Synthesizing Light Field Video from Monocular Video

Shrisudhan Govindarajan, Prasan Shedligeri, Sarah, and Kaushik Mitra

Indian Institute of Technology Madras, Chennai, India

Abstract. The hardware challenges associated with light-field (LF) imaging has made it difficult for consumers to access its benefits like applications in post-capture focus and aperture control. Learning-based techniques which solve the ill-posed problem of LF reconstruction from sparse (1, 2 or 4) views have significantly reduced the need for complex hardware. LF *video* reconstruction from sparse views poses a special challenge as acquiring ground-truth for training these models is hard. Hence, we propose a self-supervised learning-based algorithm for LF video reconstruction from monocular videos. We use self-supervised geometric, photometric and temporal consistency constraints inspired from a recent learning-based technique for LF video reconstruction from stereo video. Additionally, we propose three key techniques that are relevant to our monocular video input. We propose an explicit disocclusion handling technique that encourages the network to use information from adjacent input temporal frames, for inpainting disoccluded regions in a LF frame. This is crucial for a self-supervised technique as a single input frame does not contain any information about the disoccluded regions. We also propose an adaptive low-rank representation that provides a significant boost in performance by tailoring the representation to each input scene. Finally, we propose a novel refinement block that is able to exploit the available LF image data using supervised learning to further refine the reconstruction quality. Our qualitative and quantitative analysis demonstrates the significance of each of the proposed building blocks and also the superior results compared to previous state-of-the-art monocular LF reconstruction techniques. We further validate our algorithm by reconstructing LF videos from monocular videos acquired using a commercial GoPro camera. An open-source implementation is also made available[1].

Keywords: Light-fields, Plenoptic function, Self-supervised learning

1 Introduction

Cameras have become cheap and ubiquitous in the modern world, giving consumers a capability to acquire photos and videos anywhere and anytime. The last decade saw an accelerated improvement in image sensors and lens quality, leading to a significant improvement in the picture quality from these tiny

[1] https://github.com/ShrisudhanG/Synthesizing-Light-Field-Video-from-Monocular-Video

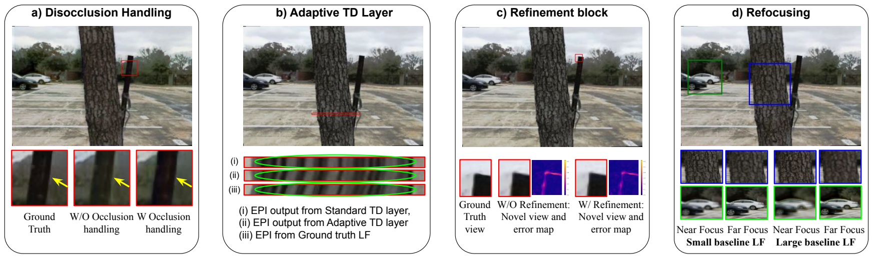

Fig. 1. We propose three novel techniques: a) disocclusion handling, b) adaptive low-rank representation for LF and c) a novel refinement block, for LF video reconstruction from monocular video. Combining these with the self-supervised cost functions inspired by [35], we can reconstruct high-fidelity LF videos, even with varying baselines. As shown in (d) this allows us to control the synthetic defocus blur using the output video.

cameras. Towards the end of the decade, the focus shifted towards more and more innovative software, pushing the limits to what can be achieved with these ubiquitous cameras [8]. This push resulted in a variety of features: ranging from simple effects like background-blur to more dramatic ones like augmented reality. Features like bokeh effects and novel view synthesis became popular as they provided a sense of '3D' to the otherwise flat pictures. However, these features have currently been limited to images and there's no straightforward way of extending them to videos. In the last few years, videos have certainly become a more powerful means of communication, knowledge-sharing and even entertainment. LF imaging could provide an intuitive way of bringing these features to videos. However, there's no easy way to capture LF videos yet. Computational photography is poised to solve this, making it easy and accessible to capture LF on small form-factor devices [20]. We instead focus on existing camera hardware and aim to reconstruct LF videos from any ordinary monocular camera.

Traditionally, LF imaging required use of bulky or complex hardware setups such as camera arrays [48] and micro-lens arrays [30]. Hence, the recent focus has been on reducing the hardware complexity through the use of learning-based techniques. Typically, these involve the reconstruction of LF from sparse input views (such as 1, 2 or 4 views) [18,52,38,22]. To solve the challenges in acquiring LF videos through commercial cameras, several techniques for LF *video* reconstruction have also been proposed [2,45,35]. SeLFVi [35] is an interesting recent work that proposes a novel self-supervised technique for LF video reconstruction from stereo videos. Being a self-supervised technique it relied on an intermediate low-rank representation for LF frames achieving high-quality reconstructions. However, it requires a stereo video input where both cameras should have identical focal lengths (identical field-of-view). This can become a limitation considering that stereo cameras are still not as widespread as monocular cameras. This is especially true for consumer applications, where mostly monocular cameras are preferred.

Motivated by the availability of large and diverse sets of high-quality monocular videos we propose a novel, self-supervised learning technique for LF video reconstruction from monocular input. To start with, we preserve the self-supervised photometric, geometric and temporal consistency constraints adopted in SeLFVi (see Sec. 3.3). Further we introduce three crucial blocks that are necessary for our case of monocular video input. These are: 1) a novel loss for handling disocclusion, 2) a scene geometry adaptive intermediate low-rank representation and 3) a novel and supervised refinement block to further refine the LF video(see Fig. 1).

The challenge with just a monocular input is that there's no information on how to fill the disoccluded regions/pixels of the predicted LF. We propose a technique to *inpaint* the disoccluded regions of the estimated LF frames. The intuition is that, in a video acquired using a moving camera, occluded regions in one frame might be visible in the neighboring temporal frames. Our disocclusion handling technique (Sec. 3.4) utilizes this existing information to fill in the disoccluded regions of the LF frame.

Next, we modify the standard tensor-display (TD) based intermediate low-rank representation so that it can adapt to any input scene. While TD model [47] uses fixed displacement between the layers, we propose a modification where this displacement can be modified for each input image (Sec. 3.2). In [47], each of the layers are shown to represent a depth-plane in the scene. Hence, by estimating the displacement values for each scene, the layers are better able to represent the given LF. This idea was inspired from a similar choice of adaptive layered representation in [22] for novel view synthesis. Unlike [22] we adopt a more sophisticated approach to predict the depth planes through global scene understanding by using transformers [3]. As shown in our experiments, the adaptive low-rank representation provides a significant boost in the quality of the predicted LF frames.

