ISCTE IUL
REPOSITÓRIO
INSTITUTO UNIVERSITÁRIO DE LISBOA

Repositório ISCTE-IUL

Deposited in *Repositório ISCTE-IUL*:
2026-10-08

Deposited version:
Accepted Version

Peer-review status of attached file:
Peer-reviewed

Citation for published item:
Zubair, M., Nunes, P., Conti, C. & Soares, L. D. (2025). Swinscale-LFVS: Parallel feature integration for light field view synthesis. 2025 IEEE International Conference on Image Processing (ICIP), Proceedings (pp. 1942?1947). IEEE. https://doi.org/10.1109/ICIP55913.2025.11084705

Further information on publisher's website:
10.1109/ICIP55913.2025.11084705

Publisher's copyright statement:
This is the peer reviewed version of the following article: Zubair, M., Nunes, P., Conti, C. & Soares, L. D. (2025). Swinscale-LFVS: Parallel feature integration for light field view synthesis. 2025 IEEE International Conference on Image Processing (ICIP), Proceedings (pp. 1942?1947). IEEE. https://doi.org/10.1109/ICIP55913.2025.11084705, which has been published in final form at https://dx.doi.org/10.1109/ICIP55913.2025.11084705. This article may be used for non-commercial purposes in accordance with the Publisher's Terms and Conditions for self-archiving.

Use policy

Creative Commons CC BY 4.0
The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in the Repository
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Serviços de Informação e Documentação, Instituto Universitário de Lisboa (ISCTE-IUL)
Av. das Forças Armadas, Edifício II, 1649-026 Lisboa Portugal
Phone: +(351) 217 903 024 | e-mail: administrador.repositorio@iscte-iul.pt
https://repositorio.iscte-iul.pt

SWINSCALE-LFVS: PARALLEL FEATURE INTEGRATION 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 (LF) view synthesis aims to synthesize a dense set of LF views from a sparse set of input views. Although many recent learning-based methods have shown promising results in this task, they often rely on deep residual networks or on multiple LF representations to extract dense features, without fully exploiting the geometric structure of the LFs. In this paper, we introduce SwinScale-LFVS, a novel framework that combines the strengths of the Swin Transformer and the Multi-Scale Convolutional Network in parallel streams. The first stream uses a Swin Transformer to model local and global features using a geometry-aware Angular Mutual Self Attention (AMSA) network, and the second stream uses multi-scale 3D convolutions to extract dense features and to ensure spatial-angular consistency in synthesized LF views. The outputs from these streams are integrated and processed by an LF View Synthesis (LFVS) network to synthesize high-quality dense LF views. Extensive experiments show that SwinScale-LFVS outperforms existing methods on both real-world and synthetic datasets. The code is publicly available at https://github.com/MSP-IUL/SwinScale-LFVS.

Index Terms— *Light field view synthesis, swin transformer, angular attention, multi-scale convolutions, global features*

1. INTRODUCTION

Light field (LF) images capture both the direction and intensity of light rays in a scene, providing rich 3D geometry information. As a result, LFs are more suitable than 2D images for applications such as 3D reconstruction [1], view synthesis [2], depth estimation [3], segmentation [4], and refocusing [5]. However, acquiring LF images with a bulky camera array [6] can be expensive and cumbersome. On the other hand, commercial single-tier LF cameras [7], [8] have limited sensor resolution, which necessarily leads to a trade-off between spatial and angular resolution.

Instead of relying on physical devices to acquire LF images with higher angular resolution, learning-based methods have been developed to synthesize dense LF views from a sparse set of input views, commonly referred to as LF view synthesis, angular reconstruction, or angular super-resolution. Generally, these methods are classified as being either i) depth-dependent, or ii) non-depth-dependent. Depth-dependent methods [9], [10], [11] warp the source view images to target view locations using a depth estimation network, while non-depth-dependent methods [12], [13] synthesize dense LF views by leveraging local LF information without relying on depth maps.

Existing depth-dependent [9] and multiple representations [10], [11] LF view synthesis methods employ depth estimation networks with large receptive fields. However, these methods often produce artifacts in the synthesized views due to depth estimation errors, stemming from the localized nature of convolutional operations in CNNs, which limits their ability to capture global scene context despite large receptive fields. Moreover, methods using multiple LF representations heavily rely on specific input formats, reducing their flexibility and generalization for synthesizing a dense set of LF views. On the other hand, existing non-depth-dependent methods using deep residual networks [14], [15] fail to exploit the geometric structure of LF views, limiting the quality of synthesized dense LF views.

