SIGNET: Efficient Neural Representation for Light Fields

Brandon Yushan Feng, Amitabh Varshney
University of Maryland, College Park
{yfeng97, varshney}@umd.edu

Abstract

We present a novel neural representation for light field content that enables compact storage and easy local reconstruction with high fidelity. We use a fully-connected neural network to learn the mapping function between each light field pixel's coordinates and its corresponding color values. Since neural networks that simply take in raw coordinates are unable to accurately learn data containing fine details, we present an input transformation strategy based on the Gegenbauer polynomials, which previously showed theoretical advantages over the Fourier basis. We conduct experiments that show our Gegenbauer-based design combined with sinusoidal activation functions leads to a better light field reconstruction quality than a variety of network designs, including those with Fourier-inspired techniques introduced by prior works. Moreover, our SInusoidal Gegenbauer NETwork, or SIGNET, can represent light field scenes more compactly than the state-of-the-art compression methods while maintaining a comparable reconstruction quality. SIGNET also innately allows random access to encoded light field pixels due to its functional design. We further demonstrate that SIGNET's super-resolution capability without any additional training.

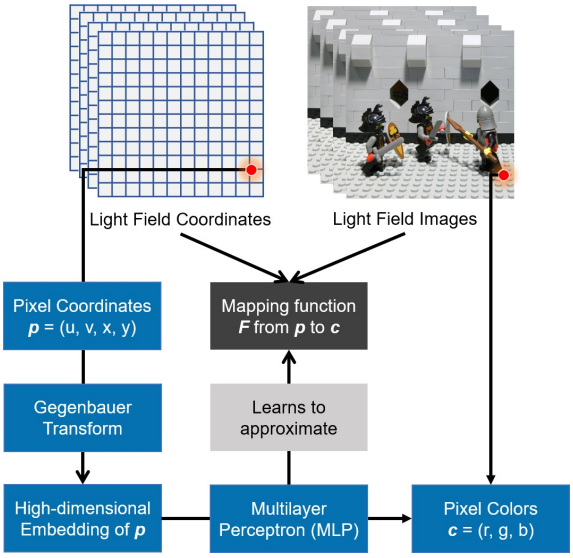


Figure 1: Overview of SIGNET. We train a MLP to approximate the mapping function from each pixel's coordinates to its color values. Our input transformation strategy based on the Gegenbauer polynomials enables the MLP to more accurately learn the high-dimensional mapping function.

1. Introduction

Light fields offer an information-rich medium for static and dynamic scenes. However, a significant barrier to their widespread adoptions is a lack of sufficiently compact representations of such high-dimensional data, making it impractical for efficient storage, editing, and streaming. For example, a 1080p 60-fps light field video captured on a 10×10 camera grid easily requires several gigabytes of storage space for every second of content.

A straightforward solution to compressing light fields is to apply existing, widely used compression methods such as JPEG and MPEG. However, due to the sheer amount of images captured in a light field, the compression rate of these single-view-based methods are far from satisfactory [48, 49]. Therefore, it is imperative to have a compact way to represent light fields by taking advantage of the overlapping and repetitive visual patterns in light fields.

Extensive research has been devoted to designing compact light field representations based on the patch-based compression strategy manifest in the JPEG standard. These methods represent each image patch as a weighted sum of a small dictionary of basis functions, and the goal is finding new ways to construct dictionaries of basis functions that achieve better compression results. Yet, previous efforts have limited success in enabling easy transmission and manipulation of light field content.

Recent advances in deep learning have led to impressive results in representing data like images and volumes [31, 43, 47] with neural networks. A common thread among these methods is incorporating Fourier-inspired modifications to the classical neural network design called multilayer perceptron (MLP). Specifically, the SIREN network [43] uses

a sinusoidal activation function between the MLP layers, while neural radiance field (NeRF) networks [31] designed for volumetric radiance data show the effectiveness of applying cosine and sine transformations on input coordinates. The improvement brought by the Fourier basis used in NeRF is further analyzed and formalized by Tancik *et al*. [47], who also successfully extend the neural representation to data like 2D images and 3D shapes.

The proven capability of MLPs to express visual content with high fidelity implies that we could potentially compress a gigapixel light field within a few megabytes. However, as shown in Section 4, the previous techniques fall short of representing light fields without visible artifacts.

In this work, we present a new framework that efficiently and accurately represents light field content using neural networks. Crucially, we introduce a novel input transformation strategy of the multi-dimensional light field coordinate based on the orthogonal Gegenbauer polynomials, which in our experiments work very well with the sinusoidal activation functions between the MLP layers. We call this network SIGNET (SInusoidal Gegenbauer NETwork), and we show its superiority for light field neural representation over a variety of Fourier-inspired input transformation strategies. SIGNET also achieves outstanding reconstruction quality with a higher compression rate than state-of-the-art dictionary-based light field compression methods. We further demonstrate how our MLP-based approach easily allows for view synthesis and super-resolution on the encoded light field scenes.

In summary, our contributions are as follows:

- We present a neural representation of light fields which achieves high reconstruction quality and compression rate and offers pixel-level random access to the encoded light field.
- We introduce an input transformation strategy for coordinate-input MLPs using Gegenbauer polynomials, which outperforms other recently proposed techniques on light field data.
- We show such a neural representation enables high-quality decoding at novel coordinates without additional training, achieving super-resolution along spatial, angular, and temporal dimensions on light fields.

2. Related work

Light Field Compression. Traditional compression relies on classical coding strategies that typically involve analytical basis functions such as the Fourier basis and wavelets. Prior research has augmented this analytical approach with disparity [9, 21, 26, 38] and geometry information [52]. Some sophisticated applications of light field video [7, 22, 33] also integrate motion prediction and build on existing video codec algorithms such as HEVC (H.265) [45] and VP9 [32]. More recently, Le Pendu *et al*. [35] present a Fourier Disparity Layer representation for light fields, which allows upsampling [37] and compression [12, 36] in the Fourier domain.

A different approach towards light field compression involves learning a dictionary of basis functions, which is inspired by progress in sparse coding from machine learning, where dictionaries learned with data-driven algorithms have been shown to outperform analytical basis functions [1, 27, 30, 44]. However, the dictionaries learned with conventional algorithms such as K-SVD [4] still contain too much redundancy and have a high storage cost. The current state-of-the-art methods [20, 29] for light field compression improve this approach by learning an ensemble of orthogonal dictionaries with a novel pre-clustering strategy.