Finally, we explore the popular idea of self-supervised pre-training, followed by supervised learning on a small amount of data to boost the performance of a model [6][5][17]. We design a novel convolutional vision-transformer-based [9] refinement block that is trained via supervised learning on a small amount of LF *image* data. This helps in further refining the output around the depth-edges that are difficult to reconstruct with just self-supervised learning. The final output is a weighted combination of the refinement block output and the LF estimated by self-supervised learning (Sec. 3.5). To the best of our knowledge, this is the first time that vision transformers are used to supervise LF reconstruction by efficiently combining the spatio-angular information. In summary, we make the following contributions:

- High quality reconstruction of LF videos from monocular video with self-supervised learning.
- Handling disocclusions in rendering LFs using self-supervised consistency losses utilizing information from successive video frames.
- A modified TD-based low-rank representation that can adapt to the given input scene dynamically adjusting the distance between the layers.
- A novel supervised vision-transformer based refinement block to exploit the small amount of LF image data to further improve reconstruction on video.

2 Related Work

LF synthesis While the concept of LF or integral imaging is quite old [23,1], capturing these images has been complicated. While commercial LF cameras are now available in the market [30], they suffer from low spatial resolution. Over the last several years, a diverse set of camera setups and algorithms have aimed at making LF imaging simpler and more accessible. There have been setups that use coded-aperture systems [43,14,34], cameras with coded masks near the sensor [27,12] and even hybrid sensors [45]. Later, with advances in deep-learning, systems using ordinary commercial cameras such as one or more DSLRs became popular. Techniques that reconstruct LF frames from focus-defocus pair [42] or focal-stack images[4] were proposed. Several techniques were also proposed that could reconstruct LF from sparse set of views on a regular grid. The number of views could be 1-view[38,22,2,15], 2-views[52,35], 4-views[18,46,51], and even 9-views[49].

LF synthesis from monocular image As ordinary monocular cameras are ubiquitous, several techniques aim at LF reconstruction from them. As this is an ill-posed problem, learning-based techniques have been essential in this domain. A popular technique has been to first predict disparity flow [38] or appearance flow[15][54] and then warp the input image accordingly to reconstruct the LF frame. Recently, Multi-Plane Image (MPI) based representation is being used for LF prediction [13,53,37,28,22]. Li *et al.* [22] propose a modified MPI model that allowed them to significantly reduce the representation complexity. With a similar intuition, we propose a modified low-rank representation based on layered LF displays [47] for predicting the LF frames.

LF video reconstruction As commercial LF cameras such as Lytro acquire videos at only 3 frames per second (fps), LF video acquisition at high angular and temporal resolution has also been challenging. In [45] a learning-based algorithm with a hybrid camera system consisting of a general DSLR camera and a light field camera was proposed. Hajisharif *et al.* [12] proposed a single sensor-based algorithm that required a coded mask to be placed in front of the sensor. As these algorithms require complex and bulky hardware setups, techniques such as [2,35] are proposed that just require ordinary cameras. Although our self-supervised algorithm is inspired from [35], the closest work to ours is [2]. Bae *et al.*[2] utilize a large set of computer-generated data to supervise a neural network for LF video reconstruction from monocular video. In contrast, our proposed technique does not require hard-to-acquire LF video data for supervision.

Learning with layered LF representation Previously, layered LF display representations [47] have been used in conjunction with neural networks. [26] built an end-to-end pipeline from a coded aperture scene acquisition for displaying the scene on a layered LF display. Similar work in [39,21] aims at capturing a focal stack and then learning to display the scene onto the LF display. Inspired by [35], we also adopt the layered LF display based intermediate low-rank representation $\mathcal{F}$ for LF estimation. We extend the standard low-rank model to adapt

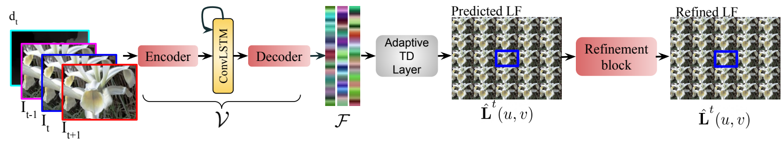

Fig. 2. Our proposed algorithm takes as input a sequence $\mathcal{I} = \{I_{t-1}, I_t, I_{t+1}, d_t\}$. A recurrent LF synthesis network $\mathcal{V}$ first predicts an intermediate low-rank representation $\mathcal{F}$ for the corresponding LF frame. An adaptive TD layer (3.2) takes the same set $\mathcal{I}$ and $\mathcal{F}$ as input and outputs the LF frame $\hat{\mathbf{L}}_t$. A set of self-supervised cost-functions (3.3, 3.4) are then imposed on $\hat{\mathbf{L}}_t$ for the end-to-end training of $\mathcal{V}$ and the adaptive TD layer. Finally, a refinement block (3.5) then takes $\hat{\mathbf{L}}_t$ and I_t as input and outputs a refined LF.

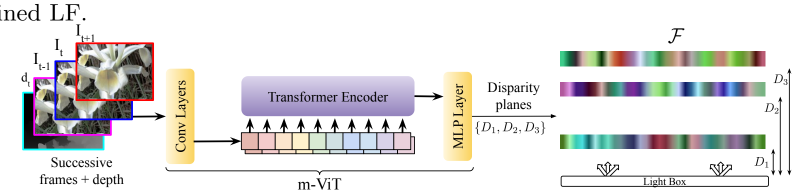

Fig. 3. We use an m-VIT block [3] to predict the displacement between the different layers of the TD based low-rank representation. Each of the layers in $\mathcal{F}$ approximately represent a scene depth plane. Instead of keeping the layers static/fixed, a scene-specific displacement value will move the layer to a depth plane where it can best represent the scene.

to the individual scene by predicting the optimal distance between the layers for each input image.

3 Monocular LF video estimation

We propose a self-supervised learning based technique for LF video reconstruction from a monocular video sequence. For each input frame of the monocular video, we reconstruct a corresponding LF video frame. As shown in Fig. 2, a deep neural network takes as input, a sequence of 3 input frames and a disparity map $\{I_{t-1}, I_t, I_{t+1}, d_t\}$ and estimates an intermediate low-rank representation of the current LF frame $\hat{\mathbf{L}}_t$. As shown in Fig. 3 and further elaborated in Sec. 3.2, we propose a modified intermediate low-rank representation adapted from [47]. After obtaining $\hat{\mathbf{L}}_t$ from the adaptive TD layer, we introduce the geometric, photometric and the temporal consistency constraints [35] to train our LF synthesis network (see Sec. 3.3). Being a self-supervised technique, we do not have any information about the disoccluded regions in $\hat{\mathbf{L}}_t$ from just I_t. Hence, we introduce a disocclusion handling technique that utilizes information from I_{t-1} and I_{t+1} to fill-in the disoccluded regions of $\hat{\mathbf{L}}_t$(see Sec. 3.4). Finally, to further refine the estimated LF frame, we propose a novel residual refinement block based on vision-transformers (see Sec. 3.5) which is trained using supervised learning.

