An Epipolar Volume Autoencoder with Adversarial Loss for Deep Light Field Super-Resolution

Minchen Zhu, Anna Alperovich, Ole Johannsen, Antonin Sulc and Bastian Goldluecke
University of Konstanz
Konstanz, Germany
firstname.lastname@uni-konstanz.de

Abstract

When capturing a light field of a scene, one typically faces a trade-off between more spatial or more angular resolution. Fortunately, light fields are also a rich source of information for solving the problem of super-resolution. Contrary to single image approaches, where high-frequency content has to be hallucinated to be the most likely source of the downscaled version, sub-aperture views from the light field can help with an actual reconstruction of those details that have been removed by downsampling. In this paper, we propose a three-dimensional generative adversarial autoencoder network to recover the high-resolution light field from a low-resolution light field with a sparse set of viewpoints. We require only three views along both horizontal and vertical axis to increase angular resolution by a factor of three while at the same time increasing spatial resolution by a factor of either two or four in each direction, respectively.

1. Introduction

The problem of super-resolution, where one wants to recover a high-resolution image from one or more low-resolution versions, is one of the state-of-the-art challenges in computer vision. It is a highly ill-posed problem, and the classical approach which solves an inverse problem requires carefully constructed image priors [5]. It is well known that having multiple low-resolution images with slightly different viewpoints is both crucial as well as successful in recovering sharp details in the reconstructed image [30, 8, 25, 9].

Such a dense collection of viewpoints is described by the light field of a scene, a four-dimensional structure which parametrizes a set of captured rays by their point of origin in the focal plane (representing the viewpoint), and their intersection coordinate with an image plane. Dedicated light field cameras efficiently acquire dozens of views by multiplexing the rays onto a single sensor, or using multiple standard industrial cameras in an array. Compared to stereo imaging, light fields are more redundant and pro-

vide richer information about the scene, furthermore, the baseline is typically much smaller. However, there is a conflict between either having more views (angular resolution) or more spatial resolution within the views. Light field super-resolution therefore aims at increasing both spatial and angular resolution, where the latter amounts to generating novel views of the scene.

Optimization-based approaches [30, 24, 22] can provide results of a good quality, however, their parameters usually need to be fine-tuned and they often require a lot of time to converge. Recently, however, inverse problems have been successfully attacked by just brute-force learning a prediction of the desired result given the observed inputs. This is due to the rapid development of deep learning techniques in the last decade, which is made possible by dramatic increase in computational power of GPUs and the availability of large amounts of training data. Architectures based on deep encoder-decoder convolutional neural networks (CNN) turned out to be very powerful for super-resolution even of single images [28, 19], where they zoom a single low-resolution image to impressive high-resolution quality. However, fine detail is necessarily hallucinated and reproduced from natural image statistics, as the downsampling removes the high-frequency content. Multiple input images are required to be able to really recover a truthful high-resolution image.

Deep learning methods are nowadays also popular in the light field community. In recent work, the rich information inherent in the light field was successfully used to built deep networks for diverse tasks such as disparity estimation, view synthesis, reflection separation and intrinsic image decomposition [2, 26, 1]. The super-resolution problem was also addressed with recent CNN architectures [8, 33]. Although these approaches give impressive results, we will discuss in our paper that there is still lots of room for improvement.

One of the biggest challenges in light field super-resolution is the high dimensionality of the problem. The light field itself is a 4D data structure, its upscaled version is two or four times larger and it requires huge amounts of GPU memory to process it with adequate batch size and a sufficient number of feature maps. Thus, the plain architecture where the input image is upscaled to the desired resolution and then refined with a convolutional network is hardly applicable to light fields. Another challenge is that the light field is 4-dimensional and cannot be directly used in the current deep learning frameworks where convolution operations are performed only in 2-dimensional and 3-dimensional spaces. Finally, capturing a ground truth light field of good quality for training is a difficult task, since for example the consumer Lytro Illum plenoptic camera produces blurry images that suffer from noise, while the higher quality Raytrix camera is of plenoptic type 2.0, and thus sub-aperture views are not available. A possibility is to use a high-quality camera mounted on a gantry, which is a time-consuming approach that can hardly be performed in a wild, and is only applicable to static scenes, so training data will necessarily be limited.

Contributions. In this work, we address the problem of light field super-resolution i.e. obtaining a light field with larger spatial and angular resolution from only five sub-aperture views of a low-resolution light field. The proposed approach uses the information from neighboring views and benefits from the dense redundant structure of the light field. A fully convolutional asymmetrical encoder-decoder is built as the first network architecture to transform the 4D structure of the light field to its upscaled version. To enhance the sharpness of the reconstructed light field, we propose a novel WGAN loss that penalizes the difference between angular and spatial derivatives of the generated light field and its ground truth. Contrary to the original GAN based architectures for super-resolution [17], where the discriminator takes generated high-resolution image and the ground truth, we feed the discriminator network also with derivative information, which is much more simple and sparse than the original light field since it contains only edge information. To avoid artifacts we use Wasserstein distance in the discriminator and adversarial losses proposed by Arjovsky *et al*. [4] instead of the original GAN by Goodfellow *et al*. [10]. Those design choices make WGAN training very stable and avoid unexpected distortions we have otherwise observed. We perform both spatial and angular super-resolution with scale factors two and four, given only a sparse set of sub-aperture views. Our network achieves results with competitive quality for both artificial and real-world light fields compared to state-of-the-art conventional methods and single image SRGAN.

