Supplementary Material for Synthesizing Light Field Video from Monocular Video

Shrisudhan Govindarajan[1], Prasan Shedligeri[1], Sarah[1], and Kaushik Mitra[1]

[1]Indian Institute of Technology Madras, India

A More Details Regarding Experimental Comparisons Performed

Our proposed self-supervised algorithm is designed to take a monocular video as input and output a light-field (LF) video sequence. We evaluate our algorithm against other state-of-the-art supervised monocular LF estimation algorithms [1, 2, 3]. For Li *et al.* [1], we use their publicly available implementation to make our comparisons. To obtain the complete LF video from [1], we have to reconstruct each frame of the video individually. Li *et al.* + Ranftl *et al.* represents a modified version of [1], where we input a depth estimate from DPT [4] instead of the original DeepLens [5] model. Since [1] is not trained on depth inputs from DPT [4], we finetune [1] on the TAMULF dataset with depth maps obtained from DPT [4]. For finetuning, we use AdamW optimizer for 5 epochs, with an initial learning rate of 2×10^{-5} and a weight decay of 0.001. While [2] is originally trained on a dataset of flower images (proposed in the same work), we finetune it on a larger and diverse TAMULF dataset from [1]. This is done because the test data has much more diverse inputs than the images in the original flowers dataset. The network is finetuned in Tensorflow using Adam optimizer for $80k$ iterations with a learning rate of 10^{-4}.

In Tab. 1 of main, we perform quantitative comparisons of various algorithms against 4 datasets: Hybrid, ViewSynth, TAMULF and Stanford. 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 [6, 7].

In case of Hybrid, we use the test sequences from the dataset. For ViewSynth datasets, we choose the synthesized video sequences from the default test set. For TAMULF dataset, we randomly chose 84 samples from 1084 light field frames, and these frames are used to synthesize videos which are used for testing the algorithms. The Stanford dataset contains 4211 light fields organised into 30 categories. Each scene is captured from $3, 4$ or 5 different camera poses, over a total of 850 scenes. We randomly select 113 light fields frames from *Tree* category to synthesize videos which was used as the test set.

B Details of our Proposed Network architecture

Light field synthesis network, $\mathcal{V}$ The synthesis network $\mathcal{V}$ is a long short term memory (LSTM) based recurrent neural network consisting of a Efficient-Net encoder [8] and a convolutional decoder with skip connections as shown in Fig. A1. The LSTM layer follows the Efficient-Net encoder and the cell output from the LSTM layer is fed as input to the decoder network. The decoder network consists of 4 upsampling blocks. In the upsampling block, the feature maps are first doubled in size spatially using bilinear interpolation. The upsampled feature map is then fed to a series of two convolutional layers of filter size 3×3, which is then followed by batch normalization. The output of the final upsampling block is then input to a final convolutional layer which outputs 36 RGB (108) channels. These 108 channels correspond to the $N = 3$ layers and $R = 12$ rank of the low-rank LF representation $\mathcal{F}$.

The displacements $D = \{D_1, D_2, D_3\}$ for the Adaptive TD layer are predicted from mini-vision transformer (m-VIT)[9] network that takes as input $\{I_{t-1}, I_t, I_{t+1}, d_t\}$. The m-VIT network used in our work is exactly the same as the one proposed in [9]. Except that in our implementation we stack successive frames and disparity map as input (10 channels) and provide it as input to m-VIT. The output of this network is a sequence of values $D = \{D_1, D_2, D_3\}$ which is used to predict the LF frame.

Supervised residual refinement module Vision Transformers(ViT) [10] form the backbone of our proposed refinement module. We divide the predicted LF frame $\hat{\mathbf{L}}_t$ into non-overlapping patches, each of size $p \times p$ ($= 32 \times 32$). A shallow ResNet-based encoder (see Fig. A2) extracts features independently from each of the $U^2 (= 49)$ patches. The encoder contains 12 bottleneck blocks [11] with max-pooling operation carried out at the first of every three blocks as shown in Fig. A2. The obtained feature embeddings are input to two transformer layers which applied multi-headed self-attention (MHSA) to these tokens. Here, $P (= 6)$ is the number of non-overlapping patches cropped from each angular view. These P tokens are stacked horizontally and vertically following the order of