We present a novel approach to this task by learning a neural representation of light fields. While our approach is rooted in the idea of basis functions, we fundamentally differ from the previous methods as we use the expressive power of neural networks with non-linear activation functions to combine the basis functions into the desired output.

Light Field Interpolation. Most approaches rely on proxy information such as depth or optical flow [8, 10, 13, 14, 28, 41]. Recently, deep learning methods have been used to infer depth and optical flow from light fields, and render novel viewpoints [6, 16, 24, 50, 51]. These methods warp the original frames to a novel viewpoint. While the results are impressive, they require access to the original light field data at run-time, incurring additional, sometimes prohibitive, costs to the light field processing pipeline.

In this paper, we show how our neural light field representation naturally enables interpolation from the compressed data without explicit learning or proxy information. Although our presented network is not specifically designed for light field super-resolution or view synthesis, our results show its promising potential to be adapted for such tasks.

Coordinate-input MLP Recent research [31, 43, 47] has shown the potential of using coordinate-input MLP networks to represent various data. The Fourier-inspired transformation achieves state-of-the-art free viewpoint synthesis on static scenes [31]. The sine activation, introduced in SIREN [43], allows a simple MLP with raw coordinate inputs to accurately model the coordinate-to-color mapping of data including images and videos. However, our experimental results show that these Fourier-inspired methods are unable to accurately model the coordinate-to-color mapping in light fields. We present a new transformation that allows the MLPs to successfully represent dense light fields, and we show its applicability for compactly representing high-resolution light fields.

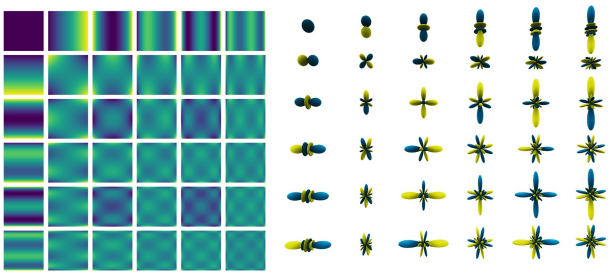


Figure 2: **Illustration of Gegenbauer (Ultraspherical) Polynomials.** We evaluate the 2D Gegenbauer basis functions on a 2D Cartesian grid (left) and a 3D polar grid (right). Only the first six orders of the basis are selected for illustration purposes.

Gegenbauer Polynomials. Previous research in applied mathematics has shown the effectiveness of Gegenbauer polynomials, also known as ultraspherical polynomials, in addressing the Gibbs phenomenon [18], which is a commonly observed artifact in MRI reconstruction using Fourier-based approximations [5, 19]. It has been shown by Gottlieb *et al*. [18] that the finite Gegenbauer expansion of such functions provides a better convergence and usually resolves the Gibbs artifact using fewer basis functions than the Fourier approach. Specifically, they show that given the first N Gegenbauer expansion coefficients, we can construct an exponentially convergent approximation to the point values of $f(x)$ in any sub-interval where f is analytic [17].

Moreover, recent studies on machine learning also show the usefulness of Gegenbauer kernels [3, 34] or hyperspherical loss functions [15, 25]. Our experiments show applying Gegenbauer transformation on light field coordinates not only improves reconstruction quality but also results in a faster neural network convergence.

3. Overview

3.1. Light Fields as Functions

Our goal is to find an accurate approximation to the mapping function $\mathcal{F}$ for a given light field, from which we could retrieve the color value of any pixel. Moreover, such a functional approximation could be parameterized using fewer bits than the original light field content, providing us with a compressed representation of the light field.

A noteworthy feature of this functional approach towards light field representation is that we could arbitrarily decode any pixel within the given light field, providing random access to the compressed data which ensures efficiency in content retrieval and streaming. Most previous compression methods discussed in Section 2 involve encoding and decoding blocks of pixels, and many video compression methods even require information from the previous frame to decode the current frame.

3.2. Function Approximations

A single-channel image represented as a 2D function $\mathcal{F}(x, y)$, could be approximated by the weighted sum of N orthogonal basis functions $\Theta(x, y)$:

$$\tilde{\mathcal{F}}(x, y) = \sum_{i=1}^{N} a_i \Theta_i(x, y) \tag{1}$$

Assuming the set of orthogonal functions is known, an image could be recovered by just using the coefficients $\{a_i\}_{i=1}^{N}$. Thus, image compression is reduced to compressing this set of coefficients. In the case of JPEG compression, cosine functions are used as the Θ_i's, and the coefficients $\{a_i\}_{i=1}^{N}$ are quantized and entropy-encoded.

If we use a similar, analytical formulation for 4D light fields, we would obtain the following approximation:

$$\tilde{\mathcal{F}}(u, v, x, y) = \sum_{i=1}^{N} a_i \Theta_i(u, v, x, y) \tag{2}$$

However, instead of analytically calculating the coefficients a_i, we propose using an L-layer MLP to compute:

$$\tilde{\mathcal{F}}(u, v, x, y) = \phi_L \circ \phi_{L-1} \circ ... \circ \phi_1([\Theta_i(u, v, x, y)]_{i=1}^{N}) \tag{3}$$

Here, ϕ_l stands for the l-th layer of the neural network with a weight matrix W_l, bias vector b_l, and an activation function σ . The output from each layer is $\phi_l(x) = \sigma(W_l x + b_l)$. We next discuss why this approach is preferred over computing analytic coefficients.

3.3. MLP for Approximation

This MLP-based formulation shares several similarities to the classic Fourier expansion method. In fact, for the 1D function case, a MLP could be constructed that has the same representation capacity as a Fourier expansion:

$$f_N(x) = \sum_{n=-N}^{N} a_n \cdot \exp(i \cdot 2\pi n x), x \in [0, 1], N \in \mathbb{Z} \tag{4}$$

This Fourier expansion is equivalent to a special two-layer MLP with activation function $\sigma(x) = \exp(ix)$. The first layer of this MLP would be a $1 \times 2N$ matrix with values $\{2\pi n\}_{n=-N}^{N}$, while the second layer would contain the $2N \times 1$ Fourier coefficients $\{a_n\}_{n=-N}^{N}$.