To address the above-mentioned issues, we propose the non-depth-dependent SwinScale-LFVS method that combines the strengths of the Swin Transformer [16] and the Multi-Scale Convolutional Network in parallel streams, followed by an LF view synthesis (LFVS) network to synthesize the dense LF views. The Swin Transformer Stream (STS) extracts both local and global features using a geometry-aware Angular Mutual Self Attention (AMSA) network, while the Multi-Scale Convolution Stream (MSCS) uses multi-scale 3D convolutions to extract dense features from the LF Sub-Aperture Images (SAIs).

The STS consists of three modules: a Shallow Feature Extraction (SFE) module, an Encoder-Decoder (ED) module, and a Feature Aggregation (FA) module. The SFE module is designed to extract low-level features from the input sparse LF. The ED module integrates an AMSA network to extract both local and global features. The AMSA network is composed of two sub-networks: Angular Mutual Attention (AMA) and Multi-Head Self Attention (MHSA). The AMA sub-network aligns views by modeling mutual angular dependencies, ensuring precise feature correlation across views. In parallel, the MHSA sub-network extracts hierarchical features by leveraging both local and global information through a window-based mechanism. Finally, the extracted hierarchical LF features are passed through the FA module to aggregate and generate a unified LF feature representation.

The MSCS focuses on extracting dense spatial-angular features using multi-scale 3D convolutions. Unlike existing transformer-based methods in image super-resolution [17] and restoration [18], which typically rely on shallow feature extraction networks, our method integrates a more sophisticated feature extraction module. Specifically, MSCS consists of two modules: a Dense Feature

This work is funded by FCT/MECI through national funds and when applicable co-funded by EU funds under UID/50008: Instituto de Telecomunicações.

Extraction (DFE) and a Feature Refinement (FR). The DFE module extracts dense features using 3D convolutional layers with varying receptive field sizes, while the FR module removes the redundant features and presents a unified representation.

This approach enhances the learning of feature representations by capturing detailed spatial-angular information, complementing the STS's ability to model global dependencies. By combining these strengths, the SwinScale-LFVS method addresses the limitations of shallow feature extraction in transformer-based networks, offering a robust solution for LF view synthesis. Finally, the STS and MSCS feature representations are integrated through concatenation operation and then processed by the LFVS network to synthesize the dense LF views.

The main contributions of this paper are summarized as follows:

- We propose a dual-stream architecture, combining STS and MSCS, to integrate local and global feature modeling with fine-grained spatial-angular detail extraction, thus enhancing LF view synthesis quality.
- We propose an AMSA network to ensure angular consistency in synthesized LF views, effectively enhancing global feature modeling and preserving coherence across angular dimensions.
- We propose an MSCS network to capture spatial-angular features across varying receptive field sizes, offering a detailed representation of LF structures and addressing the limitations of shallow feature extraction in transformer-based methods.

The paper is organized as follows: Section II reviews related work; Section III details the proposed method; Section IV discusses the experimental results; and, finally, Section V concludes the paper.

2. RELATED WORK

Available learning-based LF view synthesis methods can be categorized as: i) depth-dependent, or ii) non-depth-dependent.

2.1. Depth-Dependent Methods

Depth-dependent methods synthesize the densely sampled LF views by warping the input views to target angular positions using estimated disparity maps. Kalantari *et al.* [2] proposed a method using two sequential CNNs to estimate disparity and color values. This method ignores the correlations between the warped views, leading to limited reconstruction quality of synthesized views. Jin et al. [9] proposed a two-module framework: one for depth estimation and one for refinement. The depth estimation network was designed with a large receptive field to synthesize LF views based on warping. The refinement module enhances the quality of the novel views by integrating features effectively. Later, Jin et al. [19] used Plane-Sweep Volumes (PSVs) with pre-defined disparity ranges, followed by refinement of the warped views using spatial-angular alternating convolutions [20]. This method effectively integrates spatial and angular information to enhance LF view synthesis but is still susceptible to errors from pre-defined disparities. Yun et al. [21] constructed multiplane images by projecting the scene onto multiple planes at various disparity levels. Each plane captures the color and transparency values corresponding to specific depth layers in the scene. This representation enables the method to incorporate depth information directly into the view synthesis process. Liu et al. [10] proposed a geometry-assisted LF multi-representations method employing branched convolutional networks for different feature modalities, enabling adaptation to varying scene complexities. This geometry-aware mechanism effectively handles large disparity variations and occlusions. Zubair *et al.* [11] used Deformable Convolutional Neural Networks (DCNNs) for LF view synthesis. This architecture comprises two modules: a multi-representation view synthesis module designed to extract dense features from various LF representations, and a geometry-aware refinement module that synthesizes a dense LF with angular consistency by leveraging the structural characteristics of the sparse input.