2. Related work

CNN-based single image super-resolution. Methods based on convolutional neural networks are widely employed for inverse problems such as deblurring [32, 27], image denoising [23, 19] and image super resolution [25, 16, 12, 19, 13]. Mao *et al*. [19] introduce a fully convolutional autoencoder architecture with skip connections to deal with single image restoration including denoising and single image super-resolution. Skip connections are proven to be very useful to guide the decoding process and increase detail. In our approach, we consequently employ 3D skip connections between our encoder and decoder. Shi *et al*. [25] propose a sub-pixel convolutional layer that maps low-resolution (LR) image to high-resolution (HR) output. Their research shows that strided upconvolution can produce checkerboard patterns in the generated high-resolution images, which we avoid by first upscaling the features spatially by the means of bicubic interpolation, thus removing the need to apply strides. Ledig *et al*. [17] employ a genera-

tive adversarial network to force the network produce more natural high-resolution images. As a modification, we introduce our DiffWGAN, which not only takes the pixel value of the generated images and the ground truth, but also the derivatives of them. This way, DiffWGAN helps to enhance the details of the generated images. Some other network architectures employ unique units to enhance or refine the HR output, such as the backprojection unit [13] and the information distillation units [16]. Instead of a straightforward deep CNN architecture, Han *et al*. [12] attempt to use a recurrent neural network for single image super-resolution.

Light field super-resolution. Due to the need of sacrificing resolution to sample angular coordinates, the light fields usually suffer from the lower spatial resolution compared to standard images. Bishop *et al*. [6] introduce a variational Bayesian framework for light field super-resolution, closely related to classical approaches [5]. Shi *et al*. [24] present the light field signal reconstruction in the frequency domain. Among recent studies, Wanner and Goldluecke [30] and Pujades *et al*. [21] propose a variational super-resolution framework using estimated high-accuracy depth maps from epipolar plane images, and solve novel view synthesis as an inverse problem. Likewise, Mitra and Veeraraghavan [20] make use of depth information of the scene to construct an inference model with a Gaussian mixture model prior. Rossi and Frossard [22] adopt a multiframe approach with a graph-based regularizer.

With the continuing success of deep learning, CNN-based light field super-resolution methods have become common. Yoon *et al*. [34] train three networks of the same architecture for spatial resolution and finally combine them to achieve angular super-resolution. Farrugia and Guillemot [7] upscale the whole light field spatially by utilizing a low-rank approximation restored by a CNN.

The recent work of Alperovich *et al*. [2, 1] inspired us to use an epipolar volume convolutional autoencoder to process light fields. The autoencoder, unlike the straight forward neural network architectures, shrinks the size of the input data in the encoding process, thus leaves more space for the number of features and batches, especially for high-dimensional training data like the light fields.

3. Outline of model and losses

In this section, we present an overview of the general structure of our deep neural network model, as well as the key formulas used to construct the loss functions of the network. We start with briefly reviewing notation and basic definitions of the light field structure, and discuss our modifications to the typical encoder-decoder model and discriminator loss as introduced in [10, 4].

Encoder-decoder model for light fields. A light field is defined on 4D ray space $\mathcal{R} = \Pi \times \Omega$, where a ray is identified by four coordinates $\boldsymbol{r} = (s, t, y, x)$, which describe the intersections with to parallel planes. Here (s, t) are viewpoint coordinates on the focal plane Π, and (y, x) are coordinates on the image plane Ω. Epipolar plane images (EPIs) can be obtained by restricting 4D ray space to 2D slices, see figure 1, while for an epipolar volume, either the s- or t-coordinate is fixed. The latter is called a horizontal, the former a vertical epipolar volume. For more information and a thorough introduction on light field geometry, we refer to [11, 18].

Figure 1. A light field is defined on a 4D volume parametrized by image coordinates (x, y) and view point coordinates (s, t). Epipolar images (EPIs) are the slices in the sx- or yt-planes depicted to the right and below the center view. As the camera moves, projections of scene points trace straight lines on the EPIs, leading to characteristic patterns.

Let L_l be the low-resolution version of the light field L. With the convolutional encoder network E, we project L_l onto the latent variable space Z to obtain the latent representation $z = E(L_l)$. Using z as an input, we generate the output high-resolution light field $U(z)$ with the convolutional decoder network U. The concatenation of encoder and decoder we term the generator G, which generates high-resolution light fields from low-resolution ones. Its output should be the same as the high-resolution ground truth L_h provided to the network.

DiffWGAN discriminator for light fields. On top of the output of the generator, we model the discriminator network D. Overall, we use a deep WGAN architecture that was originally proposed by Arjovsky *et al*. [4] with a few modifications. First, we use epipolar volumes instead of plain images, thus the generator and discriminator networks use three-dimensional convolutions. Second, we propose an additional input to the WGAN. Besides the generated patch, we also feed the discriminator with the angular and spatial derivatives. We believe that for the super-resolution task, the high-frequency components are very helpful to obtain good quality and aesthetically pleasing results. In previous work [17], it was shown that GANs help with enhancing details in the high-resolution version of the image, however, it might create some unwanted distortions and hallucinate details that are not present in the original image. Since the light field in theory has more information about the scene due to its sub-aperture views, we want to focus the attention of the discriminator in particular on the sharpness of the reconstructed details.

For any light field L'_h, our discriminator $G(L'_h)$ actually

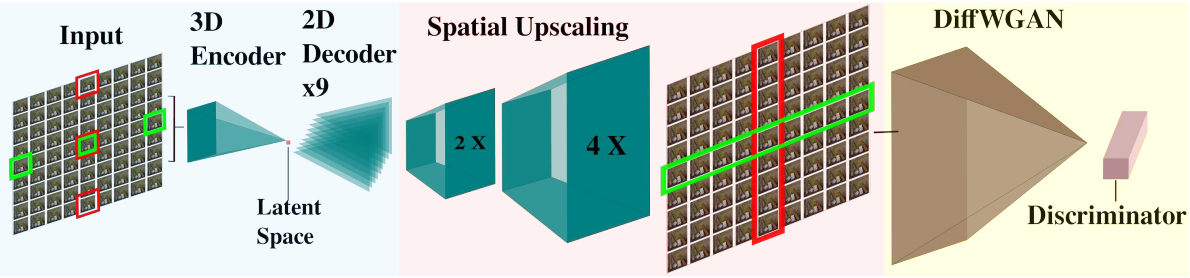

Figure 2. Proposed network architecture. The input of the network consists of three views from the vertical and horizontal stacks in the cross-hair that are framed in red and green, respectively. All views are split into 48×48 patches with 16 overlapping pixels with each of their neighbors to decrease the dimensionality of the data. The volumes of size $3 \times 48 \times 48$ are downscaled spatially to $3 \times 3 \times 3$ when they reach the latent space. The latent features are decoded and upscaled separately by nine 2D decoders to achieve the angular super-resolution. In the upscaling phase, the volumes of the decoded sub-aperture views are again spatially upscaled to twice the size of the input i.e. $9 \times 96 \times 96$ to obtain our output. To obtain the magnification factor x4, the output of the network should be passed through the network again. Finally, we introduce the DiffWGAN to distinguish the output images and the ground truth by their pixel values, as well as the spatial and angular derivatives, thus improve the details in the super-resolved images.