The same analogy generalizes to multi-dimensional inputs. For example, the Fourier expansion of a 2D function is of the following form:

$$f_{N,M}(x, y) = \sum_{n=-N}^{N} \sum_{m=-M}^{M} a_{m,n} \cdot \exp[i \cdot 2\pi(nx + my)] \tag{5}$$

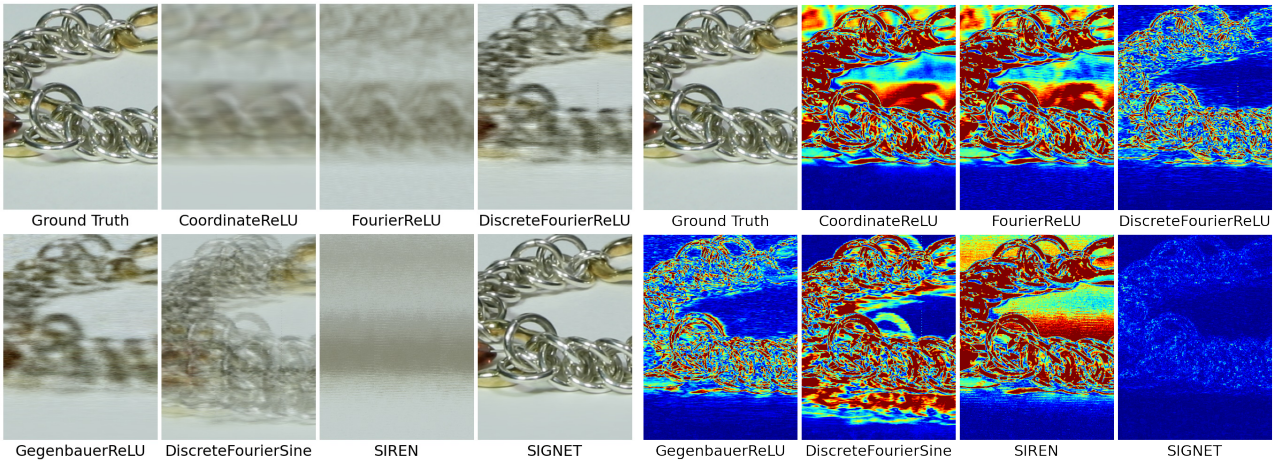


Figure 3: **Examples of reconstructed images (left) and absolute errors (right).** SIGNET achieves good accuracy while other methods find encoding this scene challenging. Here we only present the *Bracelet* scene; more qualitative results for other light field scenes may be found in the Supplementary Material.

with $x, y \in [0, 1]$ and $N, M \in \mathbb{Z}$. Notably, the Fourier coefficients $\{a_{m,n}\}$ are grouped together as a 2D matrix. While this seems incompatible with the MLP described earlier, we could flatten this 2D matrix into 1D and fit this expansion into the form of a two-layer MLP. For example, the input to the MLP would be $[x, y]^T$, the first layer could be a $2 \times 4NM$ matrix with the columns populated by every combination of (n, m), and the second layer could be a $4NM \times 1$ matrix that contains the Fourier coefficients $\{a_{m,n}\}$.

Given the examples in 1D and 2D, we can see how the same derivation naturally extends to higher dimensions. However, increasing the dimensions would lead to a combinatorial growth of number of coefficients to consider. More importantly, it would be hardly meaningful to have a MLP with even more parameters than the Fourier coefficients. Therefore, we need a MLP with far fewer parameters while ensuring that it can approximate the multi-dimensional function for light fields.

3.4. Towards Multi-dimensional Input

Recent works [39, 43, 47] have shown that typical MLPs with coordinate inputs suffer from a spectral bias when trying to approximate multi-dimensional functions like images. Two recent techniques modify coordinate-input MLPs to enable them to successfully learn on data with high-frequency details such as natural images and volumes. SIREN [43] uses sine function as the activation function between network layers, while FourierMLP [47] shows the effectiveness of transforming the input ($[x, y]$ in the 2D image case) using the cosine and sine functions as a basis.

Recalling Euler's formula, $e^{ix} = cos(x) + i \cdot sin(x)$, we observe that these two techniques have similar elements to Equation 5, where we show the equivalence between 2D discrete Fourier transform and a particular MLP. It is therefore not surprising that the coordiate-input MLP finally achieves accurate representation when it is empowered by the periodicity behind Fourier expansions, either from the sine activation used in SIREN or from the sine and cosine input transformations in FourierMLP.

Moreover, in NeRF and FourierMLP, the coordinate along each dimension is independently transformed, and the respective high-dimensional embeddings are concatenated together as input to the MLP. This concatenation is crucial since it elegantly avoids the combinatorial explosion of multi-dimensional basis.

For instance, for a 2D image with cosine bases of orders N and M along each dimension, it is sufficient to simply calculate $\{\cos(nx)\}_{n=-N}^{N}$ and $\{\cos(my)\}_{m=-M}^{M}$ in 1D and then concatenate them into a $1 \times 2(N+M)$ input, instead of the $1 \times 4NM$ input from $\{\cos(nx + my)\}_{n,m}$.

This modification enables MLPs to make use of multi-dimensional orthogonal basis functions without the quadratic increase in the input size. Therefore, we adopt this concatenation strategy for learning light fields without training an exceptionally wide MLP.