2.2. Non-Depth-Dependent Methods

Non-depth-dependent methods refer to the process of synthesizing novel views of a scene without relying on depth maps. Liu *et al.* [22] proposed a Densely Connected Multi-loss CNN (DCM-CNN) that uses a multi-loss optimization strategy by combining reconstruction, angular consistency, and perceptual losses to synthesize high-quality intermediate views from sparse inputs. This method preserves the angular consistency of the synthesized views, but multi-loss design increases computational complexity. Wu *et al.* [23] proposed a spatial-angular attention mechanism to capture non-local correlations between views. The architecture integrated spatial and angular information using a multi-scale framework to enhance reconstruction quality. This method struggles with complex occlusions and varying lighting conditions. Liu et al. [12] proposed an efficient LF view synthesis framework using SAIs for feature learning. The method uses multi-scale spatial-angular correlations and micro-pixel features to reduce artifacts in regions with large disparities. Wang *et al.* [13]. proposed a disentangling mechanism to separate spatial and angular features in LFs. This method achieves state-of-the-art performance in LF view synthesis with improved fidelity but struggles with large baseline view synthesis. Wang *et al.* [24] proposed an LF view synthesis method using residual channel attention mechanisms and a classification-guided up-sampling module to enhance feature extraction and reconstruction precision. The approach adaptively refines angular information and addresses occlusions but struggles with extremely sparse inputs.

3. PROPOSED METHOD

An LF is a 4D function $L(u, v, x, y)$ where (u, v) and (x, y) denote the angular and spatial coordinates, respectively. The objective of the proposed method is to design a framework $f(\cdot)$ capable of synthesizing a high angular-resolution LF, $L^{har}(u, v, x, y)$, with dimensions $U \times V \times X \times Y$, from a low-angular-resolution LF, $L^{lar}(u', v', x, y)$, with dimensions $U' \times V' \times X \times Y$. The entire process can be expressed as:

$$L^{har}(u, v, x, y) = f(L^{lar}(u', v', x, y), \theta) \quad (1)$$

where $f(\cdot)$ denotes the mapping function from low angular to high angular resolution and θ represents the network parameters. The architecture of the SwinScale-LFVS method is illustrated in Fig. 1.

3.1. Swin Transformer Stream (STS)

The STS is designed to extract both local and global features by using its hierarchical structure. The modules of the STS are described below.

3.1.1. Shallow Feature Extraction (SFE)

The SFE module ensures efficient initial feature representation at the channel level, while preserving the spatial structure of the input data. This module is designed to extract shallow features using a 3D convolutional layer with a kernel size of 1×1×1 as defined by (2):

