 for the LF ‘Cars’. Novel view at angular location (6, 6) is depicted. With additive Gaussian noise of standard deviation $\sigma = 0.05$ in 8 input views, the PSNR of novel views using [28] drops from 36.02 dB to 33.34 dB. When we increase the noise level to $\sigma = 0.1$ this value further drops to 29.95 dB. This degradation in the quality of reconstruction is also evident from the error maps in Fig. 2. In contrast, our reconstruction quality is robust to addition of noise. We also investigate the effect of corruption by salt and pepper noise with a probability of 0.05 on the reconstruction quality. Even in this case, the performance of [28] is severely affected, with PSNR reduction of 11 dB compared to the clean case, where as our performance only shows a marginal decrease of 0.1 dB. We note that we employ an $L_1$ loss, as it is more suited to handle salt and pepper noise when compared to the traditional MSE loss in Eq. (6). This demonstrates the flexibility of our energy minimization based approach in adapting to different noise statistics. When we used MLE loss, our PSNR dropped by about 2 dB compared to the clean case. Finally, when 50% pixels are randomly dropped from the 8 known views, the neural network based approach of [28], completely fails in reconstruction. In contrast, we can incorporate an additional mask corresponding to the pixel drop in our optimization, and consequently our reconstructions remain robust to this distortion.

**View synthesis 5 × 5:**

We compare our approach for view synthesis with [2] for two different selection of input views using masks $M_1$ and $M_2$. A qualitative comparison of the synthesized views is provided for the LF ‘Dino’ for mask $M_1$ and $M_2$ in Fig. 6. The locations of the known views are depicted in white in the inset of Fig. 6 where as gray depicts the location of the synthesized novel view shown. Extrapolating novel views away from known views is difficult. Even for this challenging case, we observe the quality of our reconstruction both with overlapping patches denoted by ‘OursOL’ and without overlapping patches is better and sharper compared to the reconstruction from the dictionary based approach of [2]. This is also evident from the error maps shown in Fig. 8. We can observe that averaging effect of overlapping patches mitigates the errors at the patch boundaries when non-overlapping are used.

The results of our quantitative evaluation on synthetic HCI data are summarized in Table. 3 where the average PSNR of the light field views is presented. Our approach without considering overlapping patches is superior by 2.63 dB and 3.13 dB to the dictionary based approach of [2] for masks $M_1$ and $M_2$, respectively. Our performance further improves when we consider overlapping patches with stride 5, where our approach is better by 4 dB and 4.4 dB, respectively. We note that for evaluating [2], we always considered overlapping patches with stride 1.

**Spatial and angular super-resolution 5 × 5:**

Fig. 9 provides visual comparison of our LF reconstruction with the approach of [2] for the task of spatial-angular super-resolution on the LF ‘Kitchen’. The masks used for the measurements is provided in the inset of ground truth view of the LF ‘Kitchen’ in Fig. 6. The central view is available in full resolution and is depicted in white. Views in red are spatially down-sampled by a factor of 3. It can be
observed that our reconstruction of the novel view (depicted in gray in the inset) with both overlapping patches and non-overlapping patches is of superior quality compared to the reconstruction from the approach of [2]. This is further substantiated by the error maps shown in the Fig. 9 which depict a much lower error in our reconstruction compared to the approach from [2]. Further reduction in error at patch boundaries is also observed when overlapping patches are used.

Table 4 provides a quantitative comparison of our method with the dictionary based approach of [2]. Our approach, even with non-overlapping patches provides better reconstruction with improvement in average PSNR of 2.35 dB and 2.9 dB over the approach of Marwah et al. [2]. Further improvement is achieved when we consider overlapping patches with stride 5, with a gain in average PSNR of 4.3 dB over the approach of [2].

6.2.2 Central View Unavailable

Coded aperture 5 × 5:

We evaluate the LF recovery from 2 coded aperture observations, using two different coded mask sets ‘Normal’ and ‘Rotated’ available from [6] for our approach, [3] and [2]. For brevity, we denote the mask sets ‘Normal’ and ‘Rotated’ by $M_1$ and $M_2$ respectively. The results of quantitative evaluation on synthetic data is summarized in Table 5. To evaluate the approach of [6], we use the publicly available trained reconstruction network corresponding to $M_1$. For $M_2$, we reproduce the values reported in [6], since a trained network is not available. The average PSNR on the test set for our method is about 1.4 dB and 1.34 dB lower than the end-to-end trained model of [6] for $M_1$ and $M_2$ respectively. This is to be expected as the training was tailored to the specific masks. However, when compared to the model-based approach of [2], the average PSNR of our method is superior by 2.93 dB and 2.3 dB for $M_1$ and $M_2$.