3.5. Gegenbauer Basis

While these Fourier-inspired transformation techniques are shown to work really well for images and volumes, we are unable to get satisfactory results when using them with MLPs to represent light fields as discussed in Section 4. Instead, inspired by the benefits of Gegenbauer reconstruction over Fourier-based reconstruction discussed in Section 2, we develop an input transformation strategy using Gegenbauer polynomials as basis functions. The n-th order

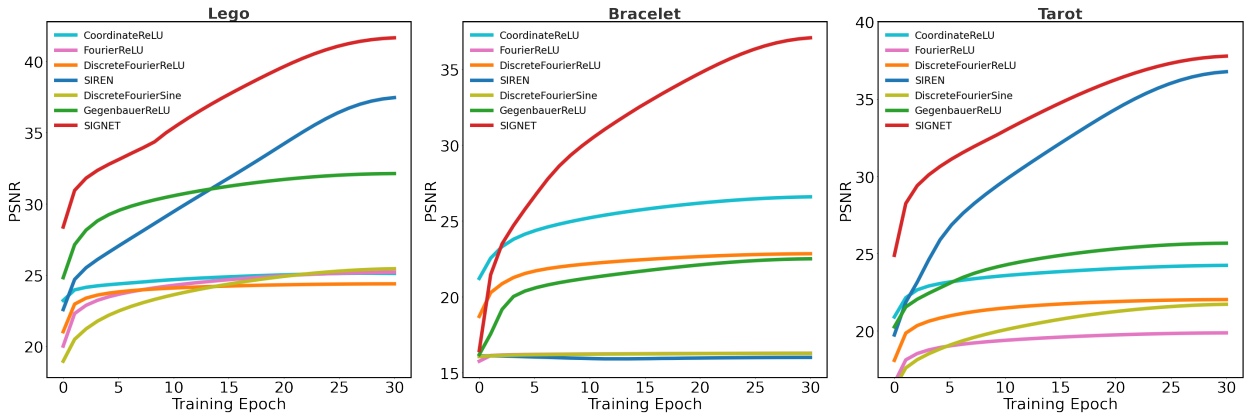


Figure 4: **Training PSNR on the static light field scenes.** Compared to other variants of coordinate-input MLP, SIGNET clearly has a faster convergence and a higher quality in representing the 4D light fields.

Table 1: **Different MLP used in experiments.** Definitions of input transformation methods are in Section 4.2

Network	Transformation	Activation
CoordinateReLU	None	ReLU
FourierReLU	Gaussian Fourier	ReLU
DiscreteFourierReLU	Discrete Fourier	ReLU
SIREN	None	Sine
DiscreteFourierSine	Discrete Fourier	Sine
GegenbauerReLU	Gegenbauer	ReLU
SIGNET	Gegenbauer	Sine

Gegenbauer polynomials could be computed recursively as:

$$G_{n+1}^{(\alpha)}(x) = \frac{1}{n}[2x(n+\alpha-1)G_n^{(\alpha)}(x) - (n+2\alpha-2)G_{n-1}^{(\alpha)}(x)], \tag{6}$$

where $-1 \leq x \leq 1$, $G_0^{(\alpha)}(x) = 1$, and $G_1^{(\alpha)}(x) = 2\alpha x$.

4. Methods

4.1. Proposed Framework

To reconstruct pixels of a 4D light field from SIGNET, a coordinate-input vector $p = [u, v, x, y]$ is transformed with a set of functions $\mathcal{S}_i$ as

$$\mathcal{E}(p) = [\mathcal{S}_1(u), ..., \mathcal{S}_{C_u}(u), \mathcal{S}_1(v), ..., \mathcal{S}_{C_v}(v), ..., \mathcal{S}_1(x), ..., \mathcal{S}_{C_x}(x), ..., \mathcal{S}_1(y), ..., \mathcal{S}_{C_y}(y)]$$

C_u, C_v, C_x, C_y are the maximum orders of basis functions used to map the coordinates along each dimension. In our case, we use the Gegenbauer basis functions to transform the input by setting $\mathcal{S}_n(z) = G_n^{\alpha}(z)$ as defined in Section 3.5. We adopt the sine activation presented by Sitzmann *et al*. [43] as it enhances the ability of a MLP to approximate functions even without an input transformation.

4.2. Comparative Evaluation

Having discussed the motivations for using MLP for representing light fields and introducing the Gegenbauer input transform, we need to validate this approach through qualitative results for MLP light field reconstruction. To objectively evaluate our network, we compare it with several other networks with different transformation strategies and activation functions (see Table 1).

The specific input transformation strategies in the second column in Table 1. *None* transformation means $\mathcal{S}_n(z) = z$ and that $C_u = C_v = C_x = C_y = 1$. With *Discrete Fourier*, the transformation function returns a tuple as $S_n(x) = [cos(2\pi nx), sin(2\pi nx)]$. With *Gaussian Fourier*, we adopt the Fourier-based transformation presented in FourierMLP [47], with the scale set at 5. This is equivalent to having $\mathcal{E}(p) = [cos(Bp^\top), sin(Bp^\top)]$ with B being a $\frac{C}{2} \times 4$ matrix with entries randomly initialized from a Gaussian distribution $\mathcal{N}(0, 5)$, and $C = C_u + C_v + C_x + C_y + C_t$.

4.3. Data and Training Setup

For static light fields, we use the Stanford Light Field Archive [2]. For light field videos, we choose the Technicolor Natural Light Field Video dataset [40]. We select these specific scenes used by Miandji *et al*. [29] for a fair comparison of performance. The details of these datasets may be found in the Supplementary Material.

We implemented the networks in PyTorch and follow the same training scheme and random seed to ensure reproducibility. More details may be found in the Supplementary Material. We train all networks for 30 epochs, each taking around 12 minutes on an NVIDIA GeForce RTX 2080 Ti.

Table 2: **Compression Performance Compared to Other Methods.** Values in the column *Size* denote the storage in megabytes(MB) for each method without further quantization. Details of the other listed methods may be found in Miandji *et al*. [29]. The storage cost of SIGNET is calculated based on the number of MLP parameters required (see Table 4 in Supp. Material) to reconstruct all pixels in each scene.