3.1 Light field frame prediction

As shown in Fig. 2, we stack three successive input frames and the corresponding disparity map as $\mathcal{I}$ =$\{I_{t-1}, I_t, I_{t+1}$,$d_t\}$ and feed it to the LF prediction network $\mathcal{V}$. With a monocular input, it's not possible to obtain a disparity map directly. Hence, we first estimate a relative depth map z_t of I_t using a pre-trained monocular depth estimation model [33]. We know that a disparity map is related to the depth map up to an affine transformation [10], defined here as $d_t = az_t + b$, where a and b are two scalars. During training, we randomly sample values of a and b and convert the relative depth map z_t to the disparity map d_t. The network $\mathcal{V}$ is a long short term memory (LSTM) based model consisting of an encoder and decoder with skip connections. The network $\mathcal{V}$ predicts an intermediate low-rank representation $\mathcal{F}$ for $\hat{\mathbf{L}}_t$ based on a modified tensor-display model [47]. We describe the process of obtaining $\hat{\mathbf{L}}_t$ from the low-rank representation $\mathcal{F}$ in Sec. 3.2.

3.2 Adaptive tensor-display model

In the previous section, we estimated the representation $\mathcal{F}$ from the network $\mathcal{V}$, based on the low-rank model proposed in [47]. In this standard model, $\mathcal{F} = [f_{-N/2}, \ldots, f_0, \ldots, f_{N/2}]$, where $f_k = [f_k^1, f_k^2, \ldots, f_k^R]$, $f_k^r \in [0,1]^{h \times w \times 3}$. Here N represents the number of layers in the low-rank model and R represents its corresponding rank. Given $\mathcal{F}$, the corresponding 4D LF frame can be computed as

$$L(x, y, u, v) = TD(\mathcal{F}) = \sum_{r=1}^{R} \prod_{n=-N/2}^{N/2} f_n^r(x + nu, y + nv) \tag{1}$$

where x, y and u, v respectively denote the spatial and angular coordinates. Further analysis into these representations in [47] showed that each layer approximately represents a particular depth plane in the scene. However, the standard model places these layers at a uniform distance from each other representing depth planes placed uniformly in the scene. In a natural image the objects in the scene could be distributed non-uniformly throughout the depth. This idea was exploited in [22], where the standard MPI model was adapted to each input image by assigning non-uniform disparity values for each MPI layer. This drastically reduced the number of MPI layers required to represent the scene up to a similar accuracy.

Motivated by this we use a m-VIT network [3], to predict a sequence of values, $D = \{D_{-N/2}, \ldots D_{N/2}\}$, that will be used in adapting the TD layer to each input (Fig. 3). m-VIT predicts one value for each layer in the representation $\mathcal{F}$ using the input $\mathcal{I} = \{I_{t-1}, I_t, I_{t+1}, d_t\}$. The values in D are used in the proposed adaptive TD layer as

$$L(x, y, u, v) = TD(\mathcal{F}; D) = \sum_{r=1}^{R} \prod_{n=-N/2}^{N/2} f_n^r(x + D_n u, y + D_n v) \ , \tag{2}$$

where D_n represents the scalar value predicted by m-VIT for layer n. After computing $\hat{\mathbf{L}}_t$ from our proposed adaptive TD layer, we impose three main self-supervised cost functions to train the prediction network $\mathcal{V}$.

3.3 Loss functions

To successfully train the LF prediction network $\mathcal{V}$, we follow [35] and define three constraints that enforce the structure of the LF video on the predicted sequence of frames.

Photometric constraint The photometric constraint is defined on the premise that the center view of $\hat{\mathbf{L}}_t$ should match the current input frame I_t. Hence, we define the loss function reflecting this as $\mathcal{L}^t_{photo} = \|\hat{\mathbf{L}}_t\,(\mathbf{0}) - I_t\|_1$, where $\hat{\mathbf{L}}_t\,(\mathbf{0})$ represents the central angular view of $\hat{\mathbf{L}}_t$.

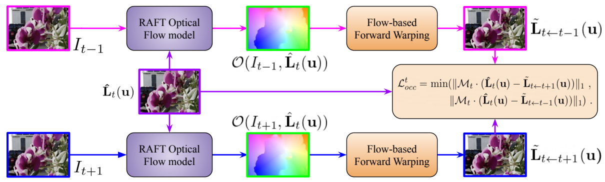

Fig. 4. We introduce disocclusion handling to constrain the synthesis network to fill in the disoccluded pixels of the LF with information from the neighboring frames. As shown, I_{t-1} is forward warped to each view of $\hat{\mathbf{L}}_t$, using the optical flow predicted from RAFT [41]. We also warp I_{t+1} to each SAIs of $\hat{\mathbf{L}}_t$, and the loss is computed as in Eq. (8).

Geometric constraint To compute the cost, we first warp all sub-aperture images (SAIs) of the $\hat{\mathbf{L}}_t$ to the SAI $\mathbf{0}$ that corresponds to I_t. In essence, we warp $\hat{\mathbf{L}}_t(\mathbf{u})$ to the SAI $\mathbf{0}$ to obtain $\hat{\mathbf{L}}_t(\mathbf{u} \to \mathbf{0})$, expressed as,

$$\hat{\mathbf{L}}_t(\mathbf{u} \to \mathbf{0}) = \mathcal{W}\left(\hat{\mathbf{L}}_t\left(\mathbf{u}\right); \left(\mathbf{u} - \mathbf{0}\right) d_t\right) \ . \tag{3}$$

Here, $\mathcal{W}$ denotes the bilinear inverse warping operator [16] that takes as input a displacement map and remaps the images. The geometric consistency error between the approximated current frame $\hat{\mathbf{L}}_t(\mathbf{u} \to \mathbf{0})$ and I_t is then defined as,

$$\mathcal{L}^t_{geo} = \sum_{\mathbf{u}} \|\hat{\mathbf{L}}_t\left(\mathbf{u} \to \mathbf{0}\right) - I_t\|_1 \ . \tag{4}$$

Temporal consistency constraint In addition to an LSTM network-based [36] recurrent framework of our network $\mathcal{V}$, we impose a temporal consistency constraint on the predicted outputs. For this, we first estimate the optical flow between successive input video frames using a pre-trained RAFT [41] network, denoted as $\mathcal{O}$. The optical flow is then computed as $o_t = \mathcal{O}(I_t, I_{t+1})$. To enforce temporal consistency, we utilize the warped angular views $\hat{\mathbf{L}}_t\left(\mathbf{u} \to \mathbf{0}\right)$ and again warp them to the video frame at $t+1$ using o_t. Then, the temporal consistency error is defined as the error between the known next frame I_{t+1} and these warped frames and is denoted as,

$$\mathcal{L}^t_{temp} = \sum_{\mathbf{u}} \|\mathcal{W}\left(\hat{\mathbf{L}}_t\left(\mathbf{u} \to \mathbf{0}; o_t\right)\right) - I_{t+1}\|_1 \ . \tag{5}$$

Minimizing this error during training explicitly enforces temporal consistency between the successive predicted LF frames.

