Acquiring a Dynamic Light Field through a Single-Shot Coded Image

Ryoya Mizuno[†], Keita Takahashi[†], Michitaka Yoshida[‡], Chihiro Tsutake[†], Toshiaki Fujii[†], Hajime Nagahara[‡]
[†]Nagoya University, Japan, [‡]Osaka University, Japan

Abstract

We propose a method for compressively acquiring a dynamic light field (a 5-D volume) through a single-shot coded image (a 2-D measurement). We designed an imaging model that synchronously applies aperture coding and pixel-wise exposure coding within a single exposure time. This coding scheme enables us to effectively embed the original information into a single observed image. The observed image is then fed to a convolutional neural network (CNN) for light-field reconstruction, which is jointly trained with the camera-side coding patterns. We also developed a hardware prototype to capture a real 3-D scene moving over time. We succeeded in acquiring a dynamic light field with 5×5 viewpoints over 4 temporal sub-frames (100 views in total) from a single observed image. Repeating capture and reconstruction processes over time, we can acquire a dynamic light field at 4× the frame rate of the camera. To our knowledge, our method is the first to achieve a finer temporal resolution than the camera itself in compressive light-field acquisition. Our software is available from our project webpage.[1]

1. Introduction

A light field is represented as a set of multi-view images, where dozens of views are aligned on a 2-D grid with tiny viewpoint intervals. This representation contains rich visual information of a target scene and thus can be used for various applications such as 3-D display [14, 38], view synthesis [20, 58], depth estimation [34, 51], synthetic refocusing [13, 25], and object recognition [17, 45]. The scope of applications will further expand if the target scene is able to move over time. However, a light field varying over time, i.e., a dynamic light field, is challenging to acquire due to the huge data rate, which is proportional to both the number of views and frame rate.

Several approaches to acquire light fields have been investigated as summarized in Fig. 1. The most straightforward approach is to construct an array of cameras [5, 37, 49], which requires bulky and costly hardware. The second approach is to insert a micro-lens array in front of an image sensor [1, 2, 24, 25, 29, 46], which enables us to capture a light field in a single-shot image. However, the spatial resolution of each viewpoint image is sacrificed for the angular resolution (number of views). In the above two approaches, the frame rate of the acquired light field is at most equivalent to that of the cameras. Moreover, the data rate is not compressed because each light ray is sampled individually.

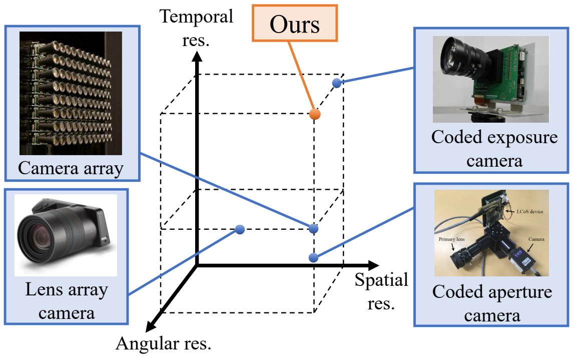


Figure 1. Our achievement compared with representative previous works (camera array [49], lens-array camera [24], coded-aperture camera [12], and coded exposure camera [54]). Axes are in relative scales w.r.t. camera's spatial resolution and frame rate.

The third approach aims to acquire a light field compressively by using a single camera equipped with a coded mask or aperture [3, 6, 7, 12, 16, 18, 22, 23, 39, 41, 43]. This kind of camera was used to obtain a small number of coded images, from which a light field with the full-sensor spatial resolution can be reconstructed. For static scenes, taking more images with different coding patterns is beneficial to achieve higher reconstruction quality. However, for moving scenes, the use of multiple coded images involves additional complexities related to scene motions. Hajisharif et al. [8] used a high dimensional light-field dictionary that spanned several temporal frames. However, their dictionary-based light-field reconstruction required a prohibitively long computation time. Sakai et al. [31] handled scene motions by alternating two coding patterns over time and by training their CNN-based algorithm on dynamic scenes. However, the light field was reconstructed only for every two temporal

[1] https://www.fujii.nuee.nagoya-u.ac.jp/Research/CompCam2

frames (at 0.5× the frame rate of the camera).

In this paper, we advance the compressive approach several steps further to innovate the imaging method for a dynamic light field. As shown in Fig. 1, our method pursues the full-sensor spatial resolution and a faster frame rate than the camera itself. To this end, we design an imaging model that synchronously applies aperture coding [12, 16, 23] and pixel-wise exposure coding [9,30,48,54] *within* a single exposure time. This coding scheme enables us to effectively embed the original information (a 5-D volume of a dynamic light field) into a single coded image (a 2-D measurement). The coded image is then fed to a CNN for light-field reconstruction, which is jointly trained with the camera-side coding patterns. We also develop a hardware prototype to capture real 3-D scenes moving over time. As a result, we succeeded in acquiring the dynamic light field with 5×5 viewpoints over 4 temporal sub-frames (100 views in total) from a single coded image. Repeating capture and reconstruction processes over time, we acquired a dynamic light field at $4\times$ the frame rate of the camera. To our knowledge, our method is the first to achieve a finer temporal resolution than the camera itself in compressive light-field acquisition.

2. Background

2.1. Computational Photography

In the literature of computational photography, aperture coding has been used to encode the viewpoint (angular) dimension of a light field [6, 12, 16, 23], while exposure coding has been adopted to encode fast temporal changes in a monocular video [9, 28, 30, 48, 54]. Our method combines them to encode both the viewpoint (angular) and temporal dimensions simultaneously. Our method is also considered as an extreme case of snapshot compressive imaging [44,56,57], where a higher dimensional (typically 3-D) data volume is compressed into a 2-D sensor measurement.

We noticed that Vargas et al. [42] recently proposed an imaging architecture similar to ours for compressive light field acquisition. However, their method was designed for static light fields. Accordingly, their image formation model implicitly assumed that the target light field should be invariant during an exposure time (during the period when the time-varying coding patterns were applied), which is theoretically incompatible with moving scenes. Moreover, they did not report hardware implementation for the pixel-wise exposure coding. In contrast, our method is designed to handle motions during each exposure time, and it is fully implemented as a hardware prototype.

We model the entire imaging pipeline (coded-image acquisition and light-field reconstruction) as a deep neural network, and jointly optimize the camera-side coding patterns and the reconstruction algorithm. This design aligns with the recent trend of deep optics [4,11,12,15,26,31,36,52,54] where optical elements and computational algorithms are jointly optimized under the framework of deep learning. However, our method is designed to handle higher dimensional data (dynamic light fields) than the previous works.

