LFVS-Mamba: State-Space Model for Light Field View Synthesis

Muhammad Zubair, Paulo Nunes, Caroline Conti, and Luís Ducla Soares

Instituto de Telecomunicações, Instituto Universitário de Lisboa (ISCTE-IUL) Lisbon, Portugal

***Abstract*—Light Field View Synthesis (LFVS) methods using Convolutional Neural Networks (CNNs) and Vision Transformers (VTs) have been extensively studied: CNNs excel at learning local spatial features via hierarchical receptive fields but cannot capture long-range global dependencies, while VTs inherently model global context through self-attention at the cost of quadratic computation and memory complexity. To address these issues, we propose LFVS-Mamba, which integrates a State-Space Module (SSM) with a Selective Scanning Mechanism to efficiently capture long-range dependencies. LFVS-Mamba processes 2D slices of the 4D LF to fully exploit spatial context, complementary angular information, and depth cues. The LFVS-Mamba comprises three modules to progressively synthesize dense LFs: (i) Shallow Feature Extraction (SFE), (ii) Spatial-Angular Depth Feature Extraction (SADFE), and (iii) Angular Upsampling (AU). Experimental results on standard LF benchmarks demonstrate that LFVS-Mamba consistently outperforms existing methods.**

Keywords—light field, view synthesis, angular consistency, state space model, cross-scanning

I. INTRODUCTION

A Light Field (LF) records both intensity and angular information of light rays simultaneously, offering a rich 3D representation of a scene. The angular information significantly broadens the scope of potential applications, including 3D scene reconstruction [1], [2], saliency detection [3], depth estimation [4], and refocusing [5]. However, acquiring dense LFs is a challenging task due to hardware limitations and trade-offs in conventional capturing systems.

Early LF acquisition through camera arrays [6] captures LF images in a single shot while computer-controlled gantries [7] capture multiple shots in a time-sequential manner. The former method is expensive due to a complex multi-camera setup. The latter method is restricted to static scenes due to its time-sequential capturing process. The introduction of plenoptic cameras, such as Lytro [8] and RayTrix [9], marked a significant advance in LF imaging, enabling direct capture of LF data for a wide range of applications. Despite this, plenoptic cameras suffer from limited sensor capabilities and an inherent trade-off between angular and spatial resolution, which constrains their performance.

Instead of acquiring dense LFs through camera devices, ongoing advances in neural network architectures have opened new avenues for LF View Synthesis (LFVS). These data-driven methods synthesize dense LFs from sparse input views, reducing acquisition cost and complexity. Convolutional Neural Networks (CNNs) have been widely adopted due to their efficient local feature extraction. However, CNNs are limited in modeling long-range dependencies, as they primarily rely on localized receptive fields [10], [11]. In contrast, Vision Transformers (VTs) mitigate this issue through the self-attention mechanism to capture global interactions between image tokens, but they incur quadratic computational costs [12], [13], [14]. Consequently, there is growing interest in alternative architectures that strike a better balance between efficiency and representational power, motivating the exploration of lightweight and attention-inspired algorithms, as State-Space Models (SSMs) [15], for LFVS.

The literature on LFVS can be categorized as being either depth-dependent or non-depth-dependent. Depth-dependent methods warp sparse input views to target angular positions using estimated depth maps. Jin *et al.* [11] use two modules for LFVS: (i) a depth-based warping module that explicitly models scene geometry using convolution layers with large receptive fields and warps the source views to target views based on the estimated depth maps, and (ii) a refinement blending module exploits spatial-angular relations among the warped and source views to obtain the dense LFs. Liu *et al.* [16] also propose a two-module approach: (i) a multi-representation view reconstruction module that extracts LF features from multiple representations for intermediate view synthesis, and (ii) a geometry-assisted refinement module that refines these views using bidirectional horizontal and vertical view stacks. Zubair *et al*. [17] extended Liu *et al*.'s work [16] by proposing the use of deformable convolution for dense LFVS.