3.4 Disocclusion handling

In a LF, pixels at depth boundary of objects get occluded and disoccluded between different SAIs. Due to the lack of ground truth data we face a major challenge when learning to fill-in the intensity values at the disoccluded pixels. Our objective is to use pixels in I_{t-1} and I_{t+1} to fill in the disoccluded pixels of $\hat{\mathbf{L}}_t$, as these frames could potentially have the necessary pixel values. Pixels from neighboring video frames have been used to inpaint the current frame in several video-inpainting techniques [19,50]. Here, we achieve this by bringing the informative intensity values to the disoccluded pixels through flow-based warping. As shown in Fig. 4, we use RAFT to obtain optical flow between the SAIs of $\hat{\mathbf{L}}_t$ and the input frames I_{t-1}, I_{t+1}. By including these warping based operations in a loss function, we train the network to automatically predict the disoccluded pixels.

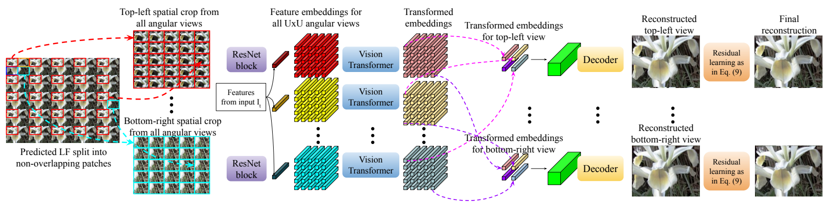

Fig. 5. A supervised residual refinement block is used to further improve the reconstruction quality of the LFs. The transformer block attends to the spatio-angular information in the estimated LF and the input frame I_t to predict the refined output.

The loss is defined on only those pixels that are disoccluded in the SAIs of the LF frame $\hat{\mathbf{L}}_t$. We obtain the disoccluded pixels by forward-warping the input frame I_t to all the SAIs of LF with the disparity map d_t. For each SAI at angular location $\mathbf{u}$, we define a binary mask $\mathcal{M}_t$ which is 1 if forward warping resulted in a hole for that particular pixel. To fill the dis-occluded pixels, we forward warp $I_{t-1}$to the predicted SAIs of $\hat{\mathbf{L}}_t$ using optical flow as shown in Fig. 4. The forward warped SAIs are obtained as:

$$\tilde{\mathbf{L}}_{t\leftarrow t-1}(\mathbf{u}) = \mathcal{W}\left(I_{t-1}; \mathcal{O}\left(I_{t-1}, \hat{\mathbf{L}}_t(\mathbf{u})\right)\right), \tag{6}$$

$$\tilde{\mathbf{L}}_{t\leftarrow t+1}(\mathbf{u}) = \mathcal{W}\left(I_{t+1}; \mathcal{O}\left(I_{t+1}, \hat{\mathbf{L}}_t(\mathbf{u})\right)\right), \tag{7}$$

where $\tilde{\mathbf{L}}_{t\leftarrow t+1}(\mathbf{u})$ and $\tilde{\mathbf{L}}_{t\leftarrow t-1}(\mathbf{u})$ represent the forwards warped SAIs from I_{t+1} and I_{t-1} respectively using optical flow. Depending on the camera motion the disoccluded pixels in $\hat{\mathbf{L}}_t$could be visible either in I_{t-1} or I_{t+1} or both. Taking this into consideration, we define the cost function as

$$\mathcal{L}^t_{occ} = \min(\|\mathcal{M}_t \cdot \left(\hat{\mathbf{L}}_t(\mathbf{u}) - \tilde{\mathbf{L}}_{t\leftarrow t-1}(\mathbf{u})\right)\|_1 \ , \ \|\mathcal{M}_t \cdot \left(\hat{\mathbf{L}}_t(\mathbf{u}) - \tilde{\mathbf{L}}_{t\leftarrow t+1}(\mathbf{u})\right)\|_1) \tag{8}$$

$\mathcal{L}^t_{occ}$ follows the concept of minimum re-projection loss followed in monocular depth estimation techniques such as [11].

3.5 Supervised residual refinement block

Recently, self-supervised pre-training on very large unlabeled datasets followed by supervised learning on a limited labeled dataset has helped in achieving state-of-the-art results [6,5,17]. Inspired by these works, we propose to use the limited dataset of LF *images* to further refine the reconstructed LF frames. As this shouldn't affect the temporal consistency of the predicted frames, the proposed refinement module follows a residual network architecture as shown in Fig. 5. And this module can be trained as a separate block from the recurrent module in the synthesis network $\mathcal{V}$.

Vision Transformers(ViT) [9] form the backbone of our proposed refinement module. As shown in Fig. 5, we divide the predicted LF frame $\hat{\mathbf{L}}_t$ into non-overlapping patches, each of size $p \times p$. For simplicity consider all the U^2 top-left patches cropped from each angular view of $\hat{\mathbf{L}}_t$. A shallow ResNet-based neural network extracts features independently from each of the U^2 patches. Additionally, we also extract features from the top-left patch of the input image I_t. The transformer module then takes as input the U^2+1 features/embeddings as input and outputs U^2+1 tokens after applying multi-headed self-attention (MHSA) [9]. An identical procedure is repeated on all the non-overlapping patches of $\hat{\mathbf{L}}_t$ to produce U^2+1 tokens each time.

As in Fig. 5, we discard the token from the input frame and consider all the P transformed tokens from a particular angular view, say bottom-right. Here, P is the number of non-overlapping patches cropped from each angular view. These P tokens are stacked horizontally and vertically following the order of cropped patches, so as to form a larger feature map. A shallow decoder network then takes these stacked tokens as input and predicts a 4 channel output. The first 3 channels form an RGB image ($\tilde{\mathbf{L}}_t^{ref}(\mathbf{u})$) and the fourth channel is the mask $M_{ref} \in [0,1]^{h \times w}$. The final output $\hat{\mathbf{L}}_t^{ref}$ is then defined as,

$$\hat{\mathbf{L}}_t^{ref}(\mathbf{u}) = M_{ref} \odot \hat{\mathbf{L}}_t(\mathbf{u}) + (1 - M_{ref}) \odot \tilde{\mathbf{L}}_t^{ref}(\mathbf{u}) \ . \tag{9}$$