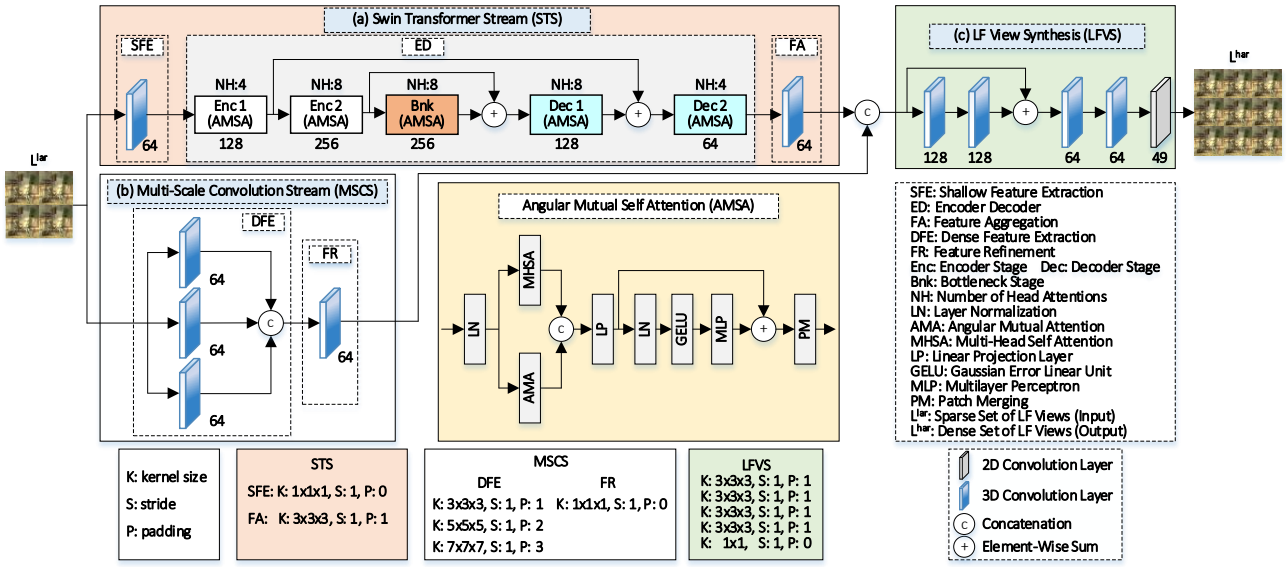


Fig. 1. SwinScale-LFVS modules: (a) Swin-Transformer Stream, (b) Multi-Scale Convolution Stream, and (c) LF View Synthesis.

$$F_s = W_{1\times1\times1} * L^{lar} + b_1 \tag{2}$$

where F_s is the extracted shallow features, W represents the weights of the convolution where its subscript represents the kernel size, L^{lar} represents the input SAIs and b is the bias term and retains the same meaning in subsequent equations (11)-(13) and (15)-(18).

3.1.2. Encoder-Decoder (ED)

The ED module is a Swin-Transformer block that consists of two encoder stages, a bottleneck stage, and two decoder stages. Each stage has an AMSA network which consists of two sub-networks: MHSA and AMA. The MHSA sub-network extracts hierarchical features by capturing local and global information, while the AMA sub-network ensures angular consistency by modeling geometry-aware mutual angular dependencies.

Encoder stage 1 is designed to extract initial spatial-angular features from shallow features, F_s, by applying Layer Normalization (LN), followed by simultaneous processing through MHSA $\left(F_{E_1}^{MHSA}\right)$ and AMA $\left(F_{E_1}^{AMA}\right)$ networks. The resulting feature representations are concatenated (C) and passed through a Linear Projection (LP) layer, to align the feature dimension.

$$E_1 = LP\left(C\left(F_{E_1}^{MHSA}\left(LN(F_s)\right), F_{E_1}^{AMA}\left(LN(F_s)\right)\right)\right) \tag{3}$$

where E_1 represents the spatial-angular features of encoder stage 1. To refine E_1, a Gaussian Error Linear Unit (GELU) activation function and an MLP (F_{E1}^{MLP}) are applied to the normalized features, enhancing non-linearity and feature representation. Next, Patch Merging (PM) is employed for spatial downsampling, where features are rearranged, followed by LN and a linear transformation to generate a structurally compact feature map.

$$E_{f1} = PM\left(F_{E_1}^{MLP}\left(GELU\left(LN(E_1)\right)\right) + E_1\right) \tag{4}$$

The output of the encoder stage 1, E_{f1}, representing initial spatial-angular features, is passed to encoder stage 2, where the operations in equations (3) and (4) are reapplied to extract the hierarchical spatial-angular features. The resulting hierarchical features, E_{f2}, are then passed to the bottleneck stage and forwarded to decoder stage 1 as a skip connection to enhance the information flow. The bottleneck stage preserves channel and spatial dimensions, enhancing feature diversity by integrating both local and global information through $\left(F_B^{MHSA}\right)$ and (F_B^{AMA}), followed by an LP layer to align the feature dimension. To ensure dimension consistency and effective feature transformation, the bottleneck features, B_f, undergo LN, followed by MLP (F_B^{MLP}) with GELU activation. The processed features are then restructured using PM to maintain hierarchical coherence for subsequent processing:

$$B_f = LP\left(C\left(F_B^{MHSA}\left(LN(E_{f2})\right), F_B^{AMA}\left(LN(E_{f2})\right)\right)\right) \tag{5}$$

$$B_{fe} = PM\left(F_B^{MLP}\left(GELU\left(LN(B_f)\right)\right) + B_f\right) \tag{6}$$

where B_{fe} represents the bottleneck stage features.