takes as additional input all of the derivatives of L'_h explicitly, i.e.

$$G(L'_h) = G(L'_h, \partial_s L'_h, \partial_t L'_h, \partial_y L'_h, \partial_x L'_h). \quad (1)$$

We omit this explicit dependency in the following to not clutter notation. The discriminator loss is built such that the target output for generated light fields is one, the target output for real light fields is zero. Thus, for a single low-/high-resolution training pair (L_l, L_h), the discriminator loss becomes

$$E_{\text{wgan}}(D) = \overline{D(L_h) - D(G(L_l))}. \quad (2)$$

By $\overline{\langle\cdot\rangle}$ we denote the mean value. In addition, the generator G is modeling the outputs such that they are similar to ground truth according to the discriminator D. The generator tries to take the output of the discriminator to zero, and thus minimizes

$$E_{\text{wgan}}(G) = \overline{D(G(L_l))}. \quad (3)$$

Note that in the actual implementation, the WGAN losses are split up into a sum of contributions for horizontal and vertical epipolar volumes. Minimization steps for discriminator and generator are performed in an alternating fashion.

Loss functions of our network. As the main loss function of the network, we use L^2-loss between the network's output and the high-resolution ground truth. To further enhance the detail of the output, we also calculate the L^2-loss between derivatives of the output and the ground truth. Again for a single training light field, the total reconstruction loss from directly comparing the generated to the ground truth high resolution light field reads

$$\begin{aligned} E_{\text{rec}}(G) = (1 - \exp(-L_h/0.5))\|G(L_l) - L_h\| \\ + \|\nabla G(L_l) - \nabla L_h\|, \end{aligned} \quad (4)$$

where the gradient is computed spatially.

Finally, we add the loss of the DiffWGAN for both vertical and horizontal stacks to our generator network, as described in the previous subsection. The total loss of the generator network for a single light field becomes

$$E_{\text{total}}(G) = E_{\text{rec}}(G) + E_{\text{wgan}}(G). \quad (5)$$

In the next section, we detail the architecture of the subnetworks, which we follow with a detailed explanation of training data and strategy we use to minimize the loss and prediction error.

4. Detailed network architecture

The light field has a very rigid structure linked to the scene geometry, see Fig. 1, which gives rich and redundant information about how to precisely match sub-aperture views, as required for super-resolution. In order to take full advantage of this structure efficiently, we propose a network architecture with a tailored 4D autoencoder, which is shown in Fig. 2. The inputs to our network are patch-wise vertical and horizontal epipolar volumes from the low-resolution light field "crosshair", a cross-shaped subset of views around a reference view. By using the autoencoder structure and the patch-wise input instead of the whole image, we can significantly reduce computational cost. The vertical volume consists of the top and bottom views and the center view which are marked by red frames in the left-hand side of Fig. 2, while the horizontal volume contains the left- and rightmost views and the center view that are framed in green. The high-resolution crosshair encompassing all views is fed as the ground truth.

Architecture of the encoder. Each epipolar volume has a spatial resolution of 48×48. The horizontal volume is

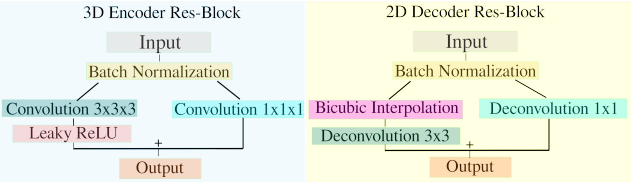

Figure 3. Residual blocks in the encoder and decoder. *Left:* The residual block of the encoder performs batch normalization, $3 \times 3 \times 3$ convolution and a leaky ReLU with $\alpha = 0.2$. For the residual connection, the input features will be transformed with a $1 \times 1 \times 1$ convolution, if necessary with stride, to achieve the correct output size before addition. *Right:* The residual block of the 2D decoder also employs batch normalization. Bicubic interpolation will upscale the features, followed by a 3×3 deconvolution with stride 1 and "VALID" padding to avoid the fabricated pixels on the border of the image. Likewise, if the input features in this block are rescaled spatially by the left chain, they are convolved with a 1×1 kernel with respective stride to allow addition.

spatially transposed such that the images exhibit the same view point motion as the vertical patches. This way, the vertical and the horizontal volumes can share the convolution kernels in the whole network. The 4D encoder has 9 layers, applying the residual block [14] to the features. This block is detailed in the left part of Fig. 3. The odd layers gradually increase the number of features, while the even layers downscale the features spatially with stride-2 convolution in the residual block. In the latent space, the vertical and horizontal volumes are downscaled from $3 \times \ 48 \times 48$ finally to $3 \times \ 3 \times 3$.

Architecture of the decoder. The first part of the 4D decoder has 9 layers as well, and generates a light field which is spatially the same size, but already has increased angular resolution. The features are spatially upscaled in 9 decoding pathways, each pathway decodes one sub-aperture view. The right side of Fig. 3 shows the structure of the decoding residual blocks. The detailed version is presented in Fig. 4. To spatially upscale the features, we grow the spatial size of them by the means of bicubic interpolation and apply unstrided transpose convolution to avoid the checkerboard artifacts in the spatial domain caused by strides [25]. At this point, the features are angularly super-resolved to the target number of nine views both in the vertical and horizontal direction.