Identical decoding step is repeated for each SAI $\mathbf{u}$ producing a refined LF frame $\hat{\mathbf{L}}_t^{ref}$. As we assume access to a LF *image* dataset, we train the refinement network by imposing L1 loss between $\hat{\mathbf{L}}_t^{ref}$ and the corresponding ground-truth $\mathbf{L}_t$ as:

$$\mathcal{L}_{ref} = \sum_{\mathbf{u}} \|\hat{\mathbf{L}}_t^{ref}(\mathbf{u}) - \mathbf{L}_t(\mathbf{u})\|_1 \ . \tag{10}$$

3.6 Overall loss

We finally add total-variation(TV)-based smoothness constraint[35] on the predicted LF frames and Bin-center density loss [3] on disparity values predicted by m-VIT. The Bin-center density loss encourages the predicted disparity planes to be close to the disparity map d_t which is provided as input to the adaptive TD layer. Including all the cost functions, the overall loss to minimize for training $\mathcal{V}$ and the adaptive TD layer becomes,

$$\mathcal{L}_{self}^t = \lambda_1 \mathcal{L}_{photo}^t + \lambda_2 \mathcal{L}_{geo}^t + \lambda_3 \mathcal{L}_{temp}^t + \lambda_4 \mathcal{L}_{occ}^t + \lambda_5 \mathcal{L}_{bins}^t + \lambda_6 \mathcal{L}_{TV}^t \ , \tag{11}$$

where the parameters λ_i control the contribution of each loss term. After the self-supervised training of the main network is completed, we then freeze these weights and train the refinement block. The refinement block is trained using a supervised cost function $\mathcal{L}_{ref}$ in Eq. (10).

3.7 Implementation details

As shown in Fig. 2, our proposed pipeline has three separate deep neural networks: (a) LF synthesis network, (b) adaptive TD layer (Fig. 3) and (c) refinement network (Fig. 5). The synthesis network $\mathcal{V}$ is a LSTM based recurrent neural network consisting of a Efficient-Net encoder [40] and a convolutional decoder with skip connections. In the adaptive TD layer, we set the low-rank representation $\mathcal{F}$ to have $N = 3$ layers and the rank $R = 12$ following [35]. The displacements $D = \{D_1, D_2, D_3\}$ are predicted from m-VIT[3] network that takes as input $\{I_{t-1}, I_t, I_{t+1}, d_t\}$. Finally, the refinement network has a backbone of the convolutional vision transformer which is supervised using a limited amount of LF *image* data. Further details of the neural networks can be found in the supplementary material.

For training our proposed synthesis network, we use the *GOPRO* monocular video dataset[29]. The *GOPRO* dataset contains monocular videos of 33 different scenes each containing 525 to 1650 monocular frames of spatial resolution 720×1280. We split the dataset into a set of 25 videos for training and 8 videos for validation. The monocular video frames are resized into frames of size 352×528 to maintain the spatial resolution of Lytro Illum light field camera. While training we obtain a monocular video of 7 frames and randomly crop a

Algorithm	Hybrid		ViewSynth		TAMULF		Stanford		Average	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Niklaus *et al.*[31]	23.87	0.873	23.19	0.903	18.12	0.811	23.19	0.892	22.10	0.870
Srinivasan *et al.*[38]	28.12	0.893	28.56	0.931	22.63	0.857	29.24	0.924	27.14	0.901
Li *et al.*[22]	31.62	0.950	29.39	0.945	25.63	0.903	30.44	0.956	29.27	0.938
Li+Ranftl[33]	31.69	0.950	29.90	0.953	25.83	0.906	31.21	0.962	29.66	0.943
Proposed(image)	**32.48**	**0.951**	**30.76**	**0.955**	**27.42**	**0.927**	**34.53**	**0.970**	**31.30**	**0.951**
Proposed	**32.66**	**0.952**	**30.97**	**0.956**	**27.24**	**0.922**	**34.98**	**0.974**	**31.47**	**0.951**

Table 1. We quantitatively compare our proposed technique with state-of-the-art algorithms on various datasets. Our algorithm consistently provides high-fidelity reconstructions. **Blue** and **green** represent the top-two performing algorithm in each column.

patch of size 176×264. The successive frames in the training data are 10 frames apart in the raw GoPro videos captured at 240 fps. This ensures that there's reasonable object motion between successive input frames which is crucial for the disocclusion handling technique. In one frame, closer objects show larger disocclusions in the predicted LF as they have higher disparity values. These objects also proportionally have larger displacements in successive frames, providing enough information to fill in the disoccluded pixels.

The relative depth map input to the network is obtained from [33] and then modified for various baseline factors to enable the synthesis network to generate LF outputs of various baseline. We randomly choose a value for $a \in \{0.8, 1.6, 2.4, 3.2\}$ and $b \in [0.2, 0.4]$ to obtain disparity $d_t = az_t + b$ as explained in Sec. 3.1. The network is trained in Pytorch[32] using AdamW [24] optimizer for 25 epochs, with an initial learning rate of 0.0001 and weight decay of 0.001. The learning rate is decreased to half the initial value when the validation loss plateaus for more more than 4 epochs. We empirically choose the hyperparameters as $\lambda_1 = 1.0$, $\lambda_2 = 1.0$, $\lambda_3 = 0.5$, $\lambda_4 = 0.2$, $\lambda_5 = 2$ and $\lambda_6 = 0.1$ in Eq. (11).

For training our residual refinement block, we freeze the weights of the synthesis network and train only the refinement block using supervised loss function in Eq. (10). We fix the value of a as 1.2 and b as 0.3 to estimate d_t which is provided as input to the synthesis network. For the supervised training, we use 1000 LF images from *TAMULF* [22] dataset. The network is trained using AdamW optimizer for 15 epochs, with an initial learning rate of 0.001 and weight decay of 0.001. The learning rate is decreased to half the initial value when the validation loss plateaus for more more than 4 epochs.

4 Experiments

To validate our proposed algorithm, we make several qualitative and quantitative comparisons with diverse LF datasets. For quantitative comparison, we mainly consider four different datasets: *Hybrid* [45], *ViewSynth* [18], *TAMULF* [22] and *Stanford* [7] containing 30, 25, 84 and 113 light field video sequences, respectively. From the *Hybrid* dataset we consider the central 7×7 views as the ground-truth light field videos, and the center-view of each LF forms the input monocular video. The rest three datasets are LF *image* datasets, and we simulate LF videos with 8 frames from each LF following the procedure described in [35][25]. The center-view of these 7×7 view LF videos form the monocular video sequence that is given as input to our algorithm. During inference, we first obtain the