For qualitative evaluation, we show sample LF reconstructions using coded masks $M_1$ on the LFs ‘Dino’ and ‘Medieval’ in Fig. 10. We can observe that our approach provides a reasonably good recovery, with performance approaching an end-to-end trained network. Our recovery is also more accurate when compared to [2]. Error maps in Fig. 10 also depict a relatively lower error from our approach when compared to [2].

To demonstrate the vulnerability of the end-to-end trained reconstruction pipeline, we altered the coded aperture mask from the set of $M_1$ and then perform LF reconstruction using the method of [6]. Minor changes were applied to only one of the two masks in the set $M_1$. First, we swap the values of the mask at locations with coordinates $(0,0)$ and $(0,2)$. With this only change, the performance of [6] dropped from 38.7 dB to 24.3 dB on the ‘Dino’ LF. When we swap the values at three sets of location, the method of [6] completely failed to reconstruct a meaningful light field (yielding a PSNR of 12.2 dB). In contrast, the effect of changes in the mask on our approach is marginal, since our optimization scheme explicitly takes the mask as an input. With the first swap in the mask, our PSNR changed to 36.86 dB, compared to 37.4 dB of the original mask. With three swaps, the PSNR value for our reconstruction is 37.07 dB, demonstrating our flexibility. Views from the
reconstructed light fields are shown in Fig. 11. We apply our reconstruction method on the real observations obtained in the work of [6]. In their setup, the black-aperture image was not completely dark. Consequently, the image obtained from the black aperture was subtracted from the observations. In Fig. 12, we show a specific view obtained from our reconstruction along with the corresponding result obtained by the authors of [6]. Close-ups near the occlusion boundaries for two different views (with appropriate vertical alignment) in Fig. 8 (c) and (d) show a comparable quality of our approach (left columns) to the results obtained by [6] (right columns).

7 CONCLUSION

We developed the first autoencoder based generative models for 4D light field patches for generic reconstruction: an unconditional model and a model conditioned on the central view. We developed algorithms for generic light field reconstruction by exploiting the strengths of our generative models. We evaluated our approach 3 different LF reconstruction tasks. Experimental results indicate that our flexible approach leads to good quality of LF estimates with performance superior to other optimization-based approaches, while being only slightly worse than that of end-to-end trained network. Further, We believe that such experimental results are very promising and could serve as a starting point for further research on more powerful light field generative models.

APPENDIX A

NETWORK ARCHITECTURES

We use the following notation to describe convolutional mappings. $C_{a \rightarrow b}^{F} \downarrow_{S}$ represents convolution filter mapping from channel dimension of $a$ to $b$ with filter size of $F$ and stride $S$. $C_{a \rightarrow b}^{F} \uparrow_{S}$ represents fractional strided convolution (transpose convolution) filter mapping from channel dimension of $a$ to $b$ with filter size of $F$ and stride $S$.

A.0.1 Conditional VAE

The architectural details of the components of CVAE in Fig. 2 are as follows:

Feature extractor:

$$C_{1 \rightarrow 6}^{(3,3)} \downarrow_{(1,1)} \rightarrow C_{6 \rightarrow 10}^{(3,3)} \downarrow_{(2,2)} \rightarrow C_{10 \rightarrow 20}^{(3,3)} \downarrow_{(1,1)} \rightarrow C_{20 \rightarrow 40}^{(3,3)} \downarrow_{(1,1)} \rightarrow C_{40 \rightarrow 50}^{(3,3)} \downarrow_{(2,2)} \rightarrow C_{50 \rightarrow 60}^{(3,3)} \downarrow_{(1,1)}$$

Partial row/column encoders Enc1, Enc2:

$$C_{5 \rightarrow 20}^{(3,3)} \downarrow_{(1,1,1)} \rightarrow C_{20 \rightarrow 40}^{(3,3)} \downarrow_{(1,2,2)} \rightarrow C_{40 \rightarrow 60}^{(3,3)} \downarrow_{(1,1,1)}$$