The features of each layer in the encoder have only three vertical or horizontal views, so to prepare the skip connections for the decoder, we simply concatenate the encoded volume alone the last dimension such that we obtain the 2D features instead of 3D volumes. Besides, we leave twice as many features in the decoded volume as in the skip connection volume, otherwise, the skip connection volume will dominate the decoding process.

Spatial upscaling part of the decoder. On top of the output with increased angular resolution, we stack a spatial upscaling network. This network increases the spatial resolution of the features first to twice the size of the input, then to four times, to obtain the super-resolved output. All the spatial upscaling operations are carried out by a decoder residual block, see Fig. 4. In the second scaling phase, the features are spatially upscaled by the decoding residual block once and subsequently passed through other 2 decoding residual blocks, without their spatial resolution being changed. Afterwards, $1 \times 1 \times 1$ convolution is applied to the volumes to finally bring the number of features to the same as the input. The upscaling phase for scale factor x4 follows the same procedure, with its input coming from the intermediate results of scale factor x2 just before its $1 \times 1 \times 1$ convolution, see Fig. 2.

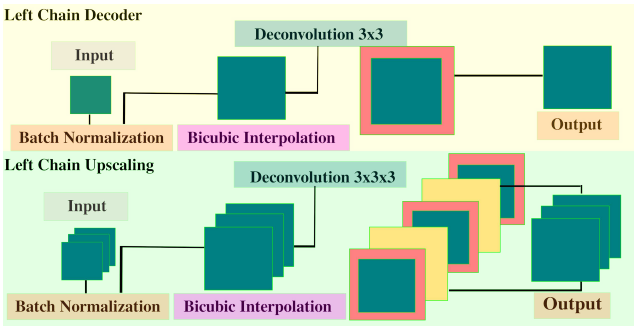

Figure 4. Residual blocks in detail. *Top:* Left chain of the residual block in the 2D decoder. The input features are first batch-normalized. Bicubic interpolation will be applied to the features for the spatial upscaling. Afterwards, the features pass through a 3×3 transpose convolution. We add 2 pixels in the spatial output size and apply stride-1 with "VALID" padding. To obtain the final output of this block, we discard again one pixel from each side of the features. *Bottom:* Left chain of the residual block in the upscaling phase. The input features are first batch-normalized. After the bicubic interpolation, the features pass through a $3 \times 3 \times 3$ transpose convolution. We add 2 pixels in the spatial and 2 views in the angular output size and apply stride-1 with "VALID" padding. To obtain the final output of this block, we discard again one pixel from each side of the features and the additional views generated at the boundaries.

Architecture of the WGAN discriminator. In order to enhance the details of the output, we add the DiffWGAN discriminator on top of our network. The discriminator input consists of the stack of concatenated light field together with its derivatives, as previously detailed in section 3. The structure is similar to the encoder, but we combine two subsequent downsampling and feature expansion layers into a single layer, making the network more shallow. In fact, the discriminator has a much easier task than the encoder, so it is reasonable to reduce capacity substantially. The discriminator distinguishes the super-resolved output from the ground truth using a Wasserstein distance on top of the features of the modified encoder chain.

Training data. The training data is generated using the

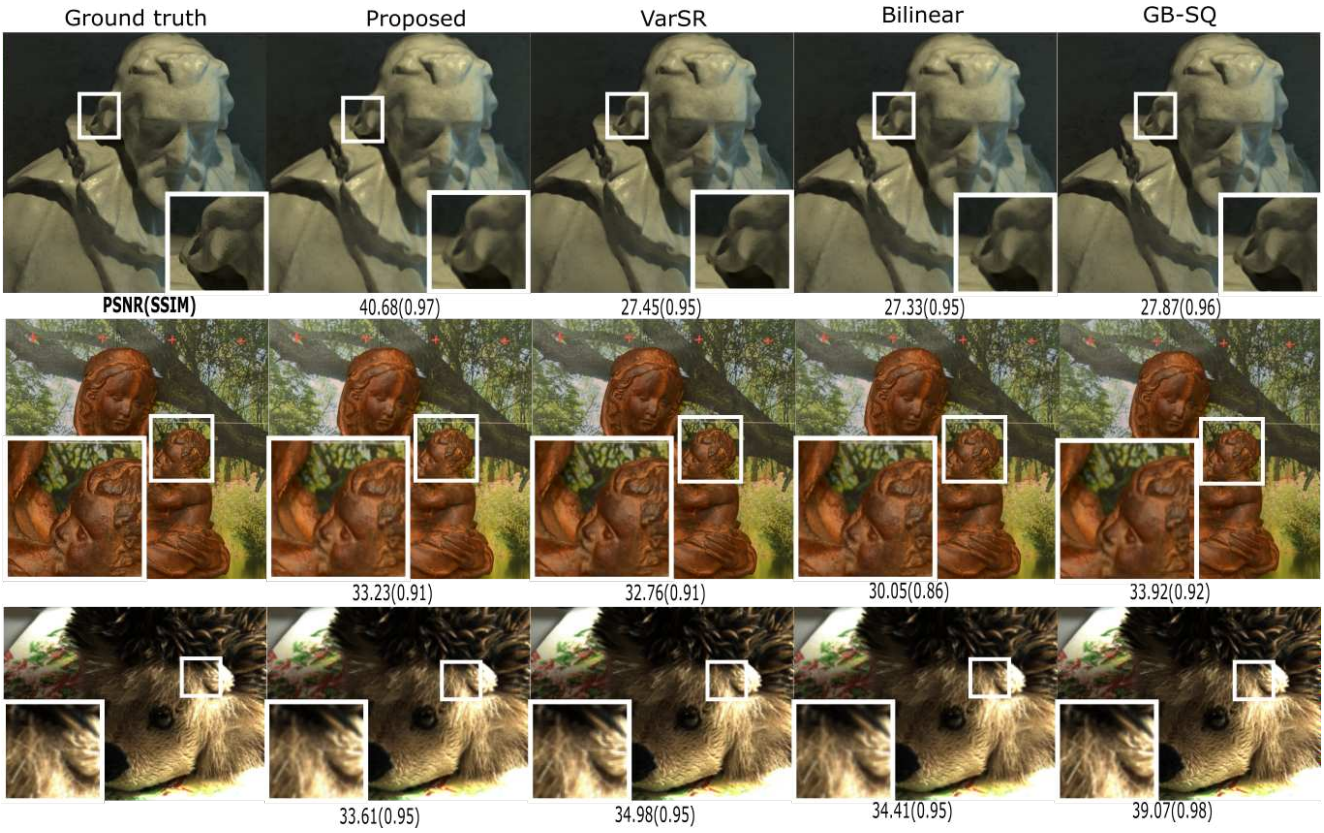

Figure 5. Visualization and quantitative evaluation of the results of scale factor x2. The images are super-resolved center views of Benchmark Cotton, HCI Maria and real-world data Hedgehog.