	Static Light Fields									Light Field Videos					
	Lego			Bracelet			Tarot			Painter			Trains		
Method	Size	PSNR	SSIM	Size	PSNR	SSIM	Size	PSNR	SSIM	Size	PSNR	SSIM	Size	PSNR	SSIM
SIGNET	**9.0**	**41.26**	**0.976**	**12.0**	38.70	0.973	**9.0**	37.47	0.975	**144**	**39.56**	**0.934**	**144**	**39.73**	**0.968**
AMDE [29]	29.3	40.90	0.973	18.1	**39.90**	**0.980**	44.2	38.54	0.973	941	38.25	0.929	809	37.00	0.946
KSVD [4]	29.3	38.39	0.959	18.1	36.73	0.973	44.3	**38.81**	**0.980**	942	38.12	0.928	807	35.06	0.928
HOSVD [29]	29.3	37.24	0.958	18.0	33.98	0.962	44.3	34.53	0.966	942	36.91	0.919	807	35.29	0.937
5D DCT [29]	29.4	37.29	0.955	18.1	32.31	0.952	44.2	33.03	0.960	941	36.79	0.915	807	35.20	0.934
CDF 9/7 [11]	29.0	33.71	0.914	18.2	31.98	0.939	44.3	29.17	0.865	941	31.69	0.822	1116	29.80	0.746

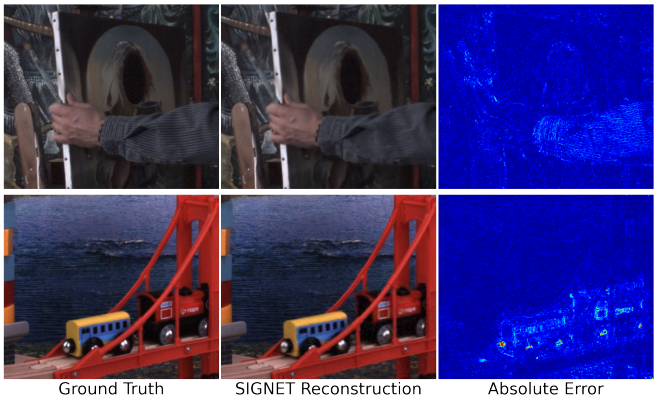


Figure 5: We show examples of reconstruction on light field video scenes *Painter* and *Trains*.

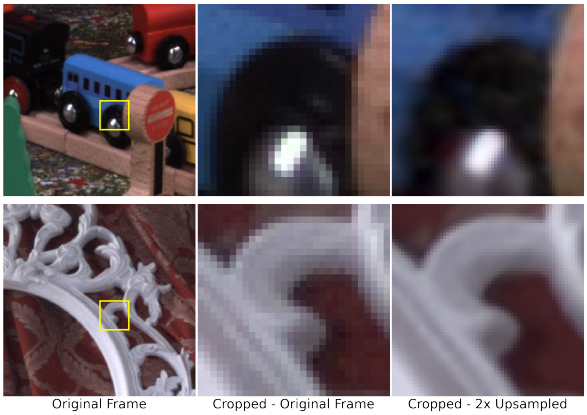


Figure 6: **Spatial Upsampling.** We evaluate the trained SIGNET on dense sampling grid points in the spatial dimensions. We show zoomed-in details in the cropped region bounded by the yellow rectangle.

5. Results

5.1. Static Light Field Reconstruction

We test the effectiveness of SIGNET and compare it against the other configurations listed in Table 1 with the same model size. For a fair comparison, we use the same number of basis functions (C_u, C_v, C_x, C_y) for the three input transform strategies: **GaussianFourier**, **DiscreteFourier** and **Gegenbauer**. We train each type of MLP on the three static light field scenes and present example results in Figure 3. More results are in the Supp. Material.

We observe that SIGNET leads to higher quality reconstructions. The network with Gegenbauer transformed input not only produces more accurate reconstruction than with other transform strategies, its results are also more visually stable. Even without the sine activation (GegenbauerReLU), the Gegenbauer transformation improves the performance of the basic ReLU MLP (CoordinateReLU), and it even achieves better performance averaged over the three scenes than Fourier-based MLPs (FourierReLU, DiscreteFourierReLU, and DiscreteFourierSine). We also show the PSNR values during the training progression for each type of MLP in Figure 4. The PSNR curves further corroborate the superiority of SIGNET, which clearly shows faster convergence and higher accuracy. Another takeaway from these results is that the success of our method is not entirely due to the sinusoidal activation (see SIREN versus SIGNET), and that the Gegenbauer transformation is indeed necessary and effective for more accurate light field representations. Therefore, for the rest of the paper, we adopt SIGNET as our default MLP setup.

Furthermore, SIGNET not only accurately reproduces the RGB values at each pixel location, but it is also parsimonious in storage cost. In Table 2 we compare the compression and reconstruction result of SIGNET against previous methods. On all three static light field scenes, we achieve reconstruction quality on par with the state-of-the-art methods while requiring much less storage. For reproduction, the Supplementary Material provides the specific network setup for the results shown in Table 2.

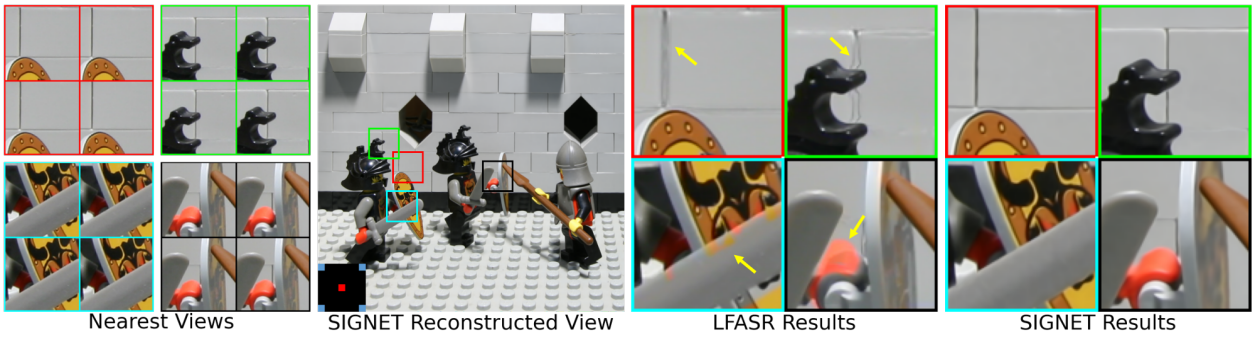


Figure 7: **Angular Upsampling.** At the bottom left corner of the reconstructed view, we show the relative positions of the reconstructed view (red square) and its four nearest views (blue squares) in the original light field. We present reconstructions at novel viewpoints from the three static scenes, and we also show results from the deep-learning-based method, LFASR [23], which is trained specifically for light field angular upsampling. Notice the LFASR results show visible artifacts pointed to by the yellow arrows, such as distorted geometry and ghosting.