Partial common encoder Enc3:

$$C_{140 \rightarrow 200}^{(3,3)} \downarrow_{(1,1)} \rightarrow C_{200 \rightarrow 250}^{(3,3)} \downarrow_{(2,2)} \rightarrow C_{250 \rightarrow 300}^{(3,3)} \downarrow_{(1,1)}$$

Partial common decoder of Dec1:

$$C_{300 \rightarrow 250}^{(3,3)} \uparrow_{(1,1)} \rightarrow C_{250 \rightarrow 200}^{(3,3)} \uparrow_{(2,2)} \rightarrow C_{200 \rightarrow 120}^{(3,3)} \uparrow_{(1,1)}$$
Fig. 12. Real result using the observation of [6]. (a) A view from our reconstructed light field. (b) Corresponding view from the result of [6]. (c) and (d) left half shows patches from two different views of our reconstruction and right half similarly shows patches from the result of [6].

row/column decoder Dec2, Dec3 of $G_{2, 0}$:

$$C^{(3, 3, 3)}_{140 \rightarrow 80} \uparrow(1, 1, 1) \rightarrow C^{(3, 3, 3)}_{80 \rightarrow 40} \uparrow(1, 2, 2) \rightarrow C^{(3, 3, 3)}_{40 \rightarrow 20} \uparrow(1, 1, 1)$$

All the convolutional layers except the last layer of generator are followed by batch norm and ReLU non-linearity. We fix the latent dimension of CVAE to be $160$. We used isotropic Gaussian prior, with variance of 2 for the latent space. The architecture is same for both the angular resolutions $5 \times 5$ and $7 \times 7$, except for padding in the first convolutional layer.

A.0.2 Unconditional VAE

We adopt a two-stage training strategy for training the unconditional VAE, as discussed in Sec. 2. In the first stage, we train a VAE to recover the input image, which is down-sampled by a factor of 2. For this stage, we use Gaussian prior with variance 2, for the latent code of dimension 90. The schematic of first stage encoder is similar to that of encoder $E_2$ of final VAE in Fig. 3. The details of the components in the first stage training are as follows:

Partial row/column encoders $Enc1, Enc2$:

$$C^{(3, 3, 3)}_{5 \rightarrow 20} \downarrow(1, 1, 1) \rightarrow C^{(3, 3, 3)}_{20 \rightarrow 40} \downarrow(1, 2, 2) \rightarrow C^{(3, 3, 3)}_{40 \rightarrow 60} \downarrow(1, 1, 1)$$

Partial common encoder $Enc3$:

$$C^{(3, 3)}_{140 \rightarrow 200} \uparrow(1, 1, 1) \rightarrow C^{(3, 3)}_{200 \rightarrow 250} \uparrow(2, 2) \rightarrow C^{(3, 3)}_{250 \rightarrow 300} \uparrow(1, 1)$$

Partial common decoder of $G_{2, 0}$ $Dec1$:

$$C^{(3, 3)}_{300 \rightarrow 200} \downarrow(1, 1, 1) \rightarrow C^{(3, 3)}_{200 \rightarrow 200} \downarrow(2, 2) \rightarrow C^{(3, 3)}_{200 \rightarrow 120} \downarrow(1, 1)$$

Partial row/column decoder $Dec2, Dec3$ of $G_{2, 0}$:

$$C^{(3, 3, 3)}_{140 \rightarrow 80} \uparrow(1, 1, 1) \rightarrow C^{(3, 3, 3)}_{80 \rightarrow 40} \uparrow(1, 2, 2) \rightarrow C^{(3, 3, 3)}_{40 \rightarrow 20} \uparrow(1, 1, 1)$$

For the second stage training, we use isotropic Gaussian prior for the latent code, with variance of 2 and latent dimension 220. The architecture of components is similar to that of stage-1 except for the common partial encoders and decoders, which have different number of channels. The architecture of these components is specified below

Partial common encoder $Enc3$ of $G_{2}$:

$$C^{(3, 3)}_{120 \rightarrow 220} \uparrow(1, 1, 1) \rightarrow C^{(3, 3)}_{220 \rightarrow 250} \uparrow(2, 2) \rightarrow C^{(3, 3)}_{250 \rightarrow 360} \uparrow(1, 1)$$