Non-depth-dependent methods exploit local LF structures, such as neighboring views, Epipolar Plane Images (EPIs) or EPI volumes to synthesize dense LFs. Wang *et al.* [18] first convert the Sub-Aperture Images (SAIs) into Micro-Images (MIs) and EPIs. Disentangling groups and blocks then extract spatial, angular, and EPI features, which are fused in multiple stages to synthesize dense LFs. Liu *et al.* [19] employ a 3D-UNet to capture multi-scale spatial-angular correlations and reshape the extracted features into MI features representation to synthesize dense LFs.

To the best of our knowledge, this is the first time in the literature that an SSM, dubbed Mamba, is proposed for LFVS. Unlike in the spatial super-resolution method in [20] that uses bi-directional subspace scanning along a single axis to enhance global receptive fields horizontally and vertically, the proposed LFVS-Mamba performs Cross-Scanning (CS) along four directions: left-to-right, top-to-bottom, right-to-left, and bottom-to-top.

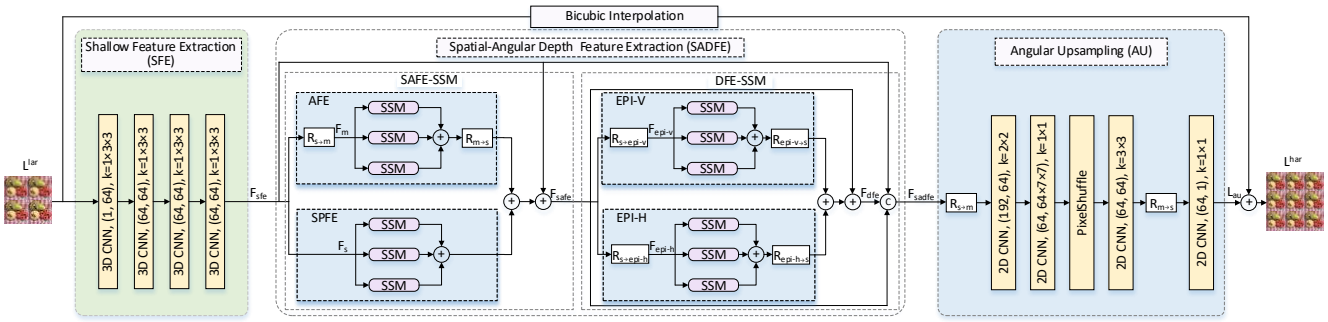


Fig. 1. LFVS Mamba architecture.

This CS strategy, based on axis-aligned scans, enables the model to encode richer angular dependencies by capturing information from multiple spatial traversal paths. The main contributions of this paper can be summarized as follows:

- We propose LFVS-Mamba to jointly capture spatial context, complementary angular information, and geometric depth cues from 2D slices via a CS mechanism.
- We design a channel-aware residual block that integrates dual state-space pathways to learn global contextual and channel-specific features, enhancing LFVS-Mamba's capacity for accurate dense LFVS.

II. Proposed Method

A. Preliminaries

The recent advances in Structured State-Space Sequence models (S4) [21] are largely inspired by continuous Linear Time-Invariant (LTI) systems, which map a 1D input, $x(t) \in \mathbb{R}$, to an output $y(t) \in \mathbb{R}$, , via an implicit latent state, $h(t) \in \mathbb{R}^N$, where N denotes the dimensionality of the latent state vector. This system follows the Linear Ordinary Differential Equation (ODE):

$$\begin{aligned} h'(t) &= A \cdot h(t) + B \cdot x(t), \\ y(t) &= C \cdot h(t), \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ is the state matrix, $B \in \mathbb{R}^{N\times 1}$ is the input matrix and $C \in \mathbb{R}^{1\times N}$ is the output matrix.