Decoder stage 1 refines hierarchical features and reconstructs dense LF representations by integrating bottleneck outputs and encoder stage 2 features via a skip connection. The bottleneck features, B_{fe}, are added to the skip connection output from encoder stage 2, E_{f2}, to generate dense local and global features:

$$D_{fb} = \left(B_{fe} + E_{f2}\right) \tag{7}$$

The combined features D_{fb} are processed through LN, followed by $\left(F_B^{MHSA}\right)$ and (F_B^{AMA}) to extract the hierarchical spatial-angular features. The output is then passed through an LP layer to align the feature dimension. The resulting features undergo LN, GELU and MLP $\left(F_{D_1}^{MLP}\right)$, followed by PM for spatial upsampling, producing dense LF features, D_{f1}:

$$D_1 = LP\left(C\left(F_{D_1}^{MHSA}\left(LN(D_{fb})\right), F_{D_1}^{AMA}\left(LN(D_{fb})\right)\right)\right) \tag{8}$$

$$D_{f1} = PM\left(F_{D_1}^{MLP}\left(GELU\left(LN(D_1)\right)\right) + D_1\right) \tag{9}$$

The upsampled features, D_{f1}, are added to the skip connection features from encoder stage 1, E_{f1}, to ensure angular consistency:

$$D_{fed} = \left(D_{f1} + E_{f1}\right) \tag{10}$$

where D_{fed} represents the dense decoder features at stage 2, which are further processed using the operations described in equations (8) and (9) to extract rich hierarchical features, D_{f2}, encapsulating both local and global information.

3.1.3. Feature Aggregation (FA)

The rich hierarchical LF features, D_{f2}, are passed through a 3D convolution layer with a kernel size of 3×3×3 to generate unified aggregated LF features:

$$F_t = W_{3\times3\times3} * D_{f2} + b_2 \tag{11}$$

where F_t represents the aggregated features.

3.2. Multi-Scale Convolution Stream (MSCS)

The MSCS is designed to capture multi-scale features, allowing the network to represent fine details and broader spatial-angular patterns. It comprises two modules: DFE and FR.

3.2.1. Dense Feature Extraction (DFE)

The DFE module processes input SAIs, L^{lar}, using three parallel 3D convolutional layers with kernel sizes 3×3×3, 5×5×5, and 7×7×7 to capture multi-scale features. This process is expressed as:

$$F_m = C(W_{3\times3\times3} * L^{lar} + b_3, W_{5\times5\times5} * L^{lar} + b_4, W_{7\times7\times7} * L^{lar} + b_5) \tag{12}$$

where F_m represents the extracted multi-scale features.

3.2.2 Feature Refinement (FR)

The FR module reduces the channel dimension and provides a unified LF feature representation by using a 3D convolutional layer with a kernel size of 1×1×1:

$$F_{ms} = W_{1\times1\times1} * F_m + b_6 \tag{13}$$

where F_{ms} represents the unified LF features.

3.3. LF View Synthesis (LFVS)

Finally, LF features from STS and MSCS are concatenated and then, in the LFVS module, processed through a 3D residual convolutional network with a 3×3×3 kernel, followed by two 3×3×3 3D convolution layers and a 1×1 2D convolutional layer, to synthesize the dense LF views. This process is expressed as:

$$F_v = C(F_{ms}, F_t) \tag{14}$$

$$F_r = F_v + W^2_{3\times3\times3} * (\emptyset\ (W^1_{3\times3\times3} * F_v + b_7) + b_8) \tag{15}$$

$$F_d = (W^3_{3\times3\times3} * F_r + b_9) \tag{16}$$

$$LF_{VS} = (W^4_{3\times3\times3} * F_d + b_{10}) \tag{17}$$

$$L^{har} = (W_{1\times1} * LF_{VS} + b_{11}) \tag{18}$$

where F_v represents the integrated features, F_r represents the residual features, $\emptyset$ represents the ReLU activation function, and F_d represents the rich LF features. The LF_{VS} represents the dense LF views that are reshaped from 3D to 2D to produce the output L^{har}.

4. EXPERIMENTAL RESULTS AND DISCUSSION

SwinScale-LFVS is trained with L1 loss using the Adam optimizer [25], a batch size of 1, and an initial learning rate of 1×10^{-4}, halved every 15 epochs. The model reconstructs 7×7 LF data from 2×2 inputs by splitting SAIs into 64×64 non-overlapping patches and is trained on 70 epochs. A synthetic LF network is trained on 20 scenes from the HCI-new dataset [26] and evaluated on 4 scenes from HCI-new [26] and 5 scenes from the HCI-old dataset [27]. A real-world LF network is trained on 100 LF scenes from Kalantari *et al.* [2] and the Stanford Lytro Archive [28]. Performance evaluation is conducted on three real-world LF datasets: 30 LF scenes from the UCSD dataset [2], 25 scenes from the occlusion dataset [28], and 15 scenes from the reflective dataset [28].