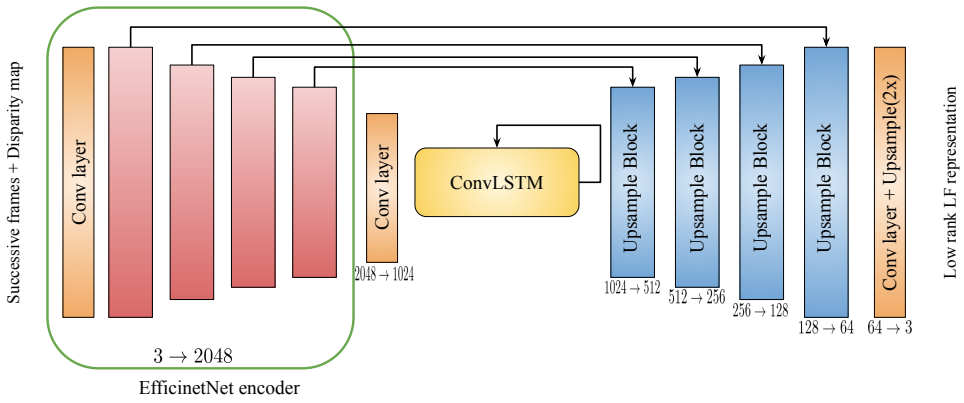

Figure A1: We show the detailed network architecture of the LF synthesis network $\mathcal{V}$ that predicts the low-rank representation $\mathcal{F}$. For estimation of the low-rank LF representation, we use a encoder-decoder based network with ConvLSTM module used for learning temporal information. The encoder block follows the Efficient-Net[8] architecture and the decoder consists of bilinear interpolation operation followed by convolution and batch-normalization operation.

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 shallow network consists of 4 upsample blocks where each block has the exact same configuration followed in LF synthesis network, $\mathcal{V}$.

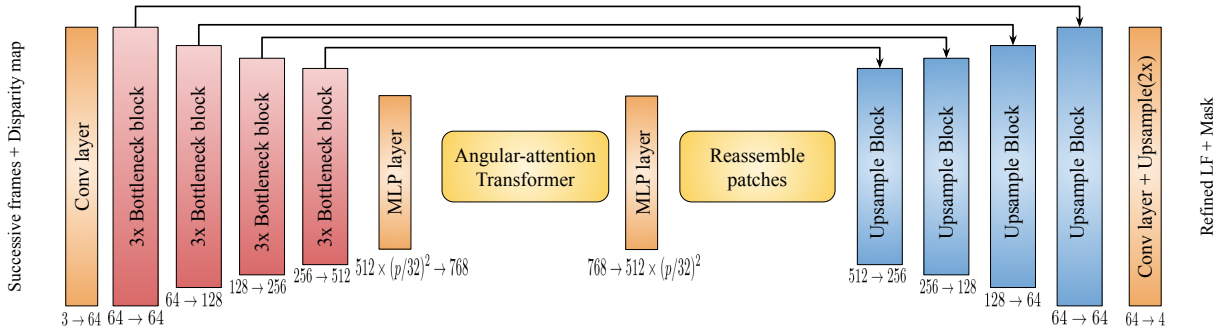

Figure A2: We show the detailed architecture of the supervised Refinement module in our proposed algorithm (see Fig. 5 of main). It takes as input the spatial patches cropped from each LF sub-aperture image (SAI) and produces a feature map for independently for each patch. The 'Bottleneck block' used here is identical to the ResNet-based bottleneck block proposed in [11]. The output features from encoder are then flattened and passed through a MLP layer to get the feature embeddings. The transformer layer performs multi-headed self-attention (MHSA) on the embeddings and outputs tokens which are then reassembled spatially to form larger feature maps (see Sec. 3.5 of main). These feature maps are input to the decoder block identical to the decoder block in $\mathcal{V}$. The refined LF and mask are obtained as outputs from the decoder block.

C Ablation on pre-trained optical flow and depth estimation networks

Our proposed self-supervised algorithm aims to synthesize LF video sequence from monocular video by enforcing geometric consistency via relative depth map, temporal consistency and dis-occlusion handling via optical flow. For this purpose, we use a pre-trained DPT[4] network to estimate the relative depth map from monocular frame and pre-trained RAFT[12] network to estimate the optical flow between successive input video frames. Since the performance of our proposed methods depends highly on the pre-trained depth and optical flow network, we compare the effect on performance by replacing DPT and RAFT models with DeepLens[5] and LiteFlowNet[13] models for relative depth and optical flow estimation. As shown in Tab. A1, when the DPT and/or RAFT models are replaced with DeepLens and LiteFlowNet respectively, we observe a decline in the performance of the LF synthesis network, $\mathcal{V}$, as evidenced by the reduction in PSNR and SSIM scores. From this we conclude that the