Eq. (1) is discretized to integrate the continuous SSM into a deep learning framework. A learnable step size, Δ, converts "continuous" parameters A, B and C into "discrete" ones $\bar{A}$, $\bar{B}$ and $\bar{C}$ via the Zero-Order-Hold (ZOH) rule:

$$\begin{aligned} \bar{A} &= exp(\Delta A), \\ \bar{B} &= (\Delta A)^{-1} \cdot (exp(\Delta A) - I) \cdot \Delta B, \\ \bar{C} &= C. \end{aligned} \tag{2}$$

The discretized SSM can be computed either by linear recursion or a global convolution with kernel $\bar{K}$:

$$\begin{aligned} h_k &= \bar{A} \cdot h_{k-1} + \bar{B} \cdot x_k, \quad k = 0, \cdots, L-1 \\ y_k &= \bar{C} \cdot h_k, \end{aligned} \tag{3}$$

$$\begin{aligned} \bar{K} &= (\bar{C} \cdot \bar{B}, \bar{C} \cdot \bar{A} \cdot \bar{B}, \cdots, \bar{C} \cdot \bar{A}^{L-1} \cdot \bar{B}), \\ y &= x * \bar{K}, \end{aligned} \tag{4}$$

where L is the sequence length, '$*$' denotes the convolution operation, $\bar{K} \in \mathbb{R}^L$ is the structured kernel, and y is the output.

B. Overall Architecture

A LF is a 4D function $L(u, v, x, y)$, where (u, v) and (x, y) are the angular and spatial coordinates, respectively. The objective of the proposed LFVS-Mamba is to synthesize dense LFs, $L^{har}(u, v, x, y)$, with higher angular resolution of $U \times V$, and a spatial resolution of $X \times Y$, from a sparse set of input views, $L^{lar}(u', v', x, y)$, with lower angular resolution $U' \times V'$. This process is formulated as:

$$L^{har}(u, v, x, y) = f(L^{lar}(u', v', x, y), \theta) \tag{5}$$

where $f(\cdot)$ denotes the synthesis mapping function and θ the learnable parameters. Following [11] and [18], we convert LF images from RGB to YCbCr and process only the luminance channel of each sparse input view in L^{lar}. The architecture of LFVS-Mamba (see Fig. 1) consists of three modules: (i) Shallow Feature Extraction (SFE), (ii) Spatial-Angular-Depth Feature Extraction (SADFE), and (iii) Angular Upsampling (AU). The SADFE module integrates several SSM blocks, described separately at the end of this section.

1) Shallow Feature Extraction (SFE)

The SFE module extracts spatial features from each view independently while preserving the angular structure. Given the sparse input tensor $L^{lar} \in \mathbb{R}^{C\times U'\times V'\times X\times Y}$, where $C = 1$ denotes the single luminance channel, SFE first applies a $1 \times 3 \times 3$ 3D convolution to project the luminance channel into a higher-dimensional feature space (i.e., $Ch = 64$). This embedding is then refined with three successive $1 \times 3 \times 3$ 3D convolution layers, each followed by a Leaky ReLU activation function. Each 3D convolution uses a kernel of depth 1, confining operations to each view's spatial domain and preventing inter-view mixing, thus producing a feature tensor $F_{sfe} \in \mathbb{R}^{U'\times V'\times Ch\times X\times Y}$.

2) Spatial-Angular Depth Feature Extraction (SADFE)

The SADFE module extracts spatial context, complementary angular information, and geometric depth cues by processing F_{sfe} using four different representations: $F_s \in \mathbb{R}^{U'\times V'\times Ch\times X\times Y}$, $F_m \in \mathbb{R}^{X\times Y\times Ch\times U'\times V'}$, $F_{epi\_h} \in \mathbb{R}^{X\times V'\times Ch\times U'\times Y}$, and $F_{epi\_v} \in \mathbb{R}^{Y\times U'\times Ch\times V'\times X}$, which denote spatial, angular, horizontal EPI (EPI-H) and vertical EPI (EPI-V) feature slices. Notice that $F_s = F_{sfe}$.

The proposed SADFE module is divided into two sub-modules: (i) Spatial-Angular Feature Extraction (SAFE-SSM), and (ii) Depth Feature Extraction (DFE-SSM). SAFE-SSM uses two networks: (i) Spatial Feature Extraction (SPFE), and (ii) Angular Feature Extraction (AFE), to extract feature slices from F_s (organized as SAIs) and F_m (organized as MIs), respectively. DFE-SSM uses EPI-H and EPI-V networks to extract depth feature slices from F_{epi_h} and F_{epi_v}. Each of these networks consists of three SSM blocks.