4.1 Results Comparison with State-of-the-art Methods

SwinScale-LFVS is compared with five state-of-the-art methods: three depth-dependent [9], [10], [11], and two non-depth-dependent methods [12], [13]. All methods are retrained using publicly available models for fair comparison and performance is evaluated using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM).

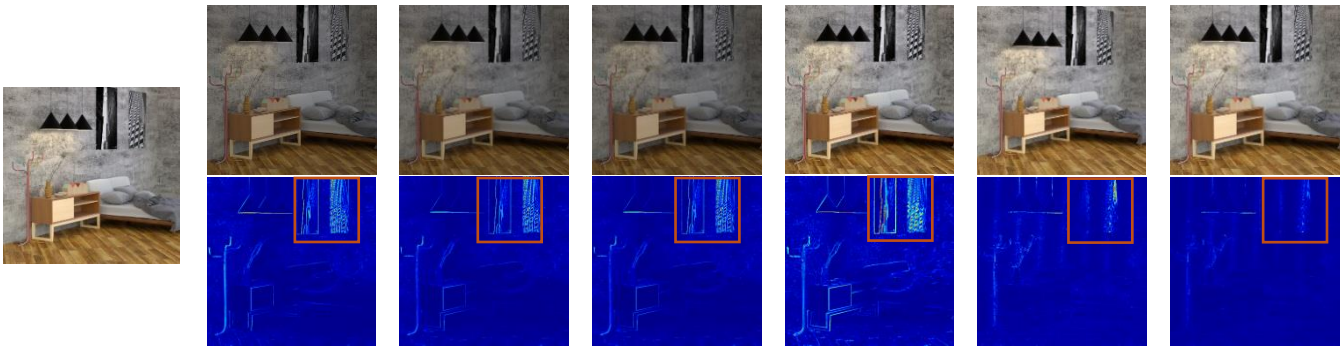
Table I and Table II summarize the results on synthetic and real-world LFs. On synthetic LFs, SwinScale-LFVS achieves average PSNR gains of 4.85 dB, 1.31 dB, 0.97 dB, 0.99 dB, and 0.72 dB over LFASR-GEO [9], GA-MRVR [10], Deformable-LFVS [11], LF-EASR [12], and Distg-ASR [13], respectively. In real-world LFs, SwinScale-LFVS shows PSNR gains of 3.24 dB, 0.73 dB, 0.45 dB, 0.65 dB, and 0.39 dB. The SSIM results align with PSNR, except for Deformable-LFVS [11]. The highest values are in bold, and the second-best results are underlined in Table I and Table II.

Table I. Quantitative comparison of SwinScale-LFVS with state-of-the-art methods on synthetic datasets (PSNR/SSIM)

Dataset	LFASR-GEO [9]	GA-MRVR [10]	Deformable-LFVS [11]	LF-EASR [12]	Distg-ASR [13]	SwinScale-LFVS
HCI-new	32.29/0.911	34.28/0.936	34.52/0.936	34.27/0.949	34.16/0.969	**35.75/0.972**
HCI-old	35.73/0.937	40.82/0.953	41.25/0.958	41.47/0.970	**42.13**/0.974	41.97/**0.975**
Average	34.01/0.924	37.55/0.945	37.89/0.955	37.87/0.960	38.14/0.972	**38.86/0.974**

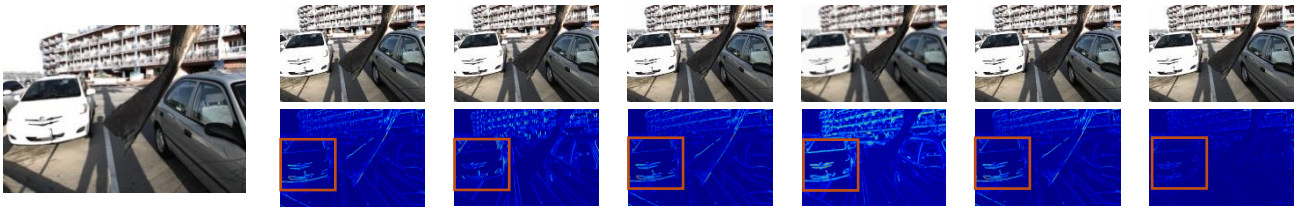
Table II. Quantitative comparison of the SwinScale-LFVS with state-of-the-art methods on real-world datasets (PSNR/SSIM)