2.2. Light-Field Reconstruction

Reconstruction of a light field from a coded/compressed measurement is considered as an inverse problem, for which several classes of methods can be used. Traditional methods [3, 18, 19] formulated this problem as energy minimization with rather simple explicitly-defined prior terms and solved them using iterative algorithms. These methods often result in insufficient reconstruction quality and long computation time. Recently, deep-learning-based methods [7, 12, 22, 41, 47, 53] have gained more popularity due to the excellent representation capability of data-driven implicit priors. Trained on a suitable dataset, these methods can acquire the capability of high-quality reconstruction. Moreover, reconstruction (inference) on a pre-trained network does not require much computation time. Hybrid approaches have also been investigated. Algorithm unrolling methods [6, 21] unroll procedures of iterative algorithms into trainable networks, whereas plug-and-play methods [56, 57] use pre-trained network models as building blocks of iterative algorithms.

We take a deep-learning-based approach and jointly optimize the entire process (coded-image acquisition and light-field reconstruction) in the spirit of deep optics. For the reconstruction part, we use a rather plain network architecture to balance the reconstruction quality and the computational efficiency. Further improvement would be expected with more sophisticated and light-field specific network architectures [6, 53]. We leave this as future work, because the main focus of this paper is the design of the image acquisition process rather than the reconstruction network.

In recent years, view synthesis from a single image [10, 27, 33, 35, 40, 50] has attracted much attention. In principle, 3-D reconstruction/rendering from an ordinary monocular image (without coding) is an ill-posed problem; the results are *hallucinated* by using the implicit scene priors learned from the training dataset rather than the physical cues. In contrast, our method aims to *recover* the 3D and motion information that is *embedded* into a single image through the camera-side coding process.

3. Proposed Method

3.1. Notations and Problem Formulation

A schematic diagram of the camera we assume is shown in Fig. 2. Each light ray coming into the camera is parameterized with five variables, (u, v, x, y, t), where (u, v) and (x, y) denote the intersections with the aperture and imaging planes, respectively, and t denotes the time *within* a sin-

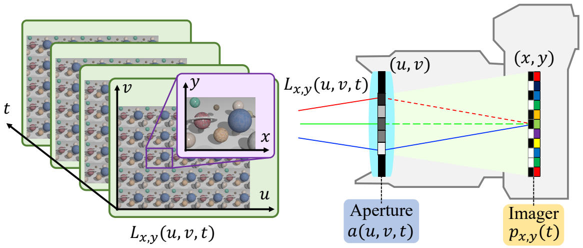


Figure 2. Example of dynamic light field (left) and schematic diagram of camera (right).

gle exposure time of the camera. We discretize the variable space into a 5-D integer grid, where the range of each variable is described as $S_\xi = [0, N_\xi)$ $(\xi \in \{x, y, u, v, t\})$. By using these variables, the intensity of a light ray is described as $L_{x,y}(u, v, t)$[2]. Since (u, v) is associated with the viewpoint (angle), $L_{x,y}(u, v, t)$ is equivalent to a set of multi-view videos, i.e., a dynamic light field.

Our aim is to acquire the latent dynamic light field $L_{x,y}(u, v, t)$: a 5-D volume with $N_x N_y N_u N_v N_t$ unknowns, from a single coded image $I_{x,y}$: a 2-D measurement with $N_x N_y$ observables. Hereafter, we assume $N_u = N_v = 5$ and $N_t = 4$ unless mentioned otherwise.

3.2. Image Acquisition Model

If the camera has no coding functionalities (in the case of an ordinary camera), the observed image is given by

$$I_{x,y} = \sum_{(u,v,t)\in S_u \times S_v \times S_t} L_{x,y}(u, v, t). \tag{1}$$

Each pixel value, $I_{x,y}$, is the sum of light rays over the viewpoint (u, v) and temporal (t) dimensions. Therefore, the variation along u, v, t dimensions is simply blurred out, making it difficult to recover.

Meanwhile, we design an imaging method that can effectively preserve the original 5-D information. We exploit the combination of aperture coding and pixel-wise exposure coding that are synchronously varied *within* a single exposure time. The observed image is given as

$$I_{x,y} = \sum_{(u,v,t)\in S_u \times S_v \times S_t} a(u, v, t)\, p_{x,y}(t)\, L_{x,y}(u, v, t). \tag{2}$$

where $a(u, v, t) \in [0, 1]$ (semi-transparency) and $p_{x,y}(t) \in \{0, 1\}$ (on/off) are coding patterns applied on the aperture and pixel planes, respectively. This imaging process can be regarded as two-step coding as follows. First, a series of aperture coding patterns, $a(u, v, t)$, is applied to $L_{x,y}(u, v, t)$ over time, which reduces the original 5-D volume into a 3-D spatio-temporal tensor, $J_{x,y}(t)$, as

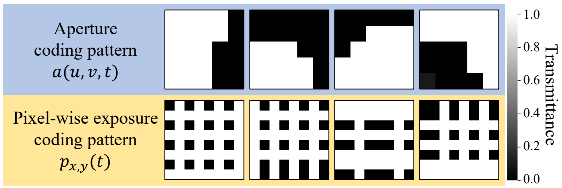

Aperture coding pattern $a(u,v,t)$
Pixel-wise exposure coding pattern $p_{x,y}(t)$
1.0
0.8
0.6
0.4
0.2
0.0
Transmittance

Figure 3. Coding patterns applied on aperture and pixel planes.

$$J_{x,y}(t) = \sum_{(u,v)\in S_u \times S_v} a(u, v, t)\, L_{x,y}(u, v, t). \tag{3}$$

Next, the 3-D tensor, $J_{x,y}(t)$, is further reduced into a 2-D measurement, $I_{x,y}$, through the pixel-wise exposure coding over time using $p_{x,y}(t)$, as

$$I_{x,y} = \sum_{t\in S_t} p_{x,y}(t)\, J_{x,y}(t). \tag{4}$$

By combining these two steps, we encode both the viewpoint (u, v) and temporal (t) dimensions and embed them into a single 2-D image.

An example of the coding patterns is shown in Fig. 3. As mentioned later, these patterns are directly linked with the parameters of a CNN (AcqNet), which is jointly trained with another CNN for light-field reconstruction (RecNet). Therefore, these coding patterns are optimized for the training dataset so as to preserve as much of the light-field information as possible in the observed image.