The AFE features are then added to the SPFE features and

to a skip connection F_{sfe}, to obtain F_{safe}, which is then passed on to DFE-SSM. Within DFE-SSM, F_{safe} is then processed using EPI representations to obtain the EPI-H and EPI-V features, which are added to a skip connection, F_{safe}, to produce the depth cues features, F_{dfe}. Finally, the output of SADFE is the concatenation along the channel dimension of three sets of features:

$$F_{sadfe} = C\left(F_{sfe}, F_{safe}, F_{dfe}\right) \tag{6}$$

Whenever features are combined, they are first rearranged to a common representation (i.e., the SAI representation).

3) Angular Upsampling (AU)

The AU module upsamples the F_{sadfe} features, rearranged as MI representation, to the target angular resolution through the following operations. First, a 2×2 convolution produces angularly downsampled features, followed by a 1×1 convolution to increase the channel dimensionality. PixelShuffle is used to rearrange the sub-pixel information into a denser 7×7 MI grid. A 3×3 convolution is used to refine the upsampled features, which are rearranged into a SAI feature representation. A final 1×1 convolution reduces the channel dimension, Ch, to a single channel per view, yielding the LF residual L_{au}. Finally, by adding it to an estimate obtained through bicubic interpolation, we obtain the dense LF, $L^{har}(u, v, x, y)$.

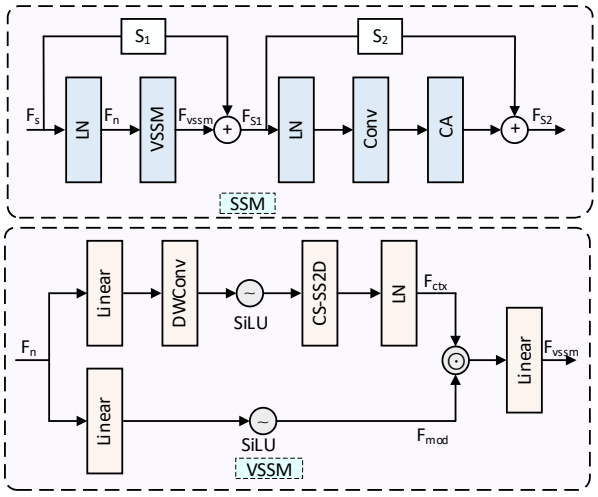


Fig. 2. SSM and VSSM architecture.

4) State-Space Module (SSM)

Since both SAFE-SSM and DFE-SSM sub-modules share the SSM blocks, we discuss only the SPFE network here. As seen in Fig. 2, the SSM comprises two stages. In Stage 1, spatial feature representation, F_s, is first normalized by LayerNorm (LN) to yield, F_n, and then processed by the Visual State-Space Module block (VSSM) [15] (see Eq. (11)) to capture the long term global spatial context. A learnable skip connection with scale factor $S_1 \in \mathbb{R}^{Ch}$ is used to regulate the spatial information between input and output, using element-wise multiplication:

$$F_{s1} = F_{vssm} + S_1 \cdot F_s \tag{7}$$

F_{vssm} captures long-range dependencies by processing features as a 1D sequence through Cross-Scan 2D Selective Scan (CS-SS2D), but this disrupts the adjacency of neighboring pixels and degrades local detail. To restore locality, in Stage 2, we first apply LN to F_{s1}, then use a bottleneck convolution (Conv) that reduces channels by a factor of $\gamma = 30$ (γ compression ratio) before restoring the original dimensionality. The Channel Attention (CA) block reweights feature maps to emphasize informative channels. Another learnable skip connection, scaled by $S_2 \in \mathbb{R}^{Ch}$, is employed to produce the final output:

$$F_{s2} = CA\left(Conv\left(LN(F_{s1})\right)\right) + S_2 \cdot F_{s1} \tag{8}$$