Blender add-on provided with [15] and follows the same procedure as in [1] for generating random light fields from a number of template scenes, objects and textures. We generated a total of 750 of these random light fields for training. In addition, we use publicly available data from the HCI database [31], the Stanford multi-camera array [29], and light fields captured with the Lytro plenoptic camera.

Training procedure. In our approach we chose the *YCbCr* color space, where the luminance component *Y* contains most of the spatial detail of the image. The channels *Cb* and *Cr* encode the blue-difference and red-difference chroma components, which are typically of low frequency. This way, we can reduce the memory requirements drastically, by following Timofte *et al*. [28] and Yoon *et al*. [34] in their observation that the *YCbCr* color space is proven to lead to the best results.

We perform training on an Intel Core i9 with 128 GB of RAM and 4 nVidia TITAN Xp GPUs. The optimization is performed with the Adam optimizer, with initial learning rate set to $1e-4$, batch size is 4. The WGAN turned out to be easy to train and will easily dominate the training [3] to produce artifacts, so we assign a small weight of $1e-3$ to the DiffWGAN loss to keep it balanced with respect to the other loss components. To satisfy the Lipschitz constraint we keep the discriminator weights in $[-5e{-}3, 5e{-}3]$ range. The loss becomes stable after approximately 10 hours, after which we dropped the learning rate to $1e-5$ and trained for another 8 hours. We then stopped training as the loss did not significantly decrease any more.

5. Experiments

Since our network is trained with patch-wise data, the output patches are reassembled to the complete high-resolution *Y* channel of the image. We compute PSNR and SSIM after the images are converted back to *RGB* color space. We evaluate our network both on publicly available light field datasets which are synthetically rendered or captured with a gantry [29, 31, 15], as well as our own data captured with Lytro Illum plenoptic camera. None of the datasets we use in evaluation has been seen during training.

Our approach super-resolves the light fields angularly from 3 views to 9 views and spatially with scale factors x2 and x4. For comparison, we picked the recent works for single image SRGAN [17] with scale factor x4, the variational approach VarSR [21] which is improved upon [30], and the graph-based method GB-SQ [22] with spatial scale factor x2.

In Fig. 5 and In Fig. 6, we visualize the super-resolved center views of scale factor x2 and scale factor x4 of the

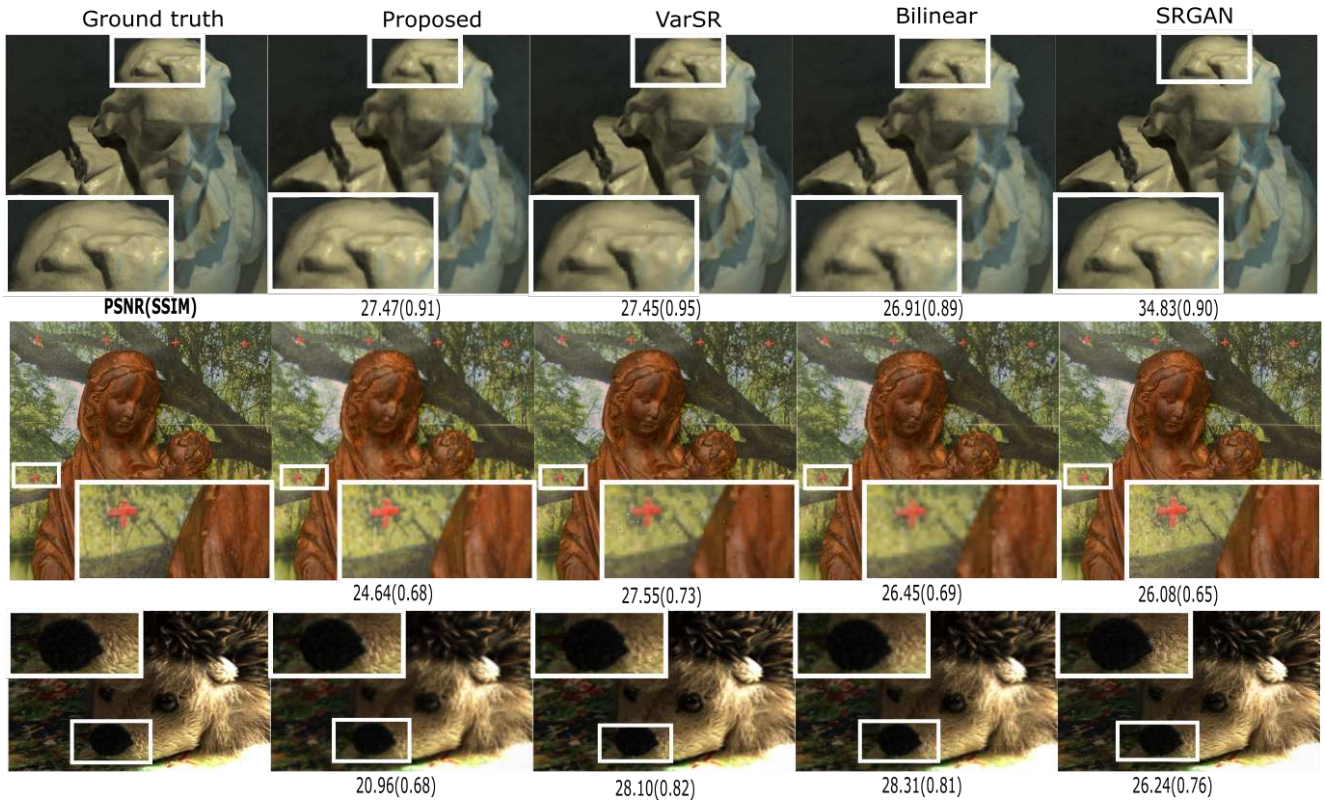

Figure 6. Visualization and quantitative evaluation of the results of scale factor x4. The images are super-resolved center views of Benchmark Cotton, HCI Maria and real-world data Hedgehog.