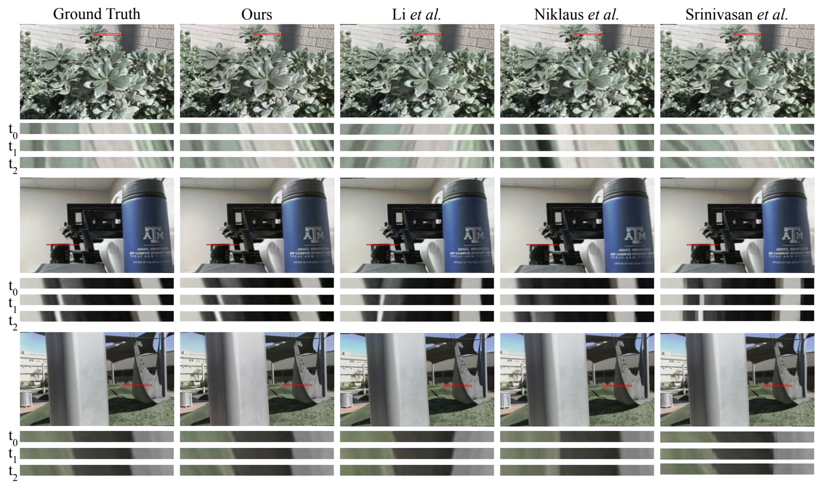

Fig. 6. We qualitatively compare our reconstruction with ground truth and other state-of-the-art techniques. We show the top-left view of t_0 and EPI images from three consecutive LF frames (t_0, t_1, t_2). As can be clearly seen from the EPI images, our technique consistently provides accurate reconstructions.

depth estimate z_t from DPT[33] and convert it to a disparity map d_t. Three consecutive temporal frames and disparity map are stacked and input to the complete model represented in Fig. 2 to obtain the LF frame output.

4.1 Light field video reconstruction

We quantitatively and qualitatively compare the results of our proposed algorithm with previously proposed monocular LF estimation techniques. For quantitative comparison, we use two metrics: peak signal-to-noise ratio (PSNR) (higher is better) and structural similarity index measure (SSIM) (higher is better). As shown in Tab. 1, we compare the performance of our proposed algorithm with Niklaus *et al.* [31], Srinivasan *et al.* [38] and Li *et al.* [22].

Li *et al.* [22] takes a single frame and a relative depth estimate from [44] as input. To obtain the complete LF video, we have to reconstruct each frame of the video individually. In Tab. 1, Li *et al.* + Ranftl *et al.* represents a modified [22], where we input a depth estimate from DPT[33] instead of the original DeepLens[44] model. This is done to ensure a fair comparison with our technique as we also use DPT, which is a state-of-the-art monocular depth estimation technique based on vision transformers. However, [22] is not trained for inputs from DPT [33]. Hence, we finetune [22] on the TAMULF dataset with depth maps from DPT [33]. Srinivasan *et al.* [38] is another single image LF estimation model. While the original network is trained on a dataset of flower images (proposed in the same work), we finetune it on a larger and diverse TAMULF dataset from [22]. Finally, we also compare our algorithm with Niklaus *et al.* [31] that takes a single frame as input. We used the default implementation provided by the authors for comparison, which is already trained on a diverse dataset. As all these techniques are image-based and don't have any temporal information, we also compare with a *downgraded* version of our algorithm 'Proposed(image)'. In

Algorithm	Hybrid		ViewSynth		TAMULF		Stanford		Average	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Base	30.76	0.945	29.07	0.947	25.74	0.918	31.70	0.963	29.32	0.943
Base+occ	31.78	0.949	29.71	0.948	26.51	**0.919**	32.99	0.965	30.25	0.945
Base+occ+adpt	**32.26**	**0.950**	**30.69**	**0.954**	**26.96**	**0.919**	**34.45**	**0.973**	**31.09**	**0.949**
Proposed	**32.66**	**0.952**	**30.97**	**0.956**	**27.24**	**0.922**	**34.98**	**0.974**	**31.47**	**0.951**

Table 2. We consider a baseline model 'Base' that is trained only with the self-supervised constraints as proposed in SeLFVi[35]. We then successively enhance the 'Base' model with disocclusion handling, adaptive TD layer and the refinement block and compare the performance boost in each case.

this model, we repeat the current frame as three successive input frames of our proposed algorithm.

Tab. 1 details the quantitative comparisons of various algorithms against all 4 datasets: Hybrid, ViewSynth, TAMULF and Stanford. Our proposed reconstruction outperforms previous state-of-the-art techniques. We also notice that even our image-based model 'Proposed(image)' outperforms the previous image-based LF prediction techniques. We can also see clear distinction when we compare the images qualitatively in Fig. 6, especially when the EPI for the LF views are taken into account. We also validate our algorithm on monocular videos acquired from a commercial GoPro camera. While we show some results from GoPro dataset in Fig. 7, please refer to the supplementary material for more qualitative results.

Temporal consistency We evaluate and quantitatively compare the temporal consistency of the videos predicted from our proposed algorithm. For this, we first predict optical flow via [41] between all SAIs of successive ground-truth LF frames. We then compute the mean squared error between the current estimated LF and the previous LF warped to the current frame. We provide quantitative comparison in the supplementary material.

4.2 Ablation Study

Our proposed technique contains three key building blocks that enable us to work with monocular videos. Here, we evaluate the contribution of each of the three building blocks to the reconstruction quality. As shown in Tab. 2 we evaluate the effect of each block by successively adding the proposed blocks to the baseline model and quantitatively comparing the reconstructed LF videos. The baseline model can also be thought of as an extension of SeLFVi[35] to the case of monocular videos. Here, we utilize *only* the geometric, photometric and temporal consistency constraints proposed in SeLFVi. The LF synthesis network architecture $\mathcal{V}$ remains identical in all the models.

Disocclusion handling (Base vs Base+occ): Enforcing the disocclusion handling constraint helps the synthesis network to learn to fill in the disoccluded pixels in the estimated LF frames as shown in Fig. 7. Quantitatively, we also observe a boost of 0.9dB PSNR in comparison to the baseline model.

Adaptive TD layer (Base+occ vs Base+occ+adpt): Our modified adaptive TD layer can accurately represent the depth planes in the LF as can be seen from the EPI images in Fig. 8. Quantitatively, we get a significant performance boost of about 0.7dB PSNR.

Fig. 7. The model trained without disocclusion handling leads to a halo-like artifact around depth-edges in the SAIs of the frames. With the proposed disocclusion handling technique, the network learns to accurately fill-in the disoccluded pixels.

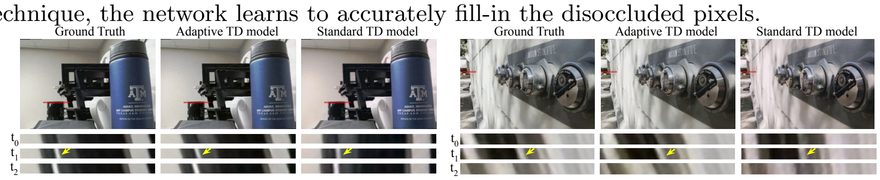

Fig. 8. As seen in the EPI images, the standard TD model is unable to represent the depth for the scene accurately compared to the proposed adaptive TD model. By separately determining the depth planes for each scene, adaptive TD model gives a more accurate reconstruction.