Model	Depth	Flow	Refine	Average	
				PSNR	SSIM
V1	DPT	RAFT	✗	31.09	0.949
V2	DPT	RAFT	✔	31.47	0.951
V3	Deeplens	RAFT	✗	30.53	0.945
V4	Deeplens	RAFT	✔	30.76	0.946
V5	DPT	LiteFlow	✗	29.60	0.949
V6	DPT	LiteFlow	✔	31.27	0.950
V7	Deeplens	LiteFlow	✗	29.01	0.941
V8	Deeplens	LiteFlow	✔	29.59	0.941

Table A1: **Effect of depth and optical flow accuracy on our proposed method:** Accurate depth and optical flow estimates are necessary to enforce the geometric and temporal consistency on the predicted LF video. Estimated optical flow is also used to handle disoccluded pixels in the LF. Hence, we compare the effect of replacing these state-of-the-art models (DPT and RAFT) with slightly less accurate models (DeepLens[5] and LiteFlowNet[13]). We observe that using less accurate depth and optical flow estimates causes our reconstruction results to degrade. And this degradation is more pronounced when we don't use the proposed supervised refinement module. This also points to the conclusion that our proposed model is indeed utilizing the information from the depth and optical flow estimates. Also, our proposed supervised refinement module tries to correct these errors using the available ground-truth data.

proposed self-supervised algorithm indeed utilizes the information from the relative depth map and optical flow map and the quality of these estimates affect the performance of the LF synthesis network, $\mathcal{V}$. Also, our proposed supervised refinement module tries to correct these errors using the available ground-truth data, irrespective of the relative depth and optical flow networks, resulting in similar quality LF estimates after refinement.

D Temporal consistency of synthesized LF video sequence

Our proposed algorithm aims to reconstruct LF *video* sequences where temporal consistency is a crucial factor. We evaluate and quantitatively compare the temporal consistency of the videos predicted from our proposed algorithm. For evaluating temporal consistency between successive predicted LF frames, we first predict optical flow via [12] between all SAIs of successive ground-truth LF frames, i.e., $\mathbf{L}_t$, $\mathbf{L}_{t-1}$. The current estimated LF is then warped to the previous LF frame using the estimated ground-truth optical flow. We then compute the mean squared error between the previous estimated LF and the current LF warped to the previous LF. The warping error is used as a measure of temporal stability between successive predicted LF frames. We calculate the temporal stability function (lower is better) as

$$\mathcal{E}_{temp}(\hat{\mathbf{L}}_t; \mathbf{L}_t, \mathbf{L}_{t-1}) = \sum_{\mathbf{u}} \|\mathcal{W}\left(\hat{\mathbf{L}}_t\left(\mathbf{u}\right); \mathcal{O}(\mathbf{L}_t\left(\mathbf{u}\right), \mathbf{L}_{t-1}\left(\mathbf{u}\right))\right) - \mathbf{L}_{t-1}\|_2 \tag{A1}$$

To estimate the temporal stability of the network for a video, the $\mathcal{E}_{temp}$ function is then averaged over the entire video. In table Tab. A2, we compare two of our models, 'Base+occ+adpt' (without refinement block) and 'Proposed' (with refinement block), with previous learning-based techniques. We observe that our proposed algorithm performs significantly better than previous state-of-the-art techniques for LF estimation. Although the 'Proposed' model is trained on images without the temporal consistency constraint, its performance does not degrade compared to 'Base+occ+adpt' model which is trained on monocular videos with explicit temporal consistency loss.

E Identified depth planes for adaptive tensor-display model

We propose an adaptive tensor-display based low rank representation for estimating LF from the given input frames. Here, we showcase how the distance values $D = \{D_1, D_2, D_3\}$ adapt to the varying input depth maps, thereby providing superior reconstruction results than that of the standard tensor-display based low-rank representation. In Fig. A3, we show various LF images predicted by different scaled versions of the disparity map. Specifically, we scale the input disparity map by the factors 1 and 2 and predict the corresponding LF. We observe that the predicted depth planes adapt to the input disparity maps providing superior reconstruction results.