The normalized spatial features, F_n, are processed through dual state paths in VSSM. In the first path, a linear layer expands the number of channels to λCh, where λ denotes a predefined expansion ratio. This expanded representation is refined by a depthwise 2D convolution (DWConv) followed by SiLU and then processed by CS-SS2D, where the feature maps are split into four 1D sequences (left-to-right, top-to-bottom, right-to-left, bottom-to-top), each sequence is passed through the Selective Scan mechanism (S6) [22] to capture long-range context, and the sequences are merged back into a 2D map that is finally normalized with LN:

$$F_{ctx} = LN\left(CS\text{-}SS2D\left(SiLU\left(DWConv\left(Lin(F_n)\right)\right)\right)\right) \tag{9}$$

In the second path, F_n is projected through another linear layer to produce the $\lambda \cdot Ch$, followed by SiLU activation:

$$F_{mod} = SiLU\left(Lin(F_n)\right) \tag{10}$$

The two path outputs, spatial context, F_{ctx}, and modulation, F_{mod}, are fused via a Hadamard product, '$\odot$'. Finally, a linear layer compresses the features back to its original dimension Ch, resulting in an output tensor F_{vssm}:

$$F_{vssm} = Lin(F_{ctx} \odot F_{mod}) \tag{11}$$

III. Results and Discussion

LFVS is trained for 70 epochs independently on synthetic and real-world datasets using non-overlapping 64×64 patches, L1 loss, and the Adam optimizer [23] with batch size 1. Two networks are independently trained on these datasets. The synthetic network is trained on 20 scenes from the HCI-new dataset [24] and evaluated on HCI-new [24] and the HCI-old dataset [25]. The real-world network is trained on 100 scenes of Kalantari *et al*. [26] and the Stanford Lytro Archive [27], and evaluated on three real-world datasets: UCSD [26], Occlusion [27],and Reflective [27].

Table I and Table II report the average PSNR and SSIM values on synthetic and real-world datasets (best values are in bold, and the second-best values are underlined). On the synthetic datasets, LFVS-Mamba achieves the highest average PSNR of 38.97 dB, outperforming LFASR-GEO [11] by 4.96 dB, GA-MRVR [16] by 1.42 dB, Deformable-LFVS [17] by 1.08 dB, LF-EASR [19] by 1.10 dB, and Distg-ASR [18] by 0.83 dB. On real-world datasets, it attains an average PSNR of 41.09 dB, surpassing LFASR-GEO [11] by 3.27 dB, GA-MRVR [16] by 0.76 dB, Deformable-LFVS [17] by 0.48 dB, LF-EASR [19] by 0.68 dB, and Distg-ASR [18] by 0.42 dB. SSIM results are similar in both cases.

Fig. 3 and Fig. 4 show a visual comparison of LFVS-

Mamba with state-of-the-art methods. Error maps are computed from the synthesized central views and their Ground Truth (GT). Red boxes highlight a region of interest in each error map. Fig. 3's red-boxes show that LFVS-Mamba has fewer pixels with high synthesis error, producing clearer texture in the synthesized view. Fig. 4 shows a similar behavior for a real-word scene.

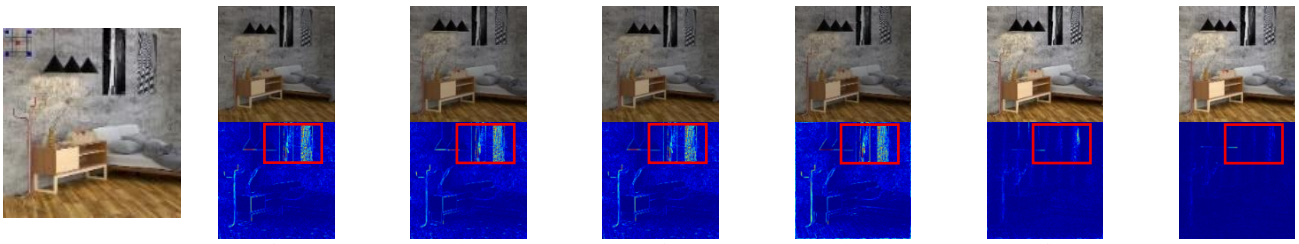
To evaluate angular consistency, we extract horizontal and vertical EPIs from the synthesized and GT LFs, and visualize the differences using error maps, EPI slices, and zoomed-in regions. Fig. 5 shows that LFVS-Mamba preserves parallax structures, while competing methods exhibit noticeable artifacts and parallax distortions.