Supervised refinement block (Base+occ+adpt vs Proposed): Finally, we evaluate the effect of the novel refinement block that is trained with supervised loss on ground-truth LF frames. We observe an expected improvement in the reconstruction quality, showing a boost in PSNR of nearly 0.4dB. We also make qualitative comparison in Fig. 9, where we see that the refinement block provides more accurate SAIs around depth edges that are difficult to reconstruct with just self-supervised learning.

4.3 Variable baseline LF prediction

Supervised techniques using LF data from a single camera produce LF images with a fixed baseline. However, our proposed network reconstructs LF frames based on the input disparity map. By scaling the disparity map by a constant factor, we can scale the disparity values input to the network, leading to LF prediction with variable baselines. In Fig. 10 we demonstrate this with 4 different scale factor for disparity maps, $1\times$, $1.5\times$, $2\times$, $2.5\times$. Note that our algorithm allows us to generate SAIs with higher baseline than that of the ground truth frames from Lytro.

5 Discussion

Our proposed algorithm is largely a self-supervised technique except for the refinement block that is supervised using ground-truth LF *image* data. The refinement block uses a transformer module for angular attention. To the best of our knowledge this is also the first attempt to employ vision transformers to LF data. Note that our proposed algorithm outperforms previous state-of-the-art techniques even without the supervised refinement module. Another point to note is that, during inference, we do not have any information about the true baseline of the LF. We only have access to a relative depth map obtained from

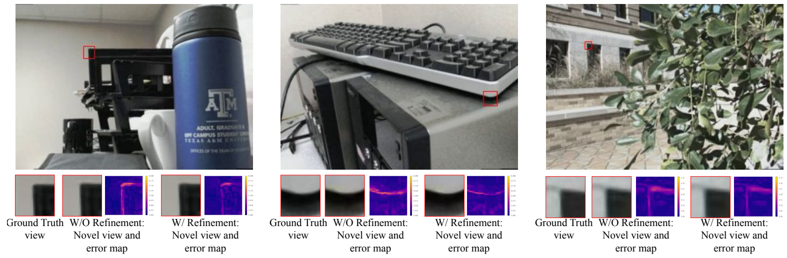

Fig. 9. The error map between the reconstructed and ground-truth shows that supervised refinement improves reconstruction at depth-edges. The refinement module corrects the baseline discrepancy The refinement block utilizes the spatial information in other SAIs through angular attention and optimizes the positioning of depth-edges correcting the baseline discrepancy between synthesized and ground truth LF.

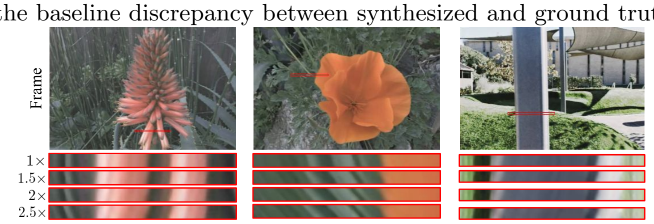

Fig. 10. With our proposed self-supervised technique, LF frames with variable baselines can be predicted by just scaling the input disparity map. We demonstrate this on 4 different scales $\{1, 1.5, 2, 2.5\}\times$. Notice the increasing slope in the EPI images from $1\times$ to $2.5\times$.

a single input image. Hence, it becomes difficult to accurately compare with the ground-truth. To solve this, we choose a scale and shift factor ($\{a, b\}$) such that the mean error between the computed disparity maps (from relative depth maps) and the true disparity maps (for a given dataset such as TAMULF) is minimum. Outside of comparison with ground truth, the true disparity map is not necessary and we can generate LF of multiple baselines as needed (Fig. 10).

6 Conclusion

We propose an algorithm for LF video reconstruction from just a monocular video input. Our baseline model utilizes the intermediate layered representation for LF and the self-supervised geometric, photometric and temporal constraints [35]. Additional modifications were proposed in this work that enabled the final model to reconstruct high-fidelity LF videos from monocular input. We propose a disoclussion handling technique that is required to fill-in disoccluded regions in the estimated LF. We also propose a adaptive TD representation that can adapt to each input scene based on the layer displacements predicted by the network. Finally, we introduce a novel supervised, transformer-based refinement block that can further refine the predicted LF. Along with superior reconstruction results, our model also enables prediction of LF frames with varying baselines. Overall, our proposed algorithm facilitates a monocular camera for applications like refocusing and novel view synthesis.

Acknowledgements This work was supported in part by Qualcomm Innovation Fellowship (QIF) India 2021.

References

1. Adelson, E.H., Bergen, J.R.: The plenoptic function and the elements of early vision. In: Computational Models of Visual Processing. pp. 3–20. MIT Press (1991) 4
2. Bae, K., Ivan, A., Nagahara, H., Park, I.K.: 5d light field synthesis from a monocular video. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 7157–7164. IEEE (2021) 2, 4
3. Bhat, S.F., Alhashim, I., Wonka, P.: Adabins: Depth estimation using adaptive bins. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4009–4018 (2021) 3, 5, 6, 9
4. Blocker, C.J., Chun, Y., Fessler, J.A.: Low-rank plus sparse tensor models for light-field reconstruction from focal stack data. In: 2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP). pp. 1–5. IEEE (2018) 4
5. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. Advances in Neural Information Processing Systems **33**, 9912–9924 (2020) 3, 8
6. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020) 3, 8
7. Dansereau, D.G., Girod, B., Wetzstein, G.: LiFF: Light field features in scale and depth. In: Computer Vision and Pattern Recognition (CVPR). IEEE (Jun 2019) 10
8. Delbracio, M., Kelly, D., Brown, M.S., Milanfar, P.: Mobile computational photography: A tour. arXiv preprint arXiv:2102.09000 (2021) 2
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 3, 8
10. Garg, R., Wadhwa, N., Ansari, S., Barron, J.T.: Learning single camera depth estimation using dual-pixels. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7628–7637 (2019) 6
11. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3828–3838 (2019) 8
12. Hajisharif, S., Miandji, E., Guillemot, C., Unger, J.: Single sensor compressive light field video camera. In: Computer Graphics Forum. vol. 39, pp. 463–474. Wiley Online Library (2020) 4
13. Huang, P.H., Matzen, K., Kopf, J., Ahuja, N., Huang, J.B.: Deepmvs: Learning multi-view stereopsis (2018) 4
14. Inagaki, Y., Kobayashi, Y., Takahashi, K., Fujii, T., Nagahara, H.: Learning to capture light fields through a coded aperture camera. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 418–434 (2018) 4
15. Ivan, A., Park, I.K., et al.: Synthesizing a 4d spatio-angular consistent light field from a single image. arXiv preprint arXiv:1903.12364 (2019) 4
16. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. arXiv preprint arXiv:1506.02025 (2015) 7
17. Jaiswal, A., Babu, A.R., Zadeh, M.Z., Banerjee, D., Makedon, F.: A survey on contrastive self-supervised learning. Technologies **9**(1), 2 (2021) 3, 8
18. Kalantari, N.K., Wang, T.C., Ramamoorthi, R.: Learning-based view synthesis for light field cameras. ACM Transactions on Graphics (TOG) **35**(6), 1–10 (2016) 2, 4, 10