Algorithm	Hybrid	ViewSynth	TAMULF	Stanford	Average
Niklaus	0.357	0.070	0.219	0.065	0.177
Srinivasan	0.195	0.020	0.070	0.014	0.075
Li	0.108	0.019	0.034	0.009	0.043
Li+Ranftl	0.108	**0.016**	0.033	0.008	0.042
Base+occ+adpt	**0.103**	0.017	**0.028**	**0.006**	**0.038**
Proposed	**0.102**	**0.016**	**0.027**	**0.006**	**0.038**

Table A2: **Evaluating temporal consistency:** We quantitatively compare the temporal consistency of the predicted LF videos through checking the consistency with the optical flow. Our proposed algorithm performs significantly better than previous state-of-the-art techniques for LF estimation. Although the refinement block of the 'Proposed' model is trained on images without the temporal consistency constraint, its performance does not degrade compared to 'Base+occ+adpt' model which is trained on monocular videos with explicit temporal consistency loss. **Blue** and **green** represent the top-two performing algorithm in each column.

F Application to video refocusing

LF have been popular because they provide a very intuitive way of doing post-capture focus control. The amount of defocus that can be achieved depends on the baseline of the LF. As demonstrated in Sec. 4.3, our technique is not limited to a single baseline. Unlike previous LF prediction techniques, a single model can output LF frames with multiple baselines. And this can be controlled by simply increasing/decreasing the scale factor used to convert a relative depth map to disparity map. As shown in Fig. A4, this can be used to control the level of blur in the defocused region. In a typical LF camera, post-capture aperture control can be used to only reduce the blur size from a maximum position. Here, by predicting a LF with a larger baseline, we can also increase the defocus blur.

References

[1] Qinbo Li and Nima Khademi Kalantari. Synthesizing light field from a single image with variable mpi and two network fusion. *ACM Trans. Graph.*, 39(6):229–1, 2020.

[2] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 2243–2251, 2017.

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

[4] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[5] Lijun Wang, Xiaohui Shen, Jianming Zhang, Oliver Wang, Zhe Lin, Chih-Yao Hsieh, Sarah Kong, and Huchuan Lu. Deeplens: Shallow depth of field from A single image. *CoRR*, abs/1810.08100, 2018.

[6] Prasan Shedligeri, Florian Schiffers, Sushobhan Ghosh, Oliver Cossairt, and Kaushik Mitra. Selfvi: Self-supervised light-field video reconstruction from stereo video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2491–2501, 2021.

[7] Jonathan Samuel Lumentut, Tae Hyun Kim, Ravi Ramamoorthi, and In Kyu Park. Deep recurrent network for fast and full-resolution light field deblurring. *IEEE Signal Processing Letters*, 26(12):1788–1792, 2019.

[8] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[9] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4009–4018, 2021.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[12] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020.