Partial common decoder $Dec1$ of $G_{2}$:

$$C^{(3, 3)}_{320 \rightarrow 250} \downarrow(1, 1, 1) \rightarrow C^{(3, 3)}_{250 \rightarrow 220} \downarrow(2, 2) \rightarrow C^{(3, 3)}_{220 \rightarrow 120} \downarrow(1, 1)$$

All the convolutional layers except the last layer of generator are followed by batch norm and ReLU non-linearity.

| LF         | $3 \times 3 \rightarrow 7 \times 7$ | $5$ views $\rightarrow 7 \times 7$ |
|------------|-----------------------------------|-----------------------------------|
| Seashorse  | 38.11 34.20 35.63                  | 33.96 35.40                      |
| Rock       | 38.24 32.86 34.93                  | 32.55 34.71                      |
| Flower1    | 37.73 33.37 34.96                  | 33.14 34.77                      |
| Flower2    | 37.47 33.04 34.81                  | 32.78 34.60                      |
| Cars       | 36.02 31.74 33.45                  | 31.55 33.30                      |
| 1085       | 43.03 41.72 42.31                  | 41.27 41.85                      |
| 1086       | 43.75 42.80 43.70                  | 42.40 43.27                      |
| 1184       | 43.75 42.33 43.65                  | 43.10 43.53                      |
| 1187       | 43.20 42.11 42.80                  | 42.00 42.72                      |
| 1306       | 42.74 39.47 40.86                  | 39.29 40.69                      |
| 1312       | 45.66 44.33 45.55                  | 44.14 45.39                      |
| 1316       | 42.78 40.23 41.11                  | 40.09 41.00                      |
| 1320       | 41.67 39.39 40.20                  | 39.24 40.07                      |
| 1324       | 46.07 44.62 45.72                  | 44.43 45.55                      |
| 1325       | 46.06 43.99 48.04                  | 47.24 47.94                      |
| 1327       | 44.16 43.00 43.92                  | 42.85 43.78                      |
| 1328       | 40.76 37.18 38.32                  | 37.03 38.22                      |
| 1340       | 44.19 41.35 42.32                  | 41.05 42.55                      |
| 1340       | 45.38 46.12 47.01                  | 45.99 46.92                      |
| 1389       | 45.63 44.76 46.35                  | 44.60 46.23                      |
| 1390       | 45.95 46.29 47.06                  | 46.17 46.94                      |
| 1411       | 36.13 32.84 33.84                  | 32.68 33.69                      |
| 1419       | 39.30 36.08 36.95                  | 35.82 36.70                      |
| 1528       | 36.28 30.91 32.68                  | 30.50 32.36                      |
| 1541       | 36.84 31.77 33.76                  | 31.39 33.49                      |
| 1554       | 33.54 28.78 30.21                  | 28.46 29.93                      |
| 1555       | 35.89 31.28 32.88                  | 31.00 32.65                      |
| 1586       | 42.44 38.98 40.88                  | 38.75 40.74                      |
| 1743       | 42.12 40.52 41.77                  | 39.94 41.25                      |

Average 41.16 38.33 39.77 38.29 39.57

TABLE 6

PSNR values in dB for $7 \times 7$ view synthesis on 30 real scenes dataset [4].

APPENDIX B

ADDITIONAL RESULTS

B.0.1 7 × 7 View Synthesis

Results of our quantitative evaluation on 30 scenes real LF set are provided in Tab. [6]. ‘Ours’ indicates our reconstruction using overlapping patches with stride 5. For each LF, we report the result of average PSNR of the luminance component of novel synthesized views. We show more qualitative results of $7 \times 7$ LF reconstruction from 5 input views, and $3 \times 3$ input views in Fig. 8. The selected 5 input views is depicted in white and the novel view displayed is depicted in gray in the inset of the ground truth views. Shown here are ground truth and reconstructed views for the LFs ‘Seahorse’, ‘Flower2’, and ‘1340’ from
References

[1] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” Stanford Tech. Report CTSR, vol. 2005, no. 2, pp. 1–11, 2005.

[2] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, “Compressive light field photography using complete dictionaries and optimized projections,” ACM Transactions on Graphics (TOG), vol. 32, no. 4, p. 46, 2013.

[3] 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, 2013.

[4] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, “Learning-based view synthesis for light field cameras,” ACM Transactions on Graphics (TOG), vol. 35, no. 6, p. 193, 2016.