TABLE I. QUANTITATIVE COMPARISON ON SYNTHETIC DATASETS FOR THE TASK OF $2 \times 2 \rightarrow 7 \times 7$

Datasets	**LFASR-GEO** [11]	**GA-MRVR** [16]	**Deformable-LFVS** [17]	**LF-EASR** [19]	**Distg-ASR** [18]	**LFVS-Mamba**
HCI-new	32.29/0.911	34.28/0.936	34.52/0.936	34.27/0.949	34.16/0.969	**35.82 /0.972**
HCI-old	35.73/0.937	40.82/0.953	41.25/0.974	41.47/0.970	**42.13**/0.974	42.11/**0.975**
Average	34.01/0.924	37.55/0.944	37.89/0.955	37.87/0.960	38.14/ 0.972	**38.97/0.974**

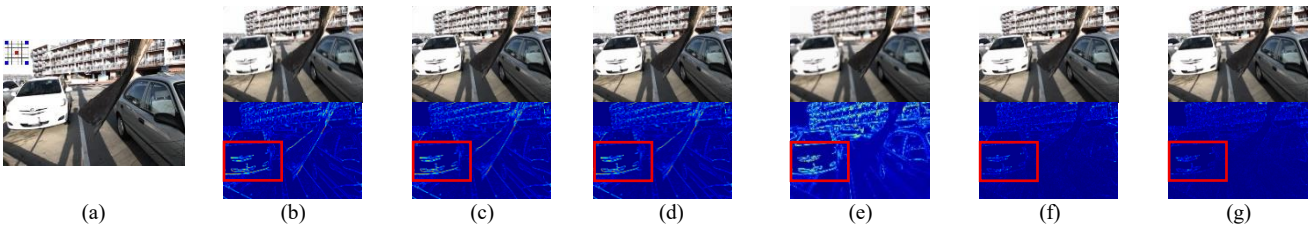
TABLE II. QUANTITATIVE COMPARISON ON REAL-WORLD DATASETS FOR THE TASK OF $2 \times 2 \rightarrow 7 \times 7$

Datasets	**LFASR-GEO** [11]	**GA-MRVR** [16]	**Deformable-LFVS** [17]	**LF-EASR** [19]	**Distg-ASR** [18]	**LFVS-Mamba**
UCSD	40.78/0.982	42.91/0.987	43.19/0.987	43.15/0.980	43.60/0.986	**43.86/0.988**
Occlusions	36.43/0.971	39.04/0.981	39.45/**0.984**	39.18/0.979	39.40/0.982	**39.88/0.984**
Reflective	36.25/0.945	39.04/0.962	39.19/0.963	38.91/0.958	39.02/0.960	**39.52/0.964**
Average	37.82/0.966	40.33/0.976	40.61/0.978	40.41/0.972	40.67/0.976	**41.09/0.979**



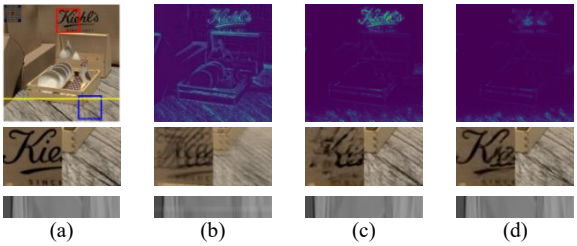
(a) (b) (c) (d) (e) (f) (g)

Fig. 3 . Qualitative results on a synthetic scene (HCI-new, Bedroom [24]): (a) GT; (b) LFASR-GEO [11]; (c) GA-MRVR [16]; (d) Deformable-LFVS [17]; (e) LF-EASR [19]; (f) Distg-ASR [18]; (g) LFVS-Mamba.