Figure 4 shows two images (close-ups of the same portion) obtained from a test scene through two imaging models: the ordinary camera (Eq. (1)) and ours (Eq. (2)). The ordinary camera obtains a simply blurred observation, while ours obtains a dappled image due to the coding patterns. To further analyze the effect of coding, we also used a primitive scene with a fronto-parallel plane (a primitive plane scene). As shown in Fig. 5, we prepared an image $G(x, y)$ with nine bright points as the texture for the plane. We then synthesized a dynamic light field using the parameters for the 2-D lateral velocity (α_x, α_y) [pixels per unit time] and disparity d [pixels per viewpoint] (corresponding to the depth) as

$$L_{x,y}(u, v, t) = G(x - du - \alpha_x t, y - dv - \alpha_y t) \tag{5}$$

from which we computed an observed image by using Eq. (2). Some resulting images obtained with different parameters are shown in Fig. 5 (the brightness is corrected for visualization). These images can be interpreted as point spreading functions (PSFs) for various motion and disparity values. Notably, these PSFs are distinct from each other. Moreover, even in a single image, the PSFs for the nine

[2] For simplicity, we assume that a light field has a single color channel. When handling a light field with RGB colors, we treat each color channel as an individual light field.

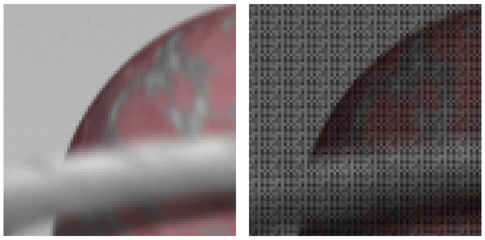


Figure 4. Example images acquired by ordinary camera Eq. (1) (left) and our imaging model of Eq. (2) (right).

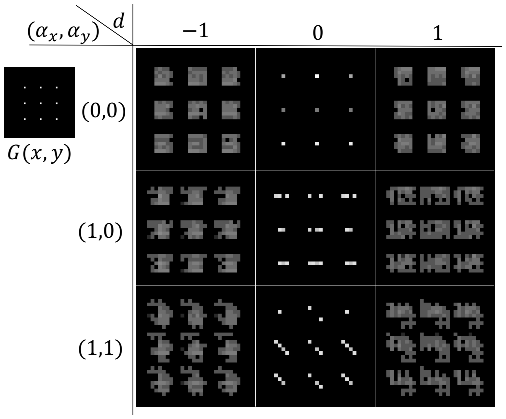


Figure 5. Our imaging model yields distinct PSFs for different motion and disparity values (coding patterns in Fig. 3 were used).

points differ from each other. These results show that both motions and disparities, which are associated with changes along the temporal (t) and viewpoint (u, v) dimensions, respectively, are *encoded* by the various shapes of PSFs depending on the spatial coordinate (x, y). The encoded information is not human readable, but can be deciphered by the RecNet that is jointly trained with the coding patterns.

3.3. Hardware Implementation

We developed a prototype camera shown in Fig. 6 that can apply aperture coding and pixel-wise exposure coding within a single exposure time.

We used a Nikon Rayfact (25 mm F1.4 SF2514MC) as the primary lens. The aperture coding was implemented using a liquid crystal on silicon (LCoS) display (Forth Dimension Displays, SXGA-3DM), which had 1280 × 1024 pixels. We divided the central area of the LCoS display into 5 × 5 regions, each with 150 × 150 pixels. Accordingly, the angular resolution of the light field was set to 5 × 5. The pixel-wise exposure coding was implemented using a row-column-wise exposure sensor [54] that had 656 × 512 pixels. We synchronized the LCoS display with the image sensor via an external circuit, so that four sets of coding patterns were synchronously applied within a single exposure time. The timing chart is shown in Fig. 7. The time duration assigned for each coding pattern was set to 17 ms.

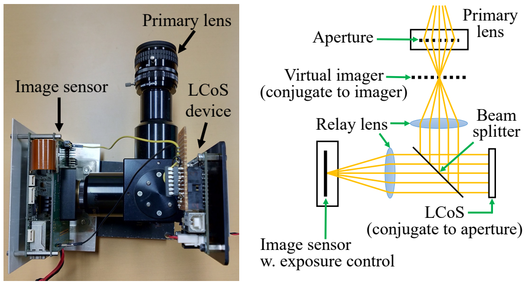

Primary lens
Image sensor
LCoS device
Aperture
Primary lens
Virtual imager (conjugate to imager)
Relay lens
Beam splitter
LCoS (conjugate to aperture)
Image sensor w. exposure control

Figure 6. Our camera prototype (left) and optical diagram (right).

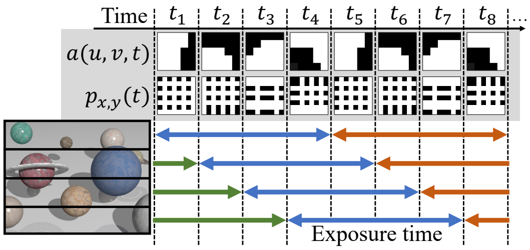


Figure 7. Time chart of our camera. Exposure timing is different for four vertically divided regions on image sensor.

Accordingly, the unit time for the target light field was also 17 ms (58.8 fps). Meanwhile, a single exposure time of the camera ranged over the 4 time units (temporal sub-frames), and thus, the interval between the two exposed images was 68 ms (14.7 fps in terms of the camera's frame rate).

We mention several restrictions resulting from the image sensor's hardware. First, the sensor was not equipped with RGB filters and was thus incapable of obtaining color information. Second, the coding patterns were not freely designable, because they were generated by the column-wise and row-wise control signals repeating for every 8×8 pixels. Therefore, the applicable coding patterns were limited to binary, 8×8-pixels periodic, and row-column separable ones. This restriction was considered in our network design as mentioned later. Finally, due to the timing of the vertical scan, the time duration covered by a single exposed image depended on the vertical position. More precisely, as shown in Fig. 7, the image sensor was vertically divided into 4 regions, each of which had a distinctive exposure timing with 17 ms differences from the neighbors. Accordingly, these regions were modulated by the same four sets of coding patterns but in different orders. To accommodate these differences, we used a single instance for AcqNet, but permuted the order of time units in the input light field for the 4 regions, respectively. We prepared 4 instances of RecNet corresponding to the 4 regions and jointly trained them with the coding patterns. This extension required four region-wise reconstruction processes conducted in parallel, but still maintained ×4 finer temporal resolution than the camera.