Benchmark Cotton, HCI Maria and real-world data Hedgehog. The PSNR and SSIM of each approach are reported below the images. One can observe over-smoothing of VarSR [21] in the scale factor x2 results, while some severe high-frequency noise appears in their scale factor x4 results. Bilinear interpolation provides rapid single image super-resolution with acceptable PSNR and SSIM, but for scale factor x4 it is significantly more blurry than the other methods. The graph-based approach GB-SQ [22] gives brilliant results in HCI Maria and real-world data Hedgehog and is visually also very sharp in Benchmark Cotton, however, its computational time means it is next to inapplicable in practice. In our experiments, it took around 8 hours to compute a 9×9 light field for scale factor x2, compared to two to three minutes on average using our approach. SRGAN seems to sometimes hallucinate additional structure, for example in the zoomed-in area of Hedgehog, the fur around the nose is sharp but apparently does not have the same shape as the ground truth. Since we use only few views to reconstruct the whole light field, our results for scale factor x4 are generally worse than state-of-the-art, but scale factor x2 reaches competing quality and even outperforms the other methods in some cases. We observe that our approach works well on the synthetic data, which can be explained by the fact that we have much more synthetic training examples than the real ones. Complete numerical results can be observed in tables 1 and 2. Note that the other methods use all sub-aperture views as an input, thus they perform only spatial super-resolution. For fare comparison, we compute PSNR and SSIM only for those views that were in the input, thus we evaluate quality of the spatial super-resolution. Additionally we illustrate numerical results for all views (newly generated and present in the input). As an ablation study we show the results of training without adversarial loss. See also Fig. 7 for additional results on the Stanford datasets.

6. Conclusions

We present an efficient approach to spatial and angular light field super-resolution based on an encoder-decoder architecture with a novel WGAN loss. Our proposed method adopts insights both from 2D [17, 28] and 3D [34, 8] CNN-based approaches to arrive at an architecture with very competitive performance. We demonstrate this in numerous experiments on public datasets [29, 31], as well as on real world light fields captured with a Lytro Illum plenoptic camera. Our architecture is simple and powerful in the sense that it fully utilizes the rich information inherited from the light field, and proves that even very few sub-aperture views are sufficient to reconstruct a high-resolution dense

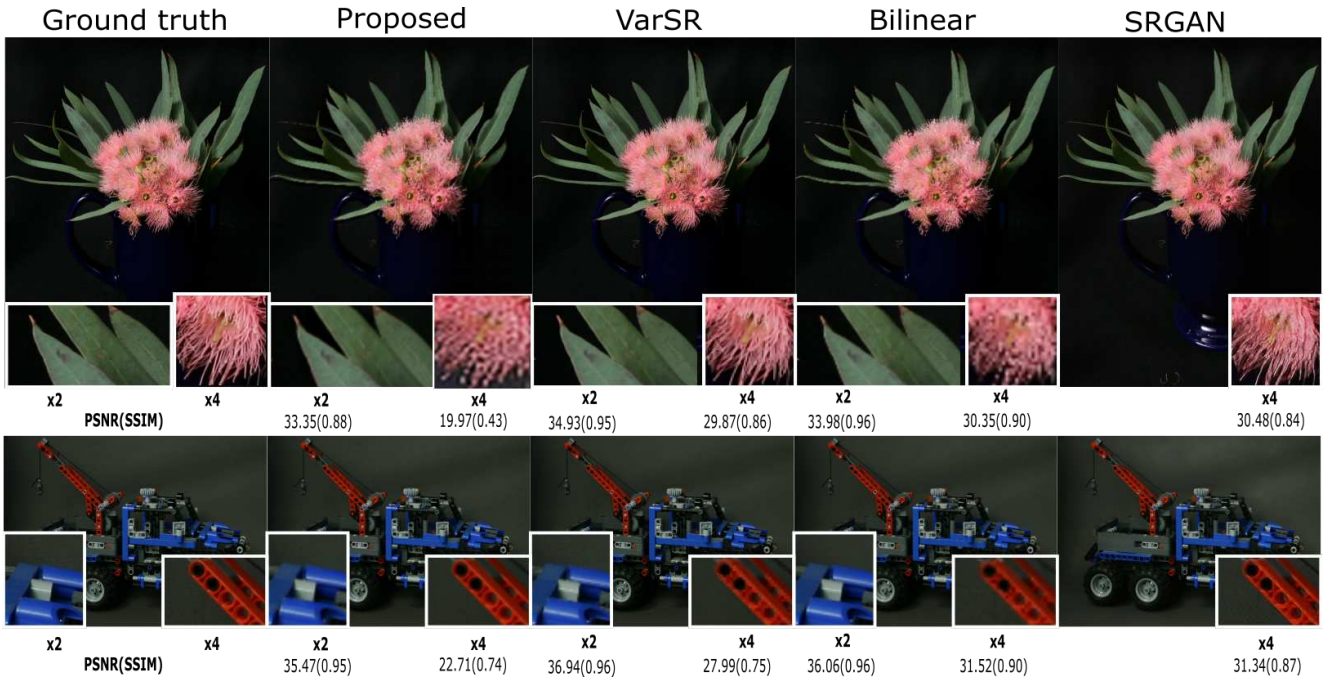

Figure 7. Results on the Stanford dataset [29]. Visually, we perform very well in particular on Truck, although PSNR does not reflect this Compared to SRGAN, our method does not hallucinate any new details. Our scale factor x4 fails to recover fine details This can be explained by small number of features in the network and lack of training examples of that kind.