19. Kim, D., Woo, S., Lee, J.Y., Kweon, I.S.: Deep video inpainting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5792–5801 (2019) 7
20. Kim, H.M., Kim, M.S., Lee, G.J., Jang, H.J., Song, Y.M.: Miniaturized 3d depth sensing-based smartphone light field camera. Sensors **20**(7), 2129 (2020) 2
21. Kobayashi, Y., Takahashi, K., Fujii, T.: From focal stacks to tensor display: A method for light field visualization without multi-view images. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2007–2011 (2017). https://doi.org/10.1109/ICASSP.2017.7952508 4
22. Li, Q., Kalantari, N.K.: Synthesizing light field from a single image with variable mpi and two network fusion. ACM Trans. Graph. **39**(6), 229–1 (2020) 2, 3, 4, 6, 10, 11
23. Lippmann, G.: Épreuves réversibles donnant la sensation du relief. J. Phys. Theor. Appl. **7**(1), 821–825 (1908). https://doi.org/10.1051/jphystap:019080070082100 4
24. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net (2019) 10
25. Lumentut, J.S., Kim, T.H., Ramamoorthi, R., Park, I.K.: Deep recurrent network for fast and full-resolution light field deblurring. IEEE Signal Processing Letters **26**(12), 1788–1792 (2019) 10
26. Maruyama, K., Inagaki, Y., Takahashi, K., Fujii, T., Nagahara, H.: A 3-d display pipeline from coded-aperture camera to tensor light-field display through cnn. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 1064–1068 (2019). https://doi.org/10.1109/ICIP.2019.8803741 4
27. Marwah, K., Wetzstein, G., Bando, Y., Raskar, R.: Compressive light field photography using overcomplete dictionaries and optimized projections. ACM Transactions on Graphics (TOG) **32**(4), 1–12 (2013) 4
28. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines (2019) 4
29. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017) 9
30. Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P.: Light field photography with a hand-held plenoptic camera. Ph.D. thesis, Stanford University (2005) 2, 4
31. Niklaus, S., Mai, L., Yang, J., Liu, F.: 3d ken burns effect from a single image. ACM Transactions on Graphics (ToG) **38**(6), 1–15 (2019) 10, 11
32. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019) 10
33. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12179–12188 (2021) 5, 10, 11
34. Sakai, K., Takahashi, K., Fujii, T., Nagahara, H.: Acquiring dynamic light fields through coded aperture camera. In: European Conference on Computer Vision. pp. 368–385. Springer (2020) 4

35. Shedligeri, P., Schiffers, F., Ghosh, S., Cossairt, O., Mitra, K.: Selfvi: Self-supervised light-field video reconstruction from stereo video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2491–2501 (2021) 2, 4, 5, 6, 9, 10, 12, 14
36. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.c.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. Advances in neural information processing systems **28** (2015) 7
37. Srinivasan, P.P., Tucker, R., Barron, J.T., Ramamoorthi, R., Ng, R., Snavely, N.: Pushing the boundaries of view extrapolation with multiplane images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 175–184 (2019) 4
38. Srinivasan, P.P., Wang, T., Sreelal, A., Ramamoorthi, R., Ng, R.: Learning to synthesize a 4d rgbd light field from a single image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2243–2251 (2017) 2, 4, 10, 11
39. Takahashi, K., Kobayashi, Y., Fujii, T.: From focal stack to tensor light-field display. IEEE Transactions on Image Processing **27**(9), 4571–4584 (2018). https://doi.org/10.1109/TIP.2018.2839263 4
40. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114. PMLR (2019) 9
41. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: European conference on computer vision. pp. 402–419. Springer (2020) 7, 12
42. Vadathya, A.K., Girish, S., Mitra, K.: A unified learning-based framework for light field reconstruction from coded projections. IEEE Transactions on Computational Imaging **6**, 304–316 (2019) 4
43. Veeraraghavan, A., Raskar, R., Agrawal, A., Mohan, A., Tumblin, J.: Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. ACM Trans. Graph. **26**(3), 69 (2007) 4
44. Wang, L., Shen, X., Zhang, J., Wang, O., Lin, Z., Hsieh, C., Kong, S., Lu, H.: Deeplens: Shallow depth of field from A single image. CoRR **abs/1810.08100** (2018) 11
45. Wang, T.C., Zhu, J.Y., Kalantari, N.K., Efros, A.A., Ramamoorthi, R.: Light field video capture using a learning-based hybrid imaging system. ACM Transactions on Graphics (TOG) **36**(4), 1–13 (2017) 2, 4, 10
46. Wang, Y., Liu, F., Wang, Z., Hou, G., Sun, Z., Tan, T.: End-to-end view synthesis for light field imaging with pseudo 4dcnn. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018) 4
47. Wetzstein, G., Lanman, D., Hirsch, M., Raskar, R.: Tensor displays: Compressive light field synthesis using multilayer displays with directional backlighting. ACM Trans. Graph. **31**(4) (Jul 2012). https://doi.org/10.1145/2185520.2185576 3, 4, 5, 6
48. Wilburn, B., Joshi, N., Vaish, V., Talvala, E.V., Antunez, E., Barth, A., Adams, A., Horowitz, M., Levoy, M.: High performance imaging using large camera arrays. ACM Trans. Graph. **24**(3), 765–776 (jul 2005). https://doi.org/10.1145/1073204.1073259 2
49. Wu, G., Zhao, M., Wang, L., Dai, Q., Chai, T., Liu, Y.: Light field reconstruction using deep convolutional network on epi. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6319–6327 (2017) 4
50. Xu, R., Li, X., Zhou, B., Loy, C.C.: Deep flow-guided video inpainting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3723–3732 (2019) 7

51. Yeung, H.W.F., Hou, J., Chen, J., Chung, Y.Y., Chen, X.: Fast light field reconstruction with deep coarse-to-fine modeling of spatial-angular clues. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018) 4
52. Zhang, Z., Liu, Y., Dai, Q.: Light field from micro-baseline image pair. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3800–3809 (2015) 2, 4
53. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. In: SIGGRAPH (2018) 4
54. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: European conference on computer vision. pp. 286–301. Springer (2016) 4