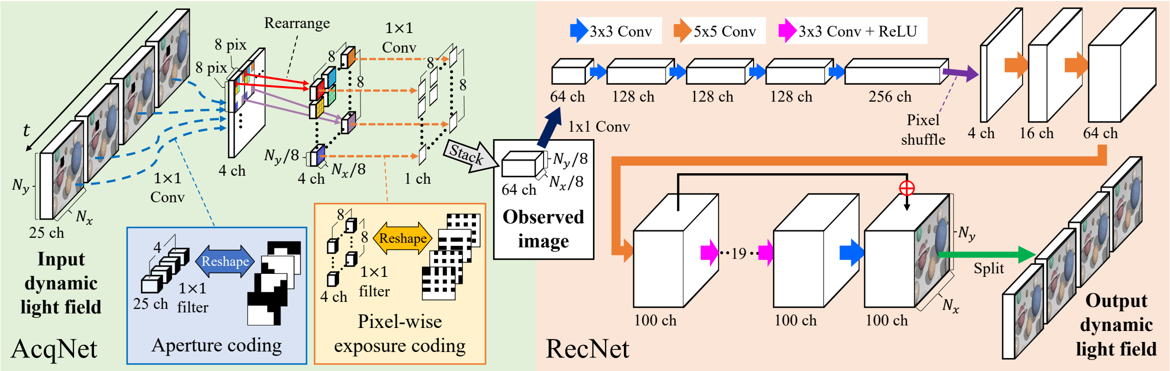


Figure 8. Our network architecture consists of AcqNet and RecNet, which correspond to coded image acquisition and light-field reconstruction processes, respectively. Dynamic light-field ranging over four temporal units is processed at once.

3.4. Network Design and Training

As shown in Fig. 8, our method was implemented as a fully convolutional network, consisting of AcqNet and RecNet. AcqNet is a differentiable representation of the image formation model with trainable coding patterns, where a target light field is compressed into a single observed image. RecNet was designed to receive the observed image as input and reconstruct the original light field. The entire network was trained end-to-end using the squared error against the ground-truth light field as the loss function. By doing so, the image acquisition and light-field reconstruction processes were jointly optimized. When a real camera was used, the coding patterns for the camera were tuned in accordance with the trained parameters of AcqNet. Then, image acquisition was conducted physically on the imaging hardware, and only the reconstruction (inference on RecNet) was performed on the computer.

AcqNet takes as input a dynamic light field over 4 consecutive time units, which has $N_x \times N_y$ pixels and 5×5 viewpoints over 4 time units. The viewpoint dimensions are unfolded into a single channel, resulting in 4 input tensors with the shape of $25 \times N_x \times N_y$. The first block of AcqNet corresponds to the aperture coding (Eq. (3)). To implement this process, we followed Inagaki et al. [12]; we used 2-D convolutional layers with 1×1 kernels and no biases, where each kernel weight corresponds to the apertures' transmittance for each viewpoint. We prepared 4 separate convolutional layers for the 4 time units, in each of which 25 channels were reduced into a single channel. The outputs from these layers are stacked along the channel dimension, resulting in a tensor of $4 \times N_x \times N_y$. The second block corresponds to the pixel-wise exposure coding (Eq. (4)), where 8×8 repetitive patterns are applied. For this process, we prepared 64 separate convolutional layers (1×1 kernels without biases), each of which takes a tensor of $4 \times N_x/8 \times N_y/8$ as input (every 8×8 pixels extracted from the tensor of $4 \times N_x \times N_y$) and reduces 4 channels into a single channel. To constrain the coding patterns to be hardware implementable (binary and row-column separable), we used the same training technique as Yoshida et al. [55] (see section 4.1 in [55]). The outputs from these layers are stacked along the channel dimension, resulting in a tensor of $64 \times N_x/8 \times N_y/8$, which is equivalent to a single observed image with $N_x \times N_y$ pixels. Finally, to account for noise during the acquisition process, Gaussian noise (zero-mean and $\sigma = 0.005$ w.r.t. the range of pixel values $[0, 1]$) is added to the observed image.

RecNet accepts an output from AcqNet (or an image acquired from a real camera) as a tensor of $64 \times N_x/8 \times N_y/8$. The first 5 convolutional layers gradually increase the number of channels to 256, while keeping the spatial size unchanged. Then, the tensor is reshaped into $4 \times N_x \times N_y$ using a pixel shuffling operation [32]. The subsequent two convolutional layers increase the number of channels to 100, followed by 19 convolutional layers and a residual connection for refinement. The output from RecNet is the latent dynamic light field represented as a tensor of $100 \times N_x \times N_y$, where 100 channels correspond to 5×5 views over 4 time units (temporal sub-frames). As mentioned in 3.3, four instances of RecNet should be used in parallel to handle the time differences among the four vertical regions.

We finally mention the training dataset. We first collected 223,020 light-field patches from 51 static light fields with intensity augmentation. Next, following Sakai et al. [31], we gave 2-D lateral motions (in-plane translations) to the collected patches to synthesize *virtually-moving* light-field samples. We used linear motions with constant velocities: (α_x, α_y) [pixels per unit time], where $\alpha_x, \alpha_y \in \{-2, 1, 0, 1, 2\}$; this is equivalent to at most ±8 pixel translation per frame in terms of the camera's frame rate. This motion model was simple and limited, but it would be sufficient for the motions *within* a single exposure time, which is short enough. We had 25 motion patterns in total, all of which were applied to each light-field patch. To sum up, we had 5,575,500 samples of dynamic light fields, each with 64×64 pixels at 5×5 viewpoints over 4 time units. Note that even a single training sample had a significant size (409,600 elements), which necessitated the network to be lightweight.

We implemented our software using PyTorch. The network was trained over five epochs using the Adam optimizer. The training took approximately seven days on a PC equipped with NVIDIA Geforce RTX 3090. We also trained our model with 8×8 views and different ranges for the assumed motions (α_x, α_y). Please refer to the supplementary material for details.

4. Experiments

We conducted several quantitative evaluations using a computer generated scene and experiments using our prototype camera. To summarize, we succeeded in acquiring a dynamic light field with 4× finer temporal resolution than the camera itself. Note that there is no baseline to compete against, because to our knowledge, no prior works have ever achieved the same goal as ours. Please refer to the supplementary video for better visualization of our results.