Table 1. Comparison of PSNR (SSIM) for different methods, for super-resolution scale factor two and over a wide range of datasets from various sources. Note that we could not compute some results for GB-SQ (marked with an asterisk) due to excessive runtime, so we took them from their paper.

	scale factor $\alpha = 2$					
Lightfield	Bilinear	GB-SQ [22]	VarSR [21]	**Our Spatial**	**Our Full**	**No WGAN**
antinous	28.43 (0.95)	27.83 (0.96)	27.29 (0.94)	39.70 (0.97)	37.07 (0.95)	32.27 (0.95)
bicycle	27.36 (0.83)	30.73 (0.90)	28.58 (0.86)	27.04 (0.86)	26.32 (0.80)	25.95 (0.81)
tomb	29.59 (0.91)	31.81 (0.94)	30.65 (0.91)	37.27 (0.91)	36.15 (0.87)	31.61 (0.86)
bedroom	27.21 (0.86)	26.66 (0.91)	26.73 (0.87)	32.06 (0.88)	30.97 (0.84)	30.03 (0.84)
herbs	30.61 (0.83)	33.61 (0.90)	31.45 (0.86)	30.16 (0.86)	28.23 (0.75)	27.69 (0.76)
cotton	27.33 (0.95)	27.87 (0.96)	27.45 (0.95)	40.68 (0.97)	39.64 (0.96)	33.14 (0.95)
platonic	33.39 (0.90)	38.42 (0.96)	34.53 (0.92)	34.13 (0.92)	32.2 (0.84)	29.84 (0.82)
rosemary	32.24 (0.94)	37.26 (0.98)	33.75 (0.96)	32.25 (0.95)	30.74 (0.92)	29.52 (0.92)
maria	30.05 (0.86)	37.25 (*)	32.78 (0.91)	33.23 (0.91)	32.61 (0.90)	30.69 (0.88)
owl2	36.21 (0.97)	41.04 (0.98)	37.93 (0.97)	35.76 (0.95)	35.12 (0.94)	30.08 (0.87)
flowers	34.23 (0.96)	36.98 (0.98)	36.03 (0.97)	34.33 (0.91)	34.27 (0.94)	29.25 (0.83)
owl-str	32.14 (0.94)	36.46 (0.97)	32.86 (0.95)	32.65 (0.95)	31.02 (0.90)	29.36 (0.91)
origami	28.90 (0.93)	32.03 (0.95)	29.86 (0.94)	29.65 (0.95)	29.34 (0.94)	27.74 (0.91)
hedgehog	34.41 (0.95)	39.07 (0.98)	34.98 (0.95)	33.61 (0.95)	32.27 (0.92)	29.39 (0.88)
eucalyptus	33.98 (0.96)	39.09 (*)	34.93 (0.95)	33.35 (0.88)	29.62 (0.84)	27.04 (0.67)
truck	36.06 (0.96)	41.57 (*)	36.94 (0.96)	35.47 (0.95)	32.45 (0.90)	29.18 (0.86)

Table 2. Comparison of PSNR (SSIM) for different methods, for super-resolution scale factor four and over the datasets presented in the table 1.

	scale factor $\alpha = 4$					
Lightfield	Bilinear	GB-SQ [22]	VarSR [21]	**Our Spatial**	**Our Full**	**No WGAN**
antinous	28.03 (0.92)	33.81 (0.91)	26.86 (0.88)	25.75 (0.92)	25.62 (0.91)	24.33 (0.90)
bicycle	23.68 (0.63)	23.82 (0.63)	24.06 (0.67)	20.38 (0.61)	20.28 (0.59)	20.80 (0.60)
tomb	29.22 (0.83)	30.62 (0.67)	28.45 (0.75)	28.02 (0.79)	27.82 (0.78)	24.96 (0.76)
bedroom	26.02 (0.74)	28.47 (0.72)	24.76 (0.68)	25.41 (0.69)	25.38 (0.68)	24.38 (0.68)
herbs	27.28 (0.69)	26.80 (0.67)	27.41 (0.72)	21.08 (0.66)	21.01 (0.63)	21.46 (0.64)
cotton	27.34 (0.92)	34.83 (0.90)	26.91 (0.89)	27.47 (0.91)	27.35 (0.90)	25.60 (0.89)
platonic	29.07 (0.72)	27.38 (0.63)	29.63 (0.77)	23.60 (0.61)	23.52 (0.59)	21.09 (0.57)
rosemary	27.59 (0.84)	27.58 (0.85)	27.64 (0.86)	22.60 (0.80)	22.47 (0.78)	22.30 (0.79)
maria	26.45 (0.69)	26.08 (0.65)	27.55 (0.73)	24.64 (0.68)	24.56 (0.66)	23.62 (0.67)
owl2	29.74 (0.89)	29.26 (0.87)	30.68 (0.90)	22.12 (0.71)	22.02 (0.70)	19.69 (0.66)
flowers	28.64 (0.87)	29.51 (0.85)	29.54 (0.89)	20.72 (0.58)	20.61 (0.56)	18.09 (0.51)
owl-str	26.48 (0.79)	25.35 (0.75)	26.66 (0.81)	22.46 (0.75)	22.24 (0.72)	20.96 (0.72)
origami	24.09 (0.78)	22.27 (0.76)	24.18 (0.76)	20.87 (0.71)	20.81 (0.70)	19.47 (0.68)
hedgehog	28.31 (0.81)	26.24 (0.76)	28.10 (0.82)	20.96 (0.68)	20.84 (0.65)	19.88 (0.64)
eucalyptus	30.35 (0.90)	30.48 (0.84)	29.87 (0.86)	19.97 (0.43)	19.8 (0.42)	18.64 (0.37)
truck	31.52 (0.90)	31.34 (0.87)	27.99 (0.75)	22.71 (0.74)	22.61 (0.72)	19.93 (0.68)

light field of a good quality. Competing state-of-the-art light field super-resolution algorithms require many more input views and/or take a lot more time to compute. Although our method outperforms state-of-the-art approaches on the synthetic data for the scale factor x2, there is still room for the improvement for the real-world data and larger upscaling factor. One future direction is to optimize the network architecture such that it can upscale with scale factor x4 and output good-quality results. Another improvement could be to balance the number of synthetic and the real-world training data, which can increase the network performance on the real light fields. In addition we propose the new DiffWGAN loss that improves the visual quality of the results. From the experiments we show that our method does not hallucinate missing information compared to the original SRGAN network.