(a) (b) (c) (d) (e) (f) (g)

Fig. 4. Qualitative results on a real-world scene (UCSD, Car [26]): (a) GT; (b) LFASR-GEO [11]; (c) GA-MRVR [16]; (d) Deformable-LFVS [17]; (e) LF-EASR [19]; (f) Distg-ASR [18]; (g) LFVS-Mamba.



(a) (b) (c) (d)

Fig. 5. Angular Consistency on Dishes (HCI-new [22]) (a) GT; (b) Deformable-LFVS [17]; (c) Distg-ASR [18]; (d) LFVS-Mamba.

To validate LFVS-Mamba components, we tested four ablation variants of LFVS-Mamba (see Table III). Variant I uses bidirectional scanning along a single axis, which restricts angular context and omits diagonal view correlations. Variant II uses channel-wise four-direction scanning, but processes each channel independently, preventing cross-channel fusion and limiting spatial-angular cue learning. Variant III excludes SAFE-SSM and relies solely on DFE-SSM, resulting in poor capture of spatial-angular correlations. Variant IV excludes DFE-SSM and relies solely on SAFE-SSM, yielding insufficient depth-informed refinement and geometric artifacts.

TABLE III. ABLATION STUDY ON SYNTHETIC DATASETS

Variant	**Method**	**Synthetic Datasets**
I	Bidirectional scanning	38.12/0.968
II	Channel-wise four directional scanning	38.28/0.970
III	SAFE-SSM	37.95/0.967
IV	DFE-SSM	38.39/0.971

IV. CONCLUSION

LFVS-Mamba efficiently exploits LF structural characteristics by independently extracting 2D slices of spatial context, complementary angular information and depth cues. This network design preserves local detail and global context with a slightly higher number of parameters than the second best method (3.19 M vs 2.74 M for Distg-ASR [18]) and less than half of LF-EASR [19], while synthesizing dense, angularly consistent LFs on synthetic and real-world datasets. LFVS-Mamba still struggles with complex occlusions. Future work will consider adaptive refinement and dynamic angular modeling to improve fidelity in such type of scenes.

REFERENCES

[1] J. Peng, Z. Xiong, Y. Zhang, D. Liu, and F. Wu, "LF-fusion: Dense and accurate 3D reconstruction from light field images," in *2017 IEEE Visual Communications and Image Processing, VCIP 2017*, Feb. 2018, vol. 2018-January, pp. 1–4. doi: 10.1109/VCIP.2017.8305046.

[2] Y. Ding, Z. Chen, Y. Ji, J. Yu, and J. Ye, "Light Field-Based Underwater 3D Reconstruction via Angular Re-Sampling," *IEEE Trans. Comput. Imaging*, vol. 9, pp. 881–893, 2023, doi: 10.1109/TCI.2023.3319983.

[3] N. Liu, W. Zhao, D. Zhang, J. Han, and L. Shao, "Light Field Saliency Detection with Dual Local Graph Learning and Reciprocative Guidance," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 4692–4701, 2021, doi: 10.1109/ICCV48922.2021.00467.

[4] W. Yan, X. Zhang, H. Chen, C. Ling, and D. Wang, "Light Field Depth Estimation Based on Channel Attention and Edge Guidance," *Proc. - 2022 Chinese Autom. Congr. CAC 2022*, vol. 2022-Janua, pp. 2595–2600, 2022, doi: 10.1109/CAC57257.2022.10054964.

[5] Y. Wang, J. Yang, Y. Guo, C. Xiao, and W. An, "Selective Light Field Refocusing for Camera Arrays Using Bokeh Rendering and Superresolution," *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 204–208, 2019, doi: 10.1109/LSP.2018.2885213.

[6] B. Wilburn *et al.*, "High performance imaging using large camera arrays," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 765–776, 2005, doi: 10.1145/1073204.1073259.

[7] V. Vaish and A. Adams, "The (NEW) Stanford Light Field Archive,." available: http://lightfield.stanford.edu/.

[8] "Light Field Forum." http://lightfield-forum.com/en/?s=Lytro

[9] "Raytrix." https://raytrix.de/

[10] S. Yun, J. Jang, and J. Paik, "Geometry-Aware Light Field Angular Super Resolution Using Multiple Receptive Field Network," *2022 Int. Conf. Electron. Information, Commun. ICEIC 2022*, pp. 2–4, 2022, doi: 10.1109/ICEIC54506.2022.9748458.