4.1. Quantitative Evaluation

Ablation study for the coding method. To validate our image acquisition model in Eq. (2), we need to analyze the effect of coding on the aperture ($a(u, v, t)$) and pixel ($p_{x,y}(t)$) planes. In addition to our original method (denoted as **A+P**), we trained three variants of our methods as follows. **Ordinary**: no coding was applied ($a(u, v, t) =$ const, $p_{x,y}(t) =$ const), which corresponded to light-field reconstruction from a single uncoded image. **A-only**: only the aperture coding was enabled ($p_{x,y}(t) =$ const). **P-only**: only the pixel-wise exposure coding was enabled ($a(u, v, t) =$ const). Furthermore, to evaluate the theoretical upper-bound, we also prepared a free-form coding over the 5-D space (denoted as **Free5D**), given by:

$$I_{x,y} = \sum_{(u,v,t) \in S_u \times S_v \times S_t} m(x, y, u, v, t) L_{x,y}(u, v, t) \qquad (6)$$

where $m(x, y, u, v, t) \in [0, 1]$ was a fully trainable modulating pattern periodic over 8×8 pixels. Note that this is only a software simulation; no hardware realization is available. The five methods mentioned so far were different in the imaging models but aimed for the same goal: reconstructing a dynamic light field (5×5 views over 4 time units) from a single observed image. For all the methods, RecNets with the same network structure were jointly trained with the respective coding patterns on the same training dataset for the same number of epochs.

For quantitative evaluation, we used a computer generated light field with 5×5 viewpoints over 200 temporal frames, which was rendered from *Planets* scene provided by Sakai et al. [31]. [3] Figure 9 visualizes several reconstructed views (at the top-left viewpoint), horizontal epipolar plane images (EPIs) along the green lines, and the differences from the ground truth (×3 pixel values). The average peak signal-to-noise ratio (PSNR) values over the 25 viewpoints are plotted along the temporal frames in Fig. 10.

As observed from these results, our method clearly outperformed the other variants and even achieved quality close to the ideal Free5D case. Meanwhile, A-only and P-only resulted in poor reconstruction quality, showing their insufficiency as coding methods. Moreover, the poor result from Ordinary case indicated that although implicit scene priors were learned from the training dataset, they alone were insufficient for high-quality reconstruction. In contrast, the success of our method can be attributed to the elaborated coding method that was simultaneously applied on the aperture and imaging planes, which helped effectively embed the original 5-D information into a single observed image. However, the reconstruction quality of our method exhibited small fluctuations over time. This was closely related to the fact that four time units (temporal frames) were processed as a group. Moreover, our method did not include mechanisms that could explicitly encourage the temporal consistency, which will be addressed in the future work.

Working range analysis. We also evaluated the effective working range against motion and disparity using a primitive plane scene. Following Eq. (5), we synthesized a dynamic light field over four time units by using a natural image in Fig. 11 (left) as the texture. The average PSNR values obtained with our method (A+P) and the three variants (A-only, P-only, and Ordinary) are shown in Fig. 11 (right). Obviously, our method (A+P) can cover a wider range of motion/disparity values than the others; P-only performed poorly for $d \neq 0$; A-only and Ordinary did not work well except for $d = \alpha_x = 0$.

In our method (A+P), the reconstruction quality degraded gradually as the velocity and disparity values increased. This means that large motions/disparities are challenging for our method. The working range for the disparity was mainly determined by the 3-D scene structures contained in the original light-field dataset, while the working range for the velocity was related to the virtual motions we assumed when synthesizing the dynamic dataset from static light fields. Note that our imaging system has densely-located viewpoints (bounded by the aperture) and a high temporal resolution (4× the frame-rate of the camera); therefore, both the motion and disparity are usually limited within a small range.

Comparison with other methods. We finally compared our method against three other methods. The first two methods [6, 31] were based on coded-aperture imaging. From

[3] https://www.fujii.nuee.nagoya-u.ac.jp/Research/CompCam/

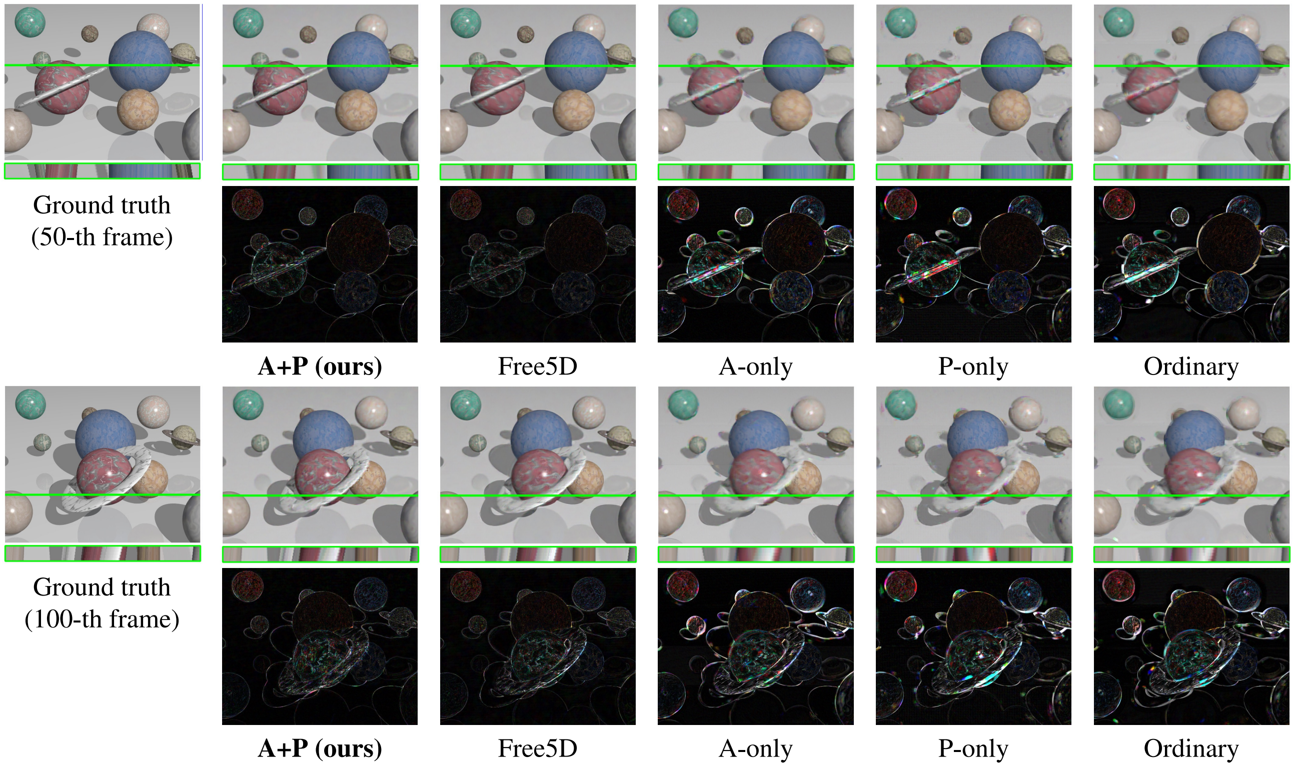


Figure 9. Visual results of our method (A+P), Free5D (ideal case), and three ablation cases (A-only, P-only, and Ordinary). Reconstructed top-left views are accompanied with horizontal EPIs along green lines and differences from ground truth (×3 brightness).