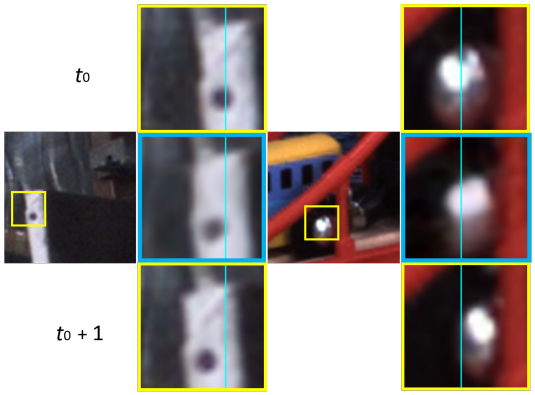


Figure 8: **Temporal Upsampling**. t_0 and $t_0 + 1$ are consecutive frames in the original video. The blue boxes contain output from frames evaluated at $t_0 + \frac{1}{2}$, which is not present in the original video. The vertical lines are drawn for easier observation of the motion trajectory.

5.2. Extension to Light Field Videos

After observing the good performance of SIGNET on static 4D light fields, we extended the same framework to light field videos, where effective compression is even more important. Since a light field video contains many more images than a static light field, it is challenging to only train one network to approximate the 5D function that covers the entire video. Therefore, we divide a light field video into smaller blocks such that each block is independently learned by a small SIGNET. Details of our temporal division may be found in the Supplementary Material.

We show example results in Figure 5 and compare the compression results with previous methods in Table 2. SIGNET has a clear-cut advantage both in terms of reconstruction quality and compression size.

5.3. Light Field Super-Resolution

SIGNET's unique advantage of is it can evaluate arbitrary coordinates, thus allowing interpolation of viewpoint, as well as spatial and temporal coordinates.

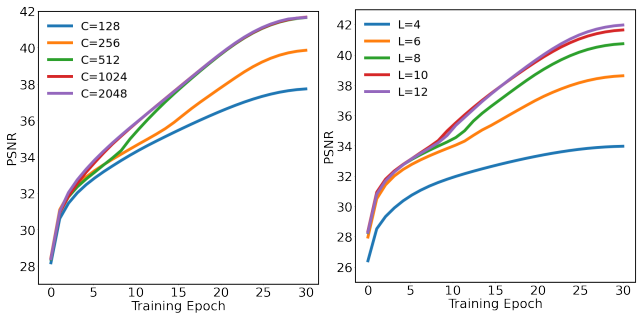
In Figure 6, we show the results of spatial upsampling with the trained SIGNET. We observe that the results do not have any perceptible artifacts. Furthermore, these results suggest that SIGNET does not merely memorize the training samples, but it also gracefully interpolates among the unsampled coordinates.

In Figure 7, we further show the upsampling results for the angular dimensions. We compare our results with LFASR [23], a state-of-the-art method that is representative of most learning-based methods that rely on depth estimation. Note that our novel views are generated without any depth information, and we do not explicitly use any information from adjacent images. SIGNET achieves a similar level of visual quality to the learning-based method equipped with deep CNNs, while avoiding the mismatching artifacts that stem from inaccurate depth or optical flow estimation. The fact that SIGNETs can generate views at unseen viewpoints implies that they actually store far more images than just the original data, which could significantly increase the effective compression rate.

Finally, we show in Figure 8 the result of upsampling along the temporal dimension on light field videos. Such results again demonstrate the trained network not only memorizes the training samples, but also implicitly derives the motion pattern between two frames.

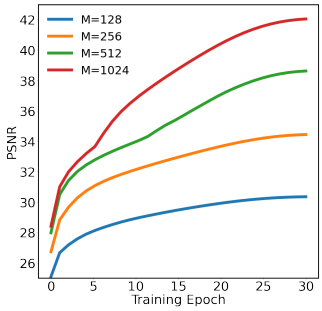
5.4. Ablation Studies

For simplicity, in this section we use the static light field scene, *Lego*, to train our networks. We first examine the effect on network performance when we modify C, the total number of Gegenbauer basis orders. We set hidden layer

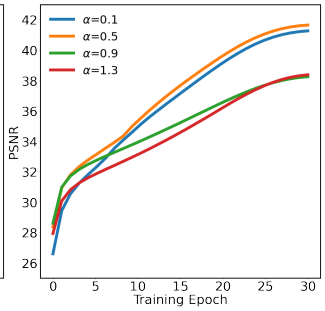


(a) Network Input Size C.

(b) Network Depth L



(c) Network Width M



(d) Gegenbauer's α

Figure 9: **Ablation Studies.** We change different aspects of SIGNET as discussed in Section 5.4.

length $L = 10$ and test network performance with different input sizes $C = \beta C_u + \beta C_v + \beta C_x + \beta C_y$ with $\beta = 0.25, 0.5, 2, 4$ and the Cs same as in Table 4 in Supp. Material. In Figure 9a we show the training curves, and the performance gain from increase in the input basis size apparently diminishes after $C > 1024$.

We then investigate how the network performance changes as we modify the number of network layers, L. We set the dimension of all intermediate layers to be 512×512. In Figure 9b, we show how the reconstruction quality changes for different configurations. The network accuracy saturates as the network expands to 10 layers.

In Figure 9c we show the effect of changing the matrix dimension size M for all hidden layers. We set the hidden layer length as $L = 4$ and the number of input basis functions as $C = 512$. As expected, we do see higher quality with more bases, although the quality also saturates to a certain ceiling. Increasing L seems to be a more effective way to increase SIGNET capacity and approximation accuracy, since it only moderately increases the storage cost; incresing M barely improves performance but significantly increases the storage cost.

We also examine the impact of varying α, a hyperparameter of Gegenbauer polynomials. In addition to our default choice of $\alpha = 0.5$, we test several other values and we show the results in Figure 9d. The results suggest that our default choice is reasonable, since the network performance seems to decline as α becomes larger than 0.5.

6. Discussion and Limitations

Compared to prior work on light field compression, SIGNET distinguishes itself with the ability to decode at coordinates not captured in the original data, thanks to its functional design.