Dataset	LFASR-GEO [9]	GA-MRVR [10]	Deformable-LFVS [11]	LF-EASR [12]	Distg-ASR [13]	SwinScale-LFVS
UCSD	40.78/0.982	42.91/0.987	43.19/0.987	43.15/0.980	43.60/0.986	**43.87/0.987**
Occlusions	36.43/0.971	39.04/0.981	39.45/**0.984**	39.18/0.979	39.40/0.982	**39.85**/0.983
Reflective	36.25/0.945	39.04/0.962	39.19/**0.963**	38.91/0.958	39.02/0.960	**39.45**/0.962
Average	37.82/0.966	40.33/0.976	40.61/**0.978**	40.41/0.972	40.67/0.976	**41.06**/0.977



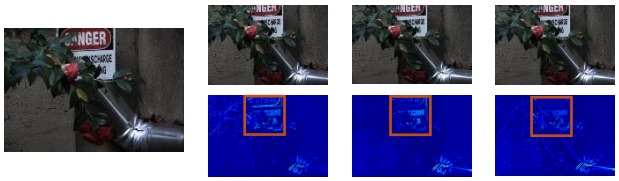
(a) (b) (c) (d) (e) (f) (g)

Fig. 2. Qualitative results on a synthetic LF (HCI-new, bedroom): (a) GT; (b) LFASR-GEO [9]; (c) GA-MRVR [10]; (d) Deformable-LFVS [11]; (e) LF-EASR [12]; (f) Distg-ASR [13]; (g) SwinScale-LFVS (proposed).



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

Fig. 3. Qualitative results on a real-world LF (UCSD, car): (a) GT; (b) LFASR-GEO [9]; (c) GA-MRVR [10]; (d) Deformable-LFVS [11]; (e) LF-EASR [12]; (f) Distg-ASR [13]; (g) SwinScale-LFVS (proposed).



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

Fig. 4. Qualitative results on a real-world scene (Reflective, Reflective_12_eslf): (a) GT; (b) Deformable-LFVS [11]; (c) Distg-ASR [13]; (d) SwinScale-LFVS (proposed).

Fig. 2 and Fig. 3 error-maps show that the SwinScale-LFVS method produces visually appealing results on HCI-new scene (bedroom) and UCSD scene (car). Depth-dependent methods [9], [10], [11] produces artifacts and blurring effects in the bedroom and car scenes, while non-depth-dependent methods [12], [13] struggle to handle the occluded regions, leading to ghosting artifacts in the synthesized views. Fig. 4 error maps reveal that the SwinScale-LFVS method synthesizes high-fidelity views with fewer erroneous pixels in the Reflective_12_eslf scene, compared to the two highest performing methods. These visual comparisons show that the SwinScale-LFVS excels in recovering fine details across synthetic and real-world scenes.

4.2. Ablation Study

To validate the effectiveness of each component in the proposed SwinScale-LFVS method, we conducted an ablation study with three experimental variants that selectively exclude specific components The results, in Table III, show that removing STS leads to the most significant drop in performance on both synthetic and real-world LFs. Conversely, the MSCS stream, while less critical, still contributes to the overall performance of the proposed method, as indicated by a noticeable decrease when it is omitted. These findings highlight the key role of STS and the complementary contributions of AMA and MSCS in achieving robust LF view synthesis.

Table III. Ablation study

Variants	Synthetic datasets	Real-world datasets
Without STS	37.80/0.921	39.63/0.919
Without AMA	38.10/0.949	40.09/0.945
Without MSCS	38.41/0.959	40.34/0.956

5. CONCLUSION

In this paper, we introduced the SwinScale-LFVS method, which synergizes the strengths of STS and MSCS. The STS captures local and global features through its hierarchical attention mechanism, ensuring geometry-aware angular consistency via the AMA sub-network. Concurrently, the MSCS extracts dense spatial-angular features using multi-scale 3D convolutions, further enhancing spatial-angular consistency. The integration of these streams, followed by the LFVS network, enables accurate and angularly consistent LF view synthesis. Although the SwinScale-LFVS method achieves high-quality LF view synthesis, it increases the computational costs due to its higher model complexity with 11.81 million parameters (M), notably when compared to the next best alternatives (i.e., 2.74 M for Distg-ASR [13], and 2.37 M for Deformable-LFVS [11]). Future work will focus on developing lightweight architectures and exploring depth-dependent methods to enhance efficiency and generalization across challenging scenes and complex datasets.