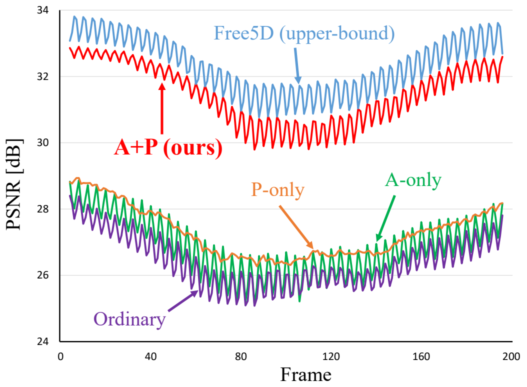


Figure 10. Quantitative reconstruction quality over time for our method (A+P), Free5D (ideal case), and three ablation cases (A-only, P-only, and Ordinary).

Guo et al. [6], we adopted a model where a light field for each time unit was reconstructed from a single observed image, which resulted in frame-by-frame observation and light-field reconstruction at the same frame rate as the camera. The method of Sakai et al. [31] observed three consecutive images over time, and reconstructed a light field for the central time. The light field was reconstructed for every two frames (at $0.5\times$ the frame rate) of the camera. We retrained Guo et al.'s and Sakai et al.'s on the same dataset as ours until convergence. In addition, we simulated a Lytro-like camera, where each of the 5×5 views was captured with the $1/5 \times 1/5$ spatial resolution at the same frame rate as the camera. The acquired 5×5 views were upsampled to the original resolution using bicubic interpolation and compared against the ground truth.

For quantitative evaluation, we used *Planets* assuming the camera's frame rate to be the same as ours; accordingly, in these three methods, image acquisition was conducted only at every four temporal frames. Note that only our method can obtain the light field at $4\times$ the frame rate of the camera, and thus, this comparison only serves as a reference. The average PSNR values over time are shown in Fig. 12. The method of Sakai et al. [31] failed to follow the fast scene motions, resulting in poor reconstruction quality. The method of Guo et al. [6] reconstructed a finely textured but geometrically inconsistent result, whereas the Lytro-like camera produced a geometrically consistent but blurred result. Our method achieved the best reconstruction quality with $\times 4$ finer temporal resolution than the camera.

Please refer to the supplementary material for more detailed analysis with different training conditions.

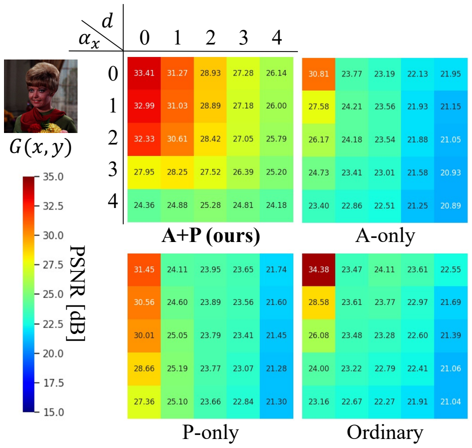


Figure 11. Performance evaluation against various motion and disparity values on primitive plane scene.

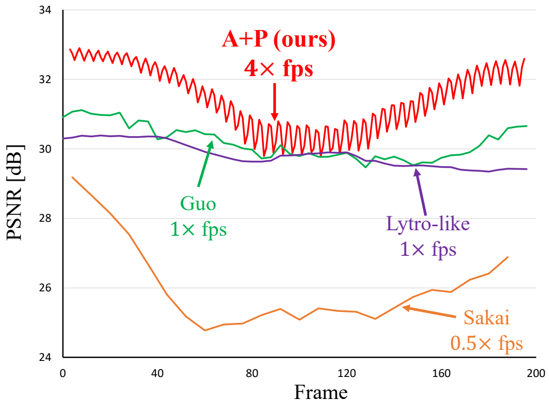
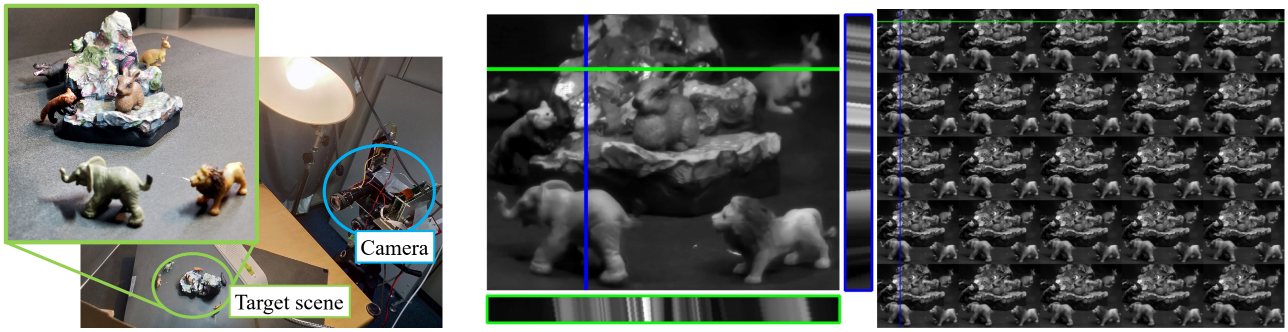


Figure 12. Quantitative quality over time compared against other methods (Guo et al. [6], Sakai et al. [31], and Lytro-like camera).



Experimental setup

Reconstructed light field

Figure 13. Experiment using our prototype camera: experimental setup (left) and reconstructed top-left view accompanied by two EPIs along green and blue lines (center), and reconstructed light field with 5 × 5 views (right).

4.2. Experiment Using Camera Prototype

We prepared a target scene by using several objects (miniature animals) placed on an electronic turntable, which produced motions in various directions. Our prototype camera was used to capture the scene at 14.7 fps, from which we reconstructed the dynamic light field at 58.8 fps (4 temporal frames from each exposed image). The reconstructed light field had 5 × 5 views, each with the full-sensor resolution (656 × 512 pixels) for each time unit. Our experimental setup and a part of the results are shown in Fig. 13. The reconstructed light field exhibited natural motions over time and consistent parallaxes among the viewpoints (refer to the supplementary video).