[11] J. Jin, J. Hou, H. Yuan, and S. Kwong, "Learning light field angular super-resolution via a geometry-aware network," *AAAI 2020 - 34th AAAI Conf. Artif. Intell.*, pp. 11141–11148, 2020, doi: 10.1609/aaai.v34i07.6771.

[12] Z. Liang, Y. Wang, L. Wang, J. Yang, S. Zhou, and Y. Guo, "Learning Non-Local Spatial-Angular Correlation for Light Field Image Super-Resolution," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 12342–12352, 2023, doi: 10.1109/ICCV51070.2023.01137.

[13] S. Wang *et al.*, "Spatial–angular–epipolar transformer for light field spatial and angular super-resolution," *Displays*, vol. 85, no. August, p. 102816, 2024, doi: 10.1016/j.displa.2024.102816.

[14] R. Cong, H. Sheng, D. Yang, Z. Cui, and R. Chen, "Exploiting Spatial and Angular Correlations with Deep Efficient Transformers for Light Field Image Super-Resolution," *IEEE Trans. Multimed.*, vol. 26, pp. 1421–1435, 2024, doi: 10.1109/TMM.2023.3282465.

[15] Y. Liu *et al.*, "VMamba: Visual State Space Model," vol. 1, pp. 1–33, 2024, [Online]. Available: http://arxiv.org/abs/2401.10166

[16] D. Liu, Z. Tong, Y. Huang, Y. Chen, Y. Zuo, and Y. Fang, "Geometry-assisted multi-representation view reconstruction network for Light Field image angular super-resolution," *Knowledge-Based Syst.*, vol. 267, p. 110390, 2023, doi: 10.1016/j.knosys.2023.110390.

[17] M. Zubair, P. Nunes, C. Conti, and L. D. Soares, "Light Field View Synthesis Using Deformable Convolutional Neural Networks," *2024 Pict. Coding Symp. PCS 2024 - Proc.*, pp. 11–15, 2024, doi: 10.1109/PCS60826.2024.10566360.

[18] Y. Wang *et al.*, "Disentangling Light Fields for Super-Resolution and Disparity Estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 425–443, 2022, doi: 10.1109/TPAMI.2022.3152488.

[19] G. Liu, H. Yue, J. Wu, and J. Yang, "Efficient Light Field Angular Super-Resolution With Sub-Aperture Feature Learning and Macro-Pixel Upsampling," *IEEE Trans. Multimed.*, pp. 1–13, 2022, doi: 10.1109/TMM.2022.3211402.

[20] R. Gao, Z. Xiao, and Z. Xiong, "Mamba-Based Light Field Super-Resolution with Efficient Subspace Scanning," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)* , vol. 15476 LNCS, pp. 421–437, 2025, doi: 10.1007/978-981-96-0917-8_24.

[21] A. Gu, K. Goel, and C. Ré, "Efficiently Modeling Long Sequences With Structured State Spaces," *ICLR 2022 - 10th Int. Conf. Learn. Represent.*, pp. 1–32, 2022.

[22] A. Gu and T. Dao, "Mamba: Linear-Time Sequence Modeling with Selective State Spaces," arXiv, 2024, arXiv:2312.00752

[23] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[24] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10113 LNCS, no. 3, pp. 19–34, 2017, doi: 10.1007/978-3-319-54187-7_2.

[25] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and benchmarks for densely sampled 4D light fields," *18th Int. Work. Vision, Model. Vis. VMV 2013*, pp. 225–226, 2013, doi: 10.2312/PE.VMV.VMV13.225-226.

[26] N. K. Kalantari, T. C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM Trans. Graph.*, vol. 35, no. 6, 2016, doi: 10.1145/2980179.2980251.

[27] A. S. Raj, M. Lowney, R. Shah, and G. Wetzstein, "Stanford Lytro Light Field Archive," 2016. http://lightfields.stanford.edu/LF2016.html