6. 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-Janua, pp. 1–4. doi: 10.1109/VCIP.2017.8305046.

[2] N. K. Kalantari, T. C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," ACM Trans. Graph., vol. 35, no. 6, pp. 1–10, 2016, doi: 10.1145/2980179.2980251.

[3] Y. Zhang et al., "Light-Field Depth Estimation via Epipolar Plane Image Analysis and Locally Linear Embedding," IEEE Trans. Circuits Syst. Video Technol., vol. 27, no. 4, pp. 739–747, 2017, doi: 10.1109/TCSVT.2016.2555778.

[4] M. F. Hamad, C. Conti, P. Nunes, and L. D. Soares, "Hyperpixels: Flexible 4D Over-Segmentation for Dense and Sparse Light Fields," IEEE Trans. Image Process., vol. 32, pp. 3790–3805, 2023, doi: 10.1109/TIP.2023.3290523.

[5] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras," Proc. IEEE Int. Conf. Comput. Vis., pp. 673–680, 2013, doi: 10.1109/ICCV.2013.89.

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

[8] Raytrix, "Raytrix," 2024, [Online]. Available: https://www.raytrix.de/.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] N. Meng, H. K. H. So, X. Sun, and E. Y. Lam, "High-Dimensional Dense Residual Convolutional Neural Network for Light Field Reconstruction," IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 3, pp. 873–886, 2021, doi: 10.1109/TPAMI.2019.2945027.

[15] J. Shi, X. Jiang, and C. Guillemot, "Deep Residual Architecture Using Pixel and Feature Cues for View Synthesis and Temporal Interpolation," IEEE Trans. Comput. Imaging, vol. 8, pp. 246–259, 2022, doi: 10.1109/TCI.2022.3160671.

[16] Z. Liu et al., "Swin Transformer," 2021 IEEE/CVF Int. Conf. Comput. Vis., pp. 9992–10002, 2021, [Online]. Available: https://ieeexplore.ieee.org/document/9710580/

[17] Z. Lu, J. Li, H. Liu, C. Huang, L. Zhang, and T. Zeng, "Transformer for Single Image Super-Resolution," IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., vol. 2022-June, pp. 456–465, 2022, doi: 10.1109/CVPRW56347.2022.00061.

[18] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "SwinIR: Image Restoration Using Swin Transformer," Proc. IEEE Int. Conf. Comput. Vis., vol. 2021-Octob, pp. 1833–1844, 2021, doi: 10.1109/ICCVW54120.2021.00210.

[19] J. Jin, J. Hou, J. Chen, H. Zeng, S. Kwong, and J. Yu, "Deep Coarse-to-Fine Dense Light Field Reconstruction with Flexible Sampling and Geometry-Aware Fusion," IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 4, pp. 1819–1836, 2022, doi: 10.1109/TPAMI.2020.3026039.

[20] H. W. F. Yeung, J. Hou, X. Chen, J. Chen, Z. Chen, and Y. Y. Chung, "Light Field Spatial Super-Resolution Using Deep Efficient Spatial-Angular Separable Convolution," IEEE Trans. Image Process., vol. 28, no. 5, pp. 2319–2330, 2019, doi: 10.1109/TIP.2018.2885236.

[21] S. Yun, J. Jang, and J. Paik, "Learning-based Light Field View Synthesis Using Multiplane Images," 2023 Int. Conf. Electron. Information, Commun. ICEIC 2023, pp. 1–3, 2023, doi: 10.1109/ICEIC57457.2023.10049922.

[22] L. Liu, Z. Nian, and C. Jung, "DCM-CNN: Densely connected multiloss convolutional neural networks for light field view synthesis," IEEE Access, vol. 8, pp. 78542–78552, 2020, doi: 10.1109/ACCESS.2020.2989453.

[23] G. Wu, Y. Wang, Y. Liu, L. Fang, and T. Chai, "Spatial-Angular Attention Network for Light Field Reconstruction," IEEE Trans. Image Process., vol. 30, pp. 8999–9013, 2021, doi: 10.1109/TIP.2021.3122089.

[24] X. Wang, Z. Wang, and S. You, "Light field angular super resolution based on residual channel attention and classification up-sampling," Multimed. Tools Appl., 2024, doi: 10.1007/s11042-024-19359-6.

[25] 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.

[26] 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.

[27] 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.

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