5. Conclusions

We proposed a method for compressively acquiring a dynamic light field (a 5-D volume) through a single-shot coded image (a 2-D measurement). Our method was embodied as a camera that synchronously applied aperture coding and pixel-wise exposure coding within a single exposure time, combined with a deep-learning-based algorithm for light-field reconstruction. The coding patterns were jointly optimized with the reconstruction algorithm, so as to embed as much of the original information as possible in a single observed image. Experimental results showed that by using a single camera alone, our method can successfully acquire a dynamic light field with 5 × 5 views at 4× the frame rate of the camera. We believe this is a significant advance in the context of compressive light-field acquisition, which will motivate the computational photography community to investigate further. Our future work will include improvement on the network design for better reconstruction quality and generalization to different configurations concerning the number of views and the number of time units included in a single exposure time.

Acknowledgement: Special thanks go to Yukinobu Sugiyama and Kenta Endo at Hamamatsu Photonics K.K. for providing the image sensor.

References

[1] Edward H Adelson and John YA Wang. Single lens stereo with a plenoptic camera. *IEEE transactions on pattern analysis and machine intelligence*, 14(2):99–106, 1992. 1

[2] Jun Arai, Fumio Okano, Haruo Hoshino, and Ichiro Yuyama. Gradient-index lens-array method based on real-time integral photography for three-dimensional images. *Applied optics*, 37(11):2034–2045, 1998. 1

[3] S. Derin Babacan, Reto Ansorge, Martin Luessi, Pablo Ruiz Mataran, Rafael Molina, and Aggelos K Katsaggelos. Compressive light field sensing. *IEEE Transactions on image processing*, 21(12):4746–4757, 2012. 1, 2

[4] Ayan Chakrabarti. Learning sensor multiplexing design through back-propagation. In *International Conference on Neural Information Processing Systems*, pages 3089–3097, 2016. 2

[5] Toshiaki Fujii, Kensaku Mori, Kazuya Takeda, Kenji Mase, Masayuki Tanimoto, and Yasuhito Suenaga. Multipoint measuring system for video and sound - 100-camera and microphone system. In *IEEE International Conference on Multimedia and Expo*, pages 437–440, 2006. 1

[6] Mantang Guo, Junhui Hou, Jing Jin, Jie Chen, and Lap-Pui Chau. Deep spatial-angular regularization for compressive light field reconstruction over coded apertures. In *European Conference on Computer Vision*, pages 278–294, 2020. 1, 2, 6, 7, 8

[7] Mayank Gupta, Arjun Jauhari, Kuldeep Kulkarni, Suren Jayasuriya, Alyosha Molnar, and Pavan Turaga. Compressive light field reconstructions using deep learning. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1277–1286, 2017. 1, 2

[8] Saghi Hajisharif, Ehsan Miandji, Christine Guillemot, and Jonas Unger. Single sensor compressive light field video camera. *Computer Graphics Forum*, 39(2):463–474, 2020. 1

[9] Yasunobu Hitomi, Jinwei Gu, Mohit Gupta, Tomoo Mitsunaga, and Shree K. Nayar. Video from a single coded exposure photograph using a learned over-complete dictionary. In *International Conference on Computer Vision*, pages 287–294, 2011. 2

[10] Ronghang Hu, Nikhila Ravi, Alexander C. Berg, and Deepak Pathak. Worldsheet: Wrapping the world in a 3D sheet for view synthesis from a single image. In *International Conference on Computer Vision*, 2021. 2

[11] Michael Iliadis, Leonidas Spinoulas, and Aggelos K. Katsaggelos. Deepbinarymask: Learning a binary mask for video compressive sensing, 2016. 2

[12] Yasutaka Inagaki, Yuto Kobayashi, Keita Takahashi, Toshiaki Fujii, and Hajime Nagahara. Learning to capture light fields through a coded aperture camera. In *European Conference on Computer Vision*, pages 418–434, 2018. 1, 2, 5

[13] Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. Dynamically reparameterized light fields. In *ACM SIGGRAPH*, pages 297–306, 2000. 1

[14] Seungjae Lee, Changwon Jang, Seokil Moon, Jaebum Cho, and Byoungho Lee. Additive light field displays: realization of augmented reality with holographic optical elements. *ACM Transactions on Graphics*, 35(4):1–13, 2016. 1

[15] Yuqi Li, Miao Qi, Rahul Gulve, Mian Wei, Roman Genov, Kiriakos N. Kutulakos, and Wolfgang Heidrich. End-to-end video compressive sensing using anderson-accelerated unrolled networks. In *International Conference on Computational Photography*, pages 137–148, 2020. 2

[16] Chia-Kai Liang, Tai-Hsu Lin, Bing-Yi Wong, Chi Liu, and Homer H Chen. Programmable aperture photography: multiplexed light field acquisition. *ACM Transactions on Graphics*, 27(3):1–10, 2008. 1, 2

[17] Kazuki Maeno, Hajime Nagahara, Atsushi Shimada, and Rin-Ichiro Taniguchi. Light field distortion feature for transparent object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2786–2793, 2013. 1

[18] Kshitij Marwah, Gordon Wetzstein, Yosuke Bando, and Ramesh Raskar. Compressive light field photography using overcomplete dictionaries and optimized projections. *ACM Transactions on Graphics*, 32(4):1–12, 2013. 1, 2

[19] Ehsan Miandji, Saghi Hajisharif, and Jonas Unger. A unified framework for compression and compressed sensing of light fields and light field videos. *ACM Transactions on Graphics*, 38(3):1–18, 2019. 2

[20] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics*, 38:1–14, 2019. 1

[21] Vishal Monga, Yuelong Li, and Yonina C. Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, 2021. 2

[22] Ofir Nabati, David Mendlovic, and Raja Giryes. Fast and accurate reconstruction of compressed color light field. In *International Conference on Computational Photography*, pages 1–11, 2018. 1, 2

[23] Hajime Nagahara, Changyin Zhou, Takuya Watanabe, Hiroshi Ishiguro, and Shree K Nayar. Programmable aperture camera using LCoS. In *European Conference on Computer Vision*, pages 337–350, 2010. 1, 2

[24] Ren Ng. *Digital light field photography*. PhD thesis, Stanford University, 2006. 1