[5] M. Gupta, A. Jauhari, K. Kulkarni, S. Jayasuriya, A. Molnar, and J. Rick Chang, C.-L. Li, B. Poczos, B. Vijaya Kumar, and A. C. A. K. Vadathya, S. Girish, and K. Mitra, “A unified learning based approach of Wu et al. 30 real scenes set using our approach and network based JOURNAL OF L...”

[6] S. D. Babacan, R. Ansorge, M. Luessi, P. R. Mataran, R. Milan, and A. K. Katsaggelos, “Compressive light field sensing,” in Proceedings of the European Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 11–20.

[7] Y. Inagaki, Y. Kobayashi, K. Takahashi, T. Fujii, and H. Nagahara, “Learning to capture light fields through a coded aperture camera,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 418–434.

[8] H. Wing Fung Yeung, J. Hou, J. Chen, Y. Ying Chung, and X. Chen, “Fast light field reconstruction with deep coarse-to-fine modeling of spatial-angular clues,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 137–152.

[9] A. K. Vadathya, S. Girish, and K. Mitra, “A unified learning based framework for light field reconstruction from coded projections,” IEEE Transactions on Computational Imaging, 2019.

[10] T. Meinhardt, M. Moller, C. Hazirbas, and D. Cremers, “Learning proximal operators: Using denoising networks for regularizing inverse imaging problems,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1781–1790.

[11] R. Ramamoorthi, R. Ng, and A. Kar, “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines,” ACM Transactions on Graphics (TOG), 2019.

[12] N. Meng, H. K. So, X. Sun, and E. Lam, “High-dimensional dense residual convolutional neural network for light field reconstruction,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1, 2019.

[13] P. P. Srinivasan, R. Tucker, J. T. Barron, R. Ramamoorthi, R. Ng, and N. Snavely, “Pushing the boundaries of view extrapolation with multigrid images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 175–184.

[14] B. Mildenhal, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines,” ACM Transactions on Graphics (TOG), 2019.

[15] S. Vagharshakyan, R. Bregovic, and A. Gotchev, “Light field reconstruction using shearlet transform,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 1, pp. 133–147, 2018.

[16] Y. Wang, Y. Liu, W. Heidrich, and Q. Dai, “The light field attachment: Turning a dslr into a light field camera using a low budget camera ring,” IEEE transactions on visualization and computer graphics, vol. 23, no. 10, pp. 2357–2364, 2016.

[17] C. J. Blocher and J. A. Fessler, “Blind unitary transform learning for inverse problems in light-field imaging,” in Proceedings of the IEEE International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[18] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai, and J. So Kweon, “Accurate depth map estimation from a lenslet light field camera,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1547–1555.

[19] M. S. M. Sajjadi, R. Köhler, B. Schölkopf, and M. Hirsch, “Depth estimation through a generative model of light field synthesis,” in Pattern Recognition - 38th German Conference (GCPR), ser. Lecture Notes in Computer Science, vol. 9796. Springer International Publishing, Sep. 2016, pp. 426–438. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-45886-1_35.

[20] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis, “Depth synthesis and landmark regression for plausible image-based navigation,” ACM Trans. Graph., vol. 32, no. 3, Jul. 2013. [Online]. Available: https://doi.org/10.1145/2487228.2487238.

[21] G. Wu, Y. Liu, Q. Dai, and T. Chai, “Learning sheared epi structure for light field reconstruction,” IEEE Transactions on Image Processing, vol. 28, no. 7, pp. 3261–3273, July 2019.

[22] G. Wu, Y. Liu, L. Fang, Q. Dai, and T. Chai, “Light field reconstruction using convolutional network on epi and extended applications,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 7, pp. 1681–1694, 2019.

[23] Y. Wang, F. Liu, Z. Wang, G. Hou, Z. Sun, and T. Tan, “End-to-end view synthesis for light field imaging with pseudo 4dcnn,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 333–348.

[24] J. Navarro and N. Sabater, “Learning occlusion-aware view synthesis for light fields,” ArXiv, vol. abs/1905.11271, 2019.

[25] M. S. K. Gul and B. K. Gunturk, “Spatial and angular resolution enhancement of light fields using convolutional neural networks,” IEEE Transactions on Image Processing, vol. 27, no. 5, pp. 2146–2159, 2018.

[26] N. Meng, H. K. So, X. Sun, and E. Lam, “High-dimensional dense residual convolutional neural network for light field reconstruction,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1, 2019.