Our results could be further enhanced by traditional image coding schemes which quantize and further encode spectral coefficients and residuals. As a quick check, we implemented quantization and weight pruning on our network weights. Our preliminary results show a further 15-fold bitrate reduction while PSNR decreased by less than 0.1. Additional weight compression techniques such as knowledge distillation could further enhance our compression rates. In contrast to prior work, SIGNET would not require sending the original light field images to the users; only the MLP weights are sufficient for decoding a high-resolution and densely sampled light field at arbitrary viewpoints.

While SIGNET achieves high reconstruction quality, a limitation is we need to retrain the network for every new scene. Latest advances [42, 46] have presented promising results in using meta-learning to speed up the training of coordinate-input MLPs. We believe that meta-learning strategies are likely to drastically reduce the training cost, and we leave explorations on this idea for future work.

Although we briefly showcase the ability of the trained networks to perform upsampling, we do not focus on further exploiting its potential in this direction. Our method is not guaranteed to achieve satisfactory view synthesis results on sparsely sampled light fields (e.g. light field with 3×3 viewpoints with a large disparity), since there is insufficient data along the angular dimensions for the network to approximate smooth interpolations. In the future, it would be desirable to design neural networks that can generalize across different scenes utilizing learned prior knowledge.

7. Conclusion

We present SIGNET, a novel framework to represent light fields with neural networks, which achieves high-fidelity reconstruction and state-of-the-art compression performance. We hope SIGNET could motivate more research into utilizing neural networks for processing light fields and other high-dimensional visual data.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. This work has been supported in part by the NSF Grants 15-64212 and 18-23321 and the State of Maryland's MPower initiative. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the authors and do not necessarily reflect the views of the research sponsors.

References

[1] A. Abdi, A. Payani, and F. Fekri. Learning Dictionary for Efficient Signal Compression. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3689–3693, 2017. 2

[2] A. Adams. The Stanford Light Field Archive, 2008. http://lightfield.stanford.edu/lfs.html. 5

[3] A. A. Afifi and E. A. Zanaty. Generalized Legendre Polynomials for Support Vector Machines (SVMs) Classification. *International Journal of Network Security & Its Applications*, 11:87–104, 2019. 3

[4] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006. 2, 6

[5] R. Archibald and A. Gelb. A Method to Reduce the Gibbs Ringing Artifact in MRI Scans While Keeping Tissue Boundary Integrity. *IEEE Transactions on Medical Imaging*, 21:305–319, 2002. 3

[6] M. Bemana, K. Myszkowski, H. Seidel, and T. Ritschel. X-Fields: Implicit Neural View-, Light- and Time-Image Interpolation. *ACM Trans. Graph.*, 39(6), Nov. 2020. 2

[7] M. Broxton, John Flynn, R. Overbeck, Daniel Erickson, Peter Hedman, Matthew DuVall, Jason Dourgarian, Jay Busch, Matt Whalen, and P. Debevec. Immersive light field video with a layered mesh representation. *ACM Transactions on Graphics (TOG)*, 39:86:1 – 86:15, 2020. 2

[8] J. Chai, S. Chan, H. Shum, and X. Tong. Plenoptic Sampling. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, page 307–318, 2000. 2

[9] C. Chang, X. Zhu, P. Ramanathan, and B. Girod. Light Field Compression Using Disparity-compensated Lifting and Shape Adaptation. *IEEE Transactions on Image Processing*, 15:793–806, 2006. 2

[10] G. Chaurasia, O. Sorkine, and G. Drettakis. Silhouette-Aware Warping for Image-Based Rendering. In *Computer Graphics Forum*, volume 30, pages 1223–1232. Wiley Online Library, 2011. 2

[11] A. Cohen, I. Daubechies, and J. Feauveau. Biorthogonal Bases of Compactly Supported Wavelets. *Communications on pure and applied mathematics*, 45(5):485–560, 1992. 6

[12] E. Dib, M. Le Pendu, and C. Guillemot. Light Field Compression Using Fourier Disparity Layers. *2019 IEEE International Conference on Image Processing (ICIP)*, pages 3751–3755, 2019. 2

[13] R. Du, S. Bista, and A. Varshney. Video Fields: Fusing Multiple Surveillance Videos into a Dynamic Virtual Environment. *Web3D '16*, page 165–172, 2016. 2

[14] R. Du, M. Chuang, W. Chang, H. Hoppe, and A. Varshney. Montage4D: Real-time Seamless Fusion and Stylization of Multiview Video Textures. *Journal of Computer Graphics Techniques*, 8(1), 2019. 2

[15] B. Y. Feng, W. Yao, Z. Liu, and A. Varshney. Deep Depth Estimation on 360° Images with a Double Quaternion Loss. In *2020 International Conference on 3D Vision (3DV)*, pages 524–533, 2020. 3

[16] J. Flynn, I. Neulander, J. Philbin, and N. Snavely. Deep Stereo: Learning to Predict New Views from the World's Imagery. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5515–5524, 2016. 2

[17] D. Gottlieb and C. Shu. On the Gibbs phenomenon IV: recovering exponential accuracy in a subinterval from a Gegenbauer partial sum of a piecewise analytic function. *Mathematics of Computation*, 64:1081–1095, 1995. 3

[18] D. Gottlieb and C. Shu. On the Gibbs Phenomenon and its Resolution. *SIAM Review*, 39:644–668, 1997. 3

[19] S. Gottlieb, J. Jung, and S. Kim. A Review of David Gottlieb's Work on the Resolution of the Gibbs Phenomenon. *Communications in Computational Physics*, 9:497–519, 2011. 3

[20] S. Hajisharif, E. Miandji, P. Larsson, K. Tran, and J. Unger. Light Field Video Compression and Real Time Rendering. *Computer Graphics Forum*, 38(7):265–276, 2019. 2

[21] A. Jagmohan, A. Sehgal, and N. Ahuja. Compression of lightfield rendered images using coset codes. *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, 1:830–834, 2003. 2

[22] X. Jiang, M. Le Pendu, R. A. Farrugia, and C. Guillemot. Light Field Compression With Homography-Based Low-Rank Approximation. *IEEE Journal of Selected Topics in Signal Processing*, 11:1132–1145, 2017. 2