[25] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, 2(11):1–11, 2005. 1

[26] Shijie Nie, Lin Gu, Yinqiang Zheng, Antony Lam, Nobutaka Ono, and Imari Sato. Deeply learned filter response functions for hyperspectral reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4767–4776, 2018. 2

[27] Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3D ken burns effect from a single image. *ACM Transactions on Graphics*, 38(6):1–15, 2019. 2

[28] Ramesh Raskar, Amit Agrawal, and Jack Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. *ACM Transactions on Graphics*, 25(3):795–804, 2006. 2

[29] Raytrix:. 3D light field camera technology, 2021. https://www.raytrix.de/. 1

[30] Dikpal Reddy, Ashok Veeraraghavan, and Rama Chellappa. P2C2: Programmable pixel compressive camera for high speed imaging. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 329–336, 2011. 2

[31] Kohei Sakai, Keita Takahashi, Toshiaki Fujii, and Hajime Nagahara. Acquiring dynamic light fields through coded aperture camera. In *European Conference on Computer Vision*, pages 368–385, 2020. 1, 2, 5, 6, 7, 8

[32] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 5

[33] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3D photography using context-aware layered depth inpainting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 2

[34] Changha Shin, Hae-Gon Jeon, Youngjin Yoon, In So Kweon, and Seon Joo Kim. EPINET: A fully-convolutional neural network using epipolar geometry for depth from light field images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4748–4757, 2018. 1

[35] Pratul P. Srinivasan, Tongzhou Wang, Ashwin Sreelal, Ravi Ramamoorthi, and Ren Ng. Learning to synthesize a 4D RGBD light field from a single image. In *IEEE International Conference on Computer Vision*, pages 2262–2270, 2017. 2

[36] He Sun, Adrian V. Dalca, and Katherine L. Bouman. Learning a probabilistic strategy for computational imaging sensor selection. In *International Conference on Computational Photography*, pages 81–92, 2020. 2

[37] Yuichi Taguchi, Takafumi Koike, Keita Takahashi, and Takeshi Naemura. TransCAIP: A live 3D TV system using a camera array and an integral photography display with interactive control of viewing parameters. *IEEE Transactions on Visualization and Computer Graphics*, 15(5):841–852, 2009. 1

[38] Keita Takahashi, Yuto Kobayashi, and Toshiaki Fujii. From focal stack to tensor light-field display. *IEEE Transactions on Image Processing*, 27(9):4571–4584, 2018. 1

[39] Salil Tambe, Ashok Veeraraghavan, and Amit Agrawal. Towards motion aware light field video for dynamic scenes. In *IEEE International Conference on Computer Vision*, pages 1009–1016, 2013. 1

[40] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2

[41] Anil Kumar Vadathya, Sharath Girish, and Kaushik Mitra. A unified learning based framework for light field reconstruction from coded projections. *IEEE Transactions on Computational Imaging*, 6:304–316, 2019. 1, 2

[42] Edwin Vargas, Julien N. P. Martel, Gordon Wetzstein, and Henry Arguello. Time-multiplexed coded aperture imaging: Learned coded aperture and pixel exposures for compressive imaging systems. In *International Conference on Computer Vision*, 2021. 2

[43] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. *ACM Transactions on Graphics*, 26(3):69, 2007. 1

[44] Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. Single disperser design for coded aperture snapshot spectral imaging. *Appl. Opt.*, 47(10):B44–B51, 2008. 2

[45] Ting-Chun Wang, Jun-Yan Zhu, Ebi Hiroaki, Manmohan Chandraker, Alexei Efros, and Ravi Ramamoorthi. A 4D light-field dataset and cnn architectures for material recognition. In *European Conference on Computer Vision*, volume 9907, pages 121–138, 2016. 1

[46] Ting-Chun Wang, Jun-Yan Zhu, Nima Khademi Kalantari, Alexei A. Efros, and Ravi Ramamoorthi. Light field video capture using a learning-based hybrid imaging system. *ACM Transactions on Graphics*, 36(4):133:1–133:13, 2017. 1

[47] Yunlong Wang, Fei Liu, Zilei Wang, Guangqi Hou, Zhenan Sun, and Tieniu Tan. End-to-end view synthesis for light field imaging with pseudo 4DCNN. In *European Conference on Computer Vision*, 2018. 2

[48] Mian Wei, Navid Sarhangnejad, Zhengfan Xia, Nikita Gusev, Nikola Katic, Roman Genov, and Kiriakos N. Kutulakos. Coded two-bucket cameras for computer vision. In *European Conference on Computer Vision*, pages 55–73, 2018. 2

[49] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. *ACM Transactions on Graphics*, 24(3):765–776, 2005. 1

[50] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 2

[51] W. Williem, In Kyu Park, and Kyoung Mu Lee. Robust light field depth estimation using occlusion-noise aware data costs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2484–2497, 2018. 1

[52] Yicheng Wu, Vivek Boominathan, Huaijin Chen, Aswin Sankaranarayanan, and Ashok Veeraraghavan. Phasecam3D — learning phase masks for passive single view depth estimation. In *International Conference on Computational Photography*, pages 1–12, 2019. 2

[53] Henry Wing Fung Yeung, Junhui Hou, Xiaoming Chen, Jie Chen, Zhibo Chen, and Yuk Ying Chung. Light field spatial super-resolution using deep efficient spatial-angular separable convolution. *IEEE Transactions on Image Processing*, 28(5):2319–2330, 2019. 2

[54] Michitaka Yoshida, Toshiki Sonoda, Hajime Nagahara, Kenta Endo, Yukinobu Sugiyama, and Rin-ichiro Taniguchi. High-speed imaging using CMOS image sensor with quasi pixel-wise exposure. *IEEE Transactions on Computational Imaging*, 6:463–476, 2020. 1, 2, 4

[55] Michitaka Yoshida, Akihiko Torii, Masatoshi Okutomi, Kenta Endo, Yukinobu Sugiyama, Rin-ichiro Taniguchi, and Hajime Nagahara. Joint optimization for compressive video

sensing and reconstruction under hardware constraints. In *European Conference on Computer Vision*, 2018. 5

[56] Xin Yuan, David J. Brady, and Aggelos K. Katsaggelos. Snapshot compressive imaging: Theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 38(2):65–88, 2021. 2

[57] Xin Yuan, Yang Liu, Jinli Suo, and Qionghai Dai. Plug-and-play algorithms for large-scale snapshot compressive imaging. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1444–1454, 2020. 2

[58] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Transactions on Graphics*, 37:1–12, 2018. 1