[27] P. P. Srinivasan, R. Tucker, J. T. Barron, R. Ramamoorthi, R. Ng, and N. Snavely, “Pushing the boundaries of view extrapolation with multigrid images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 175–184.

[28] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines,” ACM Transactions on Graphics (TOG), 2019.

[29] S. Vagharshakyan, R. Bregovic, and A. Gotchev, “Light field reconstruction using shearlet transform,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3262–3270.

[30] O. Nabati, D. Mendlovic, and R. Giryse, “Fast and accurate reconstruction of compressed color light field,” in ICCP, May 2018, pp. 1–11.

[31] S. Heber and T. Pock, “Shape from light field meets robust pca,” in European Conference on Computer Vision. Springer, 2014, pp. 751–767.

[32] O. Johannsen, A. Sulc, and B. Goldluecke, “What sparse light field coding reveals about scene structure,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3262–3270.

[33] A. Alperovich, O. Johannsen, M. Strecke, and B. Goldluecke, “Light field intrinsics with a deep encoder-decoder network,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9145–9154.

[34] P. P. Srinivasan, T. Wang, A. Sreelal, R. Ramamoorthi, and R. Ng, “Learning to synthesize a 4d rgbd light field from a single image,”...
in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2243–2251.

[41] A. Ivan, I. K. Park et al., “Synthesizing a 4d spatio-angular consistent light field from a single image,” arXiv preprint arXiv:1903.12364, 2019.

[42] B. Chen, L. Ruan, and M.-L. Lam, “Lfgan: 4d light field synthesis from a single rgb image,” ACM Trans. Multimedia Comput. Commun. Appl., vol. 16, no. 1, Feb. 2020. [Online]. Available: https://doi.org/10.1145/3366371

[43] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” arXiv preprint arXiv:1312.6114, 2013.

[44] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in neural information processing systems, 2014, pp. 2672–2680.

[45] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” arXiv preprint arXiv:1812.04948, 2018.

[46] A. Brock, J. Donahue, and K. Simonyan, “Large scale gan training for high fidelity natural image synthesis,” arXiv preprint arXiv:1809.11086, 2018.

[47] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, “Mocogan: Decomposing motion and content for video generation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1526–1535.

[48] Anonymous, “Adversarial video generation on complex datasets,” in Submitted to International Conference on Learning Representations, 2020, under review. [Online]. Available: https://openreview.net/forum?id=Byx91R4twB

[49] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” arXiv preprint arXiv:1710.10196, 2017.

[50] P. Chandramouli and K. Vaishnavi Gandikota, “Blind single image reflection suppression for face images using deep generative priors,” in The IEEE International Conference on Computer Vision (ICCV) Workshops, Oct 2019.

[51] D. Bau, H. Strobelt, W. Peebles, J. Wulff, B. Zhou, J.-Y. Zhu, and A. Torralba, “Semantic photo manipulation with a generative image prior,” ACM Trans. Graph., vol. 38, no. 4, pp. 59:1–59:11, Jul. 2019. [Online]. Available: http://doi.acm.org/10.1145/3306346.3323023

[52] V. Shah and C. Hegde, “Solving linear inverse problems using gan priors: An algorithm with provable guarantees,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 4609–4613.

[53] C. Hegde, “Algorithmic aspects of inverse problems using generative models,” in 2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton). IEEE, 2018, pp. 166–172.

[54] F. Latorre, A. eftekhari, and V. Cevher, “Fast and provable admm for learning with generative priors,” in Advances in Neural Information Processing Systems 32. Curran Associates, Inc., 2019, pp. 12 004–12 016. [Online]. Available: http://papers.nips.cc/paper/9371-fast-and-provable-admm-for-learning-with-generative-priors.pdf

[55] S. Xu, S. Zeng, and J. Romberg, “Fast compressive sensing recovery using generative models with structured latent variables,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 2967–2971.

[56] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[57] S. Heber and T. Pock, “Convolutional networks for shape from light field,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3746–3754.

[58] “Heidelberg collaboratory for image processing: 4d light field dataset,” http://hci-lightfield.iwr.uni-heidelberg.de/ 2018.
Fig. 13. Result of $7 \times 7$ view synthesis. Error maps have error magnified by a factor of 10. Results best viewed when zoomed in.