[23] J. Jin, J. Hou, H. Yuan, and S. Kwong. Learning Light Field Angular Super-Resolution via a Geometry-Aware Network. In *Proceedings of the AAAI Conference on Artificial Intelligence 2020*, 2020. 7, 13

[24] N. K. Kalantari, T. Wang, and R. Ramamoorthi. Learning-Based View Synthesis for Light Field Cameras. *ACM Transactions on Graphics (TOG)*, 35:1 – 10, 2016. 2

[25] A. Karakottas, N. Zioulis, S. Samaras, D. Ataloglou, V. Gkitsas, D. Zarpalas, and P. Daras. 360° Surface Regression with a Hyper-Sphere Loss. In *2019 International Conference on 3D Vision (3DV)*, pages 258–268, 2019. 3

[26] M. Magnor and B. Girod. Data compression for light-field rendering. *IEEE Trans. Circuits Syst. Video Technol.*, 10:338–343, 2000. 2

[27] J. Mairal, F. R. Bach, J. Ponce, and G. Sapiro. Online Dictionary Learning for Sparse Coding. In *Proceedings of the 26th annual International Conference on Machine Learning (ICML '09)*, pages 689–696, 2009. 2

[28] X. Meng, R. Du, J. F. JaJa, and A. Varshney. 3D-Kernel Foveated Rendering for Light Fields. *IEEE Transactions on Visualization and Computer Graphics*, 27(8):3350–3360, 2021. 2

[29] E. Miandji, S. Hajisharif, and J. Unger. A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos. In *ACM Transactions on Graphics (TOG)*, volume 38, pages 1 – 18, 2019. 2, 5, 6

[30] E. Miandji, J. Kronander, and J. Unger. Learning-Based Compression of Surface Light Fields for Real-Time Rendering of Global Illumination Scenes. In *SIGGRAPH Asia 2013 Technical Briefs*, SA '13, 2013. 2

[31] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Proceedings of*

the European Conference on Computer Vision (ECCV 2020), pages 405–421, 2020. 1, 2

[32] D. Mukherjee, J. Bankoski, A. Grange, J. Han, J. Koleszar, P. Wilkins, Y. Xu, and R. Bultje. The Latest Open-source Video Codec VP9 - An Overview and Preliminary Results. *2013 Picture Coding Symposium (PCS)*, pages 390–393, 2013. 2

[33] R. S. Overbeck, D. Erickson, D. Evangelakos, M. Pharr, and P. Debevec. A System for Acquiring, Processing, and Rendering Panoramic Light Field Stills for Virtual Reality. *ACM Transactions on Graphics (TOG)*, 37:1 – 15, 2018. 2

[34] L. C. Padierna, M. Carpio, A. R. Domínguez, H. J. P. Soberanes, and H. J. Fraire. A Novel Formulation of Orthogonal Polynomial Kernel Functions for SVM Classifiers: The Gegenbauer Family. *Pattern Recognition*, 84:211–225, 2018. 3

[35] M. Le Pendu, C. Guillemot, and A. Smolic. A Fourier Disparity Layer Representation for Light Fields. *IEEE Transactions on Image Processing*, 28:5740–5753, 2019. 2

[36] M. Le Pendu, C. Ozcinar, and A. Smolic. Hierarchical Fourier Disparity Layer Transmission For Light Field Streaming. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 2606–2610. IEEE, 2020. 2

[37] M. Le Pendu and A. Smolic. High Resolution Light Field Recovery with Fourier Disparity Layer Completion, Demosaicing, and Super-Resolution. *2020 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12, 2020. 2

[38] S. Pratapa and D. Manocha. RLFC: Random Access Light Field Compression using Key Views and Bounded Integer Sequence Encoding. *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2019. 2

[39] N. Rahaman, A. Baratin, D. Arpit, F. Dräxler, M. Lin, F. Hamprecht, Y. Bengio, and A. C. Courville. On the Spectral Bias of Neural Networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 4

[40] N. Sabater, G. Boisson, B. Vandame, P. Kerbiriou, F. Babon, M. Hog, R. Gendrot, T. Langlois, O. Bureller, A. Schubert, and V. Allie. Dataset and Pipeline for Multi-view Light-Field Video. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1743–1753, 2017. 5

[41] H. Shum, S. Chan, and S. Kang. *Image-based rendering*. Springer Science & Business Media, 2008. 2

[42] V. Sitzmann, E. Chan, R. Tucker, N. Snavely, and G. Wetzstein. MetaSDF: Meta-learning Signed Distance Functions. In *Advances in Neural Information Processing Systems*, NeurIPS 2020, 2020. 8

[43] V. Sitzmann, J. N. P. Martel, A. Bergman, D. B. Lindell, and G. Wetzstein. Implicit Neural Representations with Periodic Activation Functions. In *Advances in Neural Information Processing Systems*, NeurIPS 2020, 2020. 1, 2, 4, 5, 11

[44] J. Sulam, V. Papyan, Y. Romano, and M. Elad. Multilayer Convolutional Sparse Modeling: Pursuit and Dictionary Learning. *IEEE Transactions on Signal Processing*, 66:4090–4104, 2018. 2

[45] G. Sullivan, J. Ohm, W. Han, and T. Wiegand. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22:1649–1668, 2012. 2

[46] M. Tancik, B. Mildenhall, T. Wang, D. Schmidt, P. P. Srinivasan, J. Barron, and R. Ng. Learned Initializations for Optimizing Coordinate-Based Neural Representations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, pages 2846–2855, 2021. 8

[47] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. In *Advances in Neural Information Processing Systems*, NeurIPS 2020, 2020. 1, 2, 4, 5

[48] I. Viola, M. Řeřábek, and T. Ebrahimi. Comparison and Evaluation of Light Field Image Coding Approaches. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):1092–1106, 2017. 1

[49] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light Field Image Processing: An Overview. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):926–954, 2017. 1

[50] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light Field Reconstruction Using Deep Convolutional Network on EPI. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1638–1646, 2017. 2

[51] H. W. F. Yeung, J. Hou, J. Chen, Y. Y. 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)*, 2018. 2

[52] X. Zhu, A. Aaron, and B. Girod. Distributed Compression for Large Camera Arrays. *IEEE Workshop on Statistical Signal Processing, 2003*, pages 30–33, 2003. 2