Acknowledgments

This work was supported by the ERC Starting Grant "Light Field Imaging and Analysis" (LIA 336978, FP7-2014) and the SFB Transregio 161 "Quantitative Methods for Visual Computing".

References

[1] A. Alperovich, O. Johannsen, and B. Goldluecke. Intrinsic light field decomposition and disparity estimation with a deep encoder-decoder network. In *IEEE European Signal Processing Conference (EUSIPCO)*, 2018. 2, 3, 6

[2] A. Alperovich, O. Johannsen, M. Strecke, and B. Goldluecke. Light field intrinsics with a deep encoder-decoder network. In *Proc. CVPR*, 2018. 2, 3

[3] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *CoRR*, abs/1701.07875, 2017. 6

[4] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 214–223, 2017. 2, 3

[5] S. Baker and T. Kanade. Limits on Super-Resolution and How to Break Them. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(9):1167–1183, 2002. 1, 3

[6] T. 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. 3

[7] R. A. Farrugia and C. Guillemot. Light field super-resolution using a low-rank prior and deep convolutional neural networks. *CoRR*, abs/1801.04314, 2018. 3

[8] R. A. Farrugia and C. Guillemot. A simple framework to leverage state-of-the-art single-image super-resolution methods to restore light fields. *CoRR*, abs/1809.10449, 2018. 1, 2, 7

[9] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar. Fast and Robust Multiframe Super Resolution. *IEEE Transactions on Image Processing*, 13(10):1327–1344, 2004. 1

[10] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. NIPS*, pages 2672–2680, 2014. 2, 3

[11] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The Lumigraph. In *Proc. SIGGRAPH*, pages 43–54, 1996. 3

[12] W. Han, S. Chang, D. Liu, M. Yu, M. J. Witbrock, and T. S. Huang. Image super-resolution via dual-state recurrent networks. *CoRR*, abs/1805.02704, 2018. 2, 3

[13] M. Haris, G. Shakhnarovich, and N. Ukita. Deep back-projection networks for super-resolution. *CoRR*, abs/1803.02735, 2018. 2, 3

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 5

[15] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke. A dataset and evaluation methodology for depth estimation on 4d light fields. In *Proc. ACCV*, 2016. 6

[16] Z. Hui, X. Wang, and X. Gao. Fast and accurate single image super-resolution via information distillation network. *CoRR*, abs/1803.09454, 2018. 2, 3

[17] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. *Proc. CVPR*, pages 105–114, 2017. 2, 3, 6, 7

[18] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, 2006. 3

[19] X.-J. Mao, C. Shen, and Y.-B. Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Proc. NIPS*, 2016. 2

[20] K. Mitra and A. Veeraraghavan. Light field denoising, light field superresolution and stereo camera based refocussing using a GMM light field patch prior. In *Proc. CVPR Workshops*, pages 22–28, 2012. 3

[21] S. Pujades, B. Goldluecke, and F. Devernay. Bayesian view synthesis and image-based rendering principles. In *Proc. CVPR*, 2014. 3, 6, 7, 8

[22] M. Rossi and P. Frossard. Geometry-consistent light field super-resolution via graph-based regularization. *IEEE Transactions on Image Processing*, 27:4207–4218, 2018. 2, 3, 6, 7, 8

[23] C. J. Schuler, H. C. Burger, S. Harmeling, and B. Schölkopf. A machine learning approach for non-blind image deconvolution. In *Proc. CVPR*, pages 1067–1074, 2013. 2

[24] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand. Light field reconstruction using sparsity in the continuous fourier domain. *ACM Transactions on Graphics*, 34(1):12:1–12:13, 2014. 2, 3

[25] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proc. CVPR*, pages 1874–1883, 2016. 1, 2, 5

[26] 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 *Proc. ICCV*, pages 2262–2270, 2017. 2

[27] J. Sun, W. Cao, Z. Xu, and J. Ponce. Learning a convolutional neural network for non-uniform motion blur removal. *CoRR*, abs/1503.00593, 2015. 2

[28] R. Timofte, R. Rothe, and L. V. Gool. Seven ways to improve example-based single image super resolution. In *Proc. CVPR*, pages 1865–1873, 2016. 2, 6, 7

[29] V. Vaish and A. Adams. The (New) Stanford Light Field Archive. `http://lightfield.stanford.edu`, 2008. 6, 7, 8

[30] S. Wanner and B. Goldluecke. Variational light field analysis for disparity estimation and super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):606–619, 2014. 1, 2, 3, 6

[31] S. Wanner, S. Meister, and B. Goldluecke. Datasets and benchmarks for densely sampled 4D light fields. In *Vision, Modelling and Visualization (VMV)*, 2013. 6, 7

[32] L. Xu, J. Ren, C. Liu, and J. Jia. Deep convolutional neural network for image deconvolution. In *Proc. NIPS*, pages 1790–1798, 2014. 2

[33] Y. Yoon, H. Jeon, D. Yoo, J. Lee, and I. S. Kweon. Learning a deep convolutional network for light-field image super-resolution. In *ICCV Workshops*, pages 57–65, Dec 2015. 2

[34] Y. Yoon, H. Jeon, D. Yoo, J. Lee, and I. S. Kweon. Light-field image super-resolution using convolutional neural network. *IEEE Signal Processing Letters*, 24(6):848–852, June 2017. 3, 6, 7