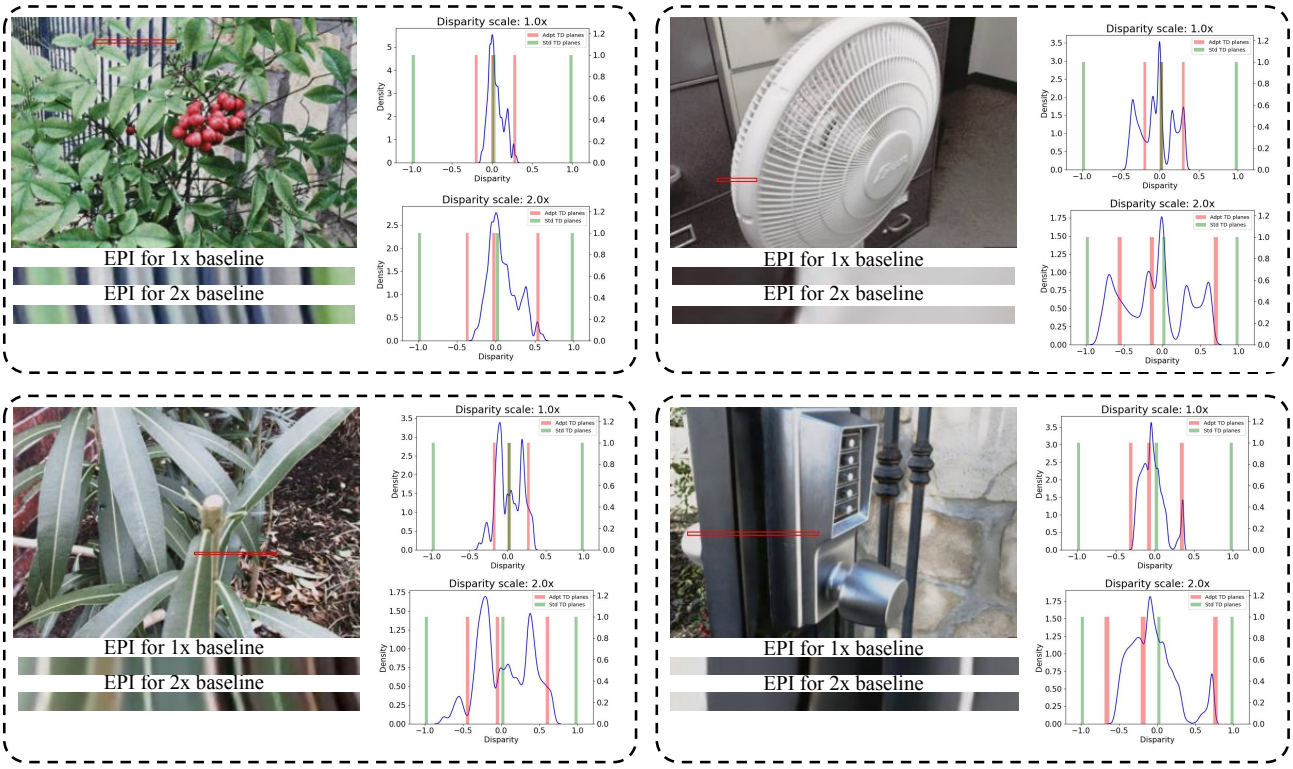

Figure A3: **Adaptive vs. vanilla tensor-display representation for LFs:** We show four different samples with EPIs for LF predicted by 1× and 2× scaled disparity inputs. Besides each figure we also show the distribution of disparity values in the 1× and 2× scaled disparity maps (shown as blue curve). In these graphs we also represent the disparity planes in the standard and adaptive TD models with green and orange bars respectively. We observe that the green bars remain constant (at $-1, 0, +1$) even when the disparity gets scaled. While the predicted orange bars adapt to the input scaled disparity maps thereby providing a more accurate representation of the LF. We also notice that the 3 orange bars are not necessarily uniformly distant from each other. Refer to supplementary video file for video results.

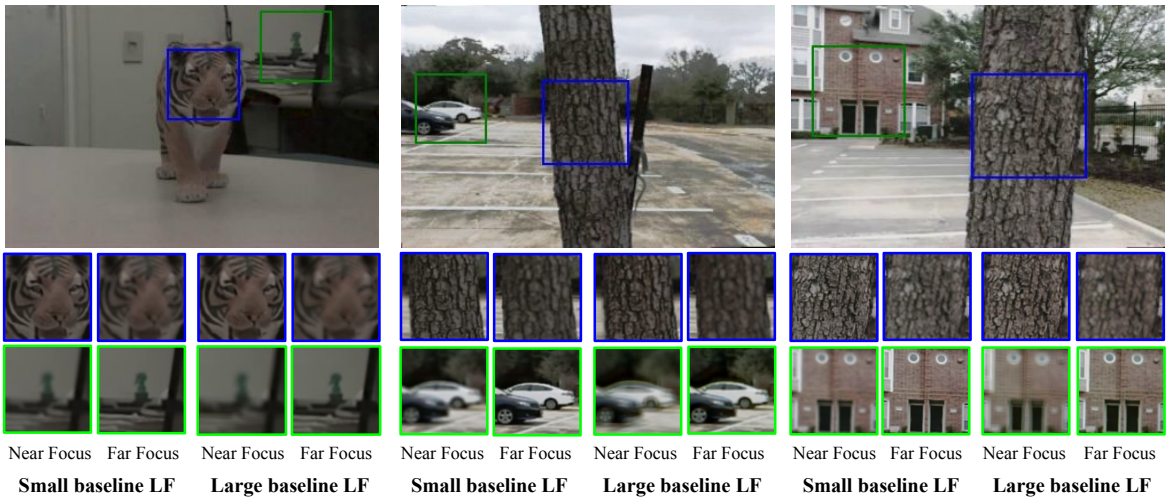

Figure A4: As our model can output LFs with varying baselines, we can demonstrate refocusing effects with varying blur sizes. As can be seen, the blur size in a large baseline LF is bigger than the one in a small baseline LF.

[13] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. LiteFlowNet: A Lightweight Convolutional Neural Network for Optical Flow Estimation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8981–8989, 2018.