Computational Visual Media
https://doi.org/10.1007/s41095-022-0297-1 Vol. 9, No. 4, December 2023, 843–858

Research Article

Light field super-resolution using complementary-view feature attention

Wei Zhang[1], Wei Ke[1] (✉), Da Yang[2] (✉), Hao Sheng[1,2], and Zhang Xiong[1,2]

Abstract Light field (LF) cameras record multiple perspectives by a sparse sampling of real scenes, and these perspectives provide complementary information. This information is beneficial to LF super-resolution (LFSR). Compared with traditional single-image super-resolution, LF can exploit parallax structure and perspective correlation among different LF views. Furthermore, the performance of existing methods are limited as they fail to deeply explore the complementary information across LF views. In this paper, we propose a novel network, called the light field complementary-view feature attention network (LF-CFANet), to improve LFSR by dynamically learning the complementary information in LF views. Specifically, we design a residual complementary-view spatial and channel attention module (RCSCAM) to effectively interact with complementary information between complementary views. Moreover, RCSCAM captures the relationships between different channels, and it is able to generate informative features for reconstructing LF images while ignoring redundant information. Then, a maximum-difference information supplementary branch (MDISB) is used to supplement information from the maximum-difference angular positions based on the geometric structure of LF images. This branch also can guide the process of reconstruction. Experimental results on both synthetic and real-world datasets demonstrate the superiority of our method. The proposed LF-CFANet has a more advanced reconstruction performance that displays faithful details with higher SR accuracy than state-of-the-art methods.

Keywords light field (LF); super-resolution (SR); attention

1 Introduction

Light field (LF) cameras, e.g., Lytro and RayTrix, provide 4D LF images, unlike conventional cameras, and thus LF imaging technology has been used widely in many applications, such as VR [1, 2], tracking [3–6], 3D reconstruction [6, 7], and saliency detection [8, 9]. As shown in Fig. 1(a), these cameras place a micro-lens array between the main lens and the sensor to provide multiple views of a scene. LF images, captured by a handheld LF camera [1, 2], record spatial information (accumulation from the same object point) and angular information (intensity values for all ray directions). However, due to the limitation of sensor resolution, the spatial resolution of LF images is much lower than that of commercial 2D cameras. Therefore, image super-resolution (SR) technology plays an important role in LF applications, as it effectively enhances the quality of LF images.

LF super-resolution (LFSR) is an ill-posed problem. This problem can be solved by exploring efficient use of sub-pixel information from different views to reconstruct SR images. Traditional methods generally solve the SR problem using multiple views based on prior disparity information, such as a Bayesian framework [10], a variational framework [11, 12], or a Gaussian mixture framework [13]. However, these methods suffer from inaccurate prior disparity

1 Faculty of Applied Sciences, Macao Polytechnic University, Macao SAR 999078, China. E-mail: W. Zhang, wei.zhang@mpu.edu.mo; W. Ke, wke@mpu.edu.mo (✉).

2 State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and Beihang Hangzhou Innovation Institute Yuhang, Xixi Octagon City, Yuhang District, Hangzhou 310023, China. E-mail: D. Yang, da.yang@buaa.edu.cn (✉); H. Sheng, shenghao@buaa.edu.cn; Z. Xiong, xiongz@buaa.edu.cn.

Manuscript received: 2022-02-24; accepted: 2022-06-11

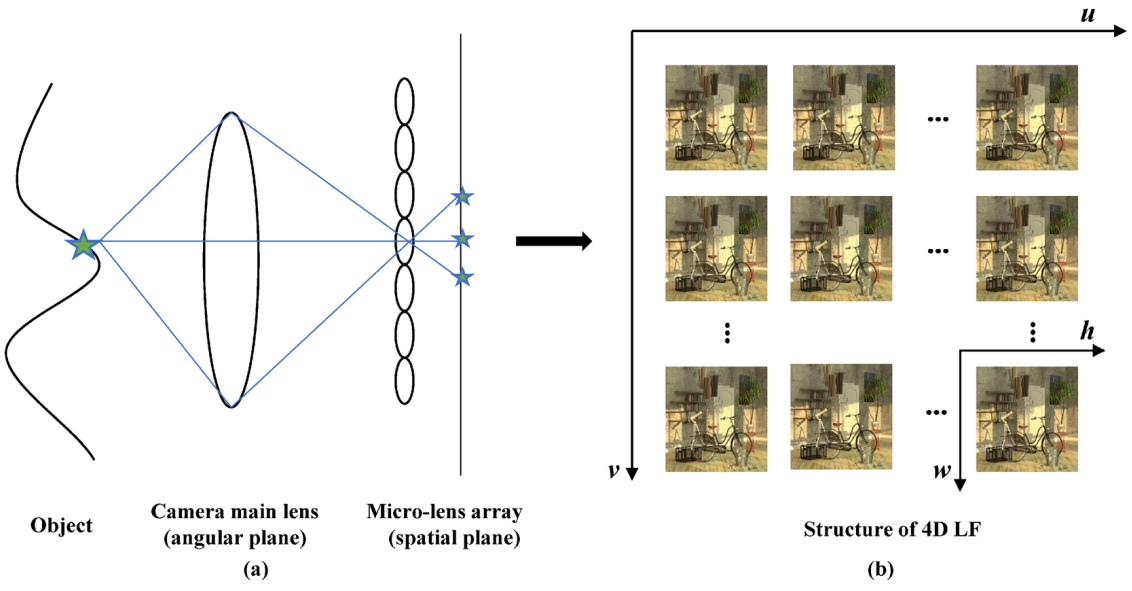

Fig. 1 Principle of LF camera and structure of 4D LF. (a) Schematic LF camera. (b) 4D LF structure. The angular position (u, v) in an LF is determined by the number of sensor pixels under each micro-lens, while the spatial position (h, w) is related to the number of micro-lenses in the array.

estimation and high computational cost. With the development of deep learning, learning-based methods [14–17] have been used to address the problem of the complex 4D structure of LF data and to improve results compared to traditional approaches. Although improvements have been continuously made [18, 19], the inherent complementary information provided by sub-aperture images (SAI) is still not fully utilised, because parallax information is treated equally for each view, and feature fusion between complementary views is inadequate. These issues limit improvement of LFSR methods.

Taking advantage of the attention mechanism in SR networks [20–22], we propose a spatial and channel attention network, namely the light field complementary-view feature attention network (LF-CFANet), to improve the spatial resolution of LF images. As shown in Fig. 2, this network consists of two main modules, the residual complementary-view spatial and channel attention module (RCSCAM) and the maximum-difference information supplementary branch (MDISB). Specifically, the RCSCAM is designed to fuse the complementary information among pairs of LF images. With RCSCAM, reconstruction features can be combined with complementary sub-pixel information and local similarity information from different auxiliary views by computing an attention map. Meanwhile, providing this module with a channel attention mechanism allows it to capture global channel-level information by adaptively adjusting the response of the feature map for each channel. To guide LF reconstruction both effectively and efficiently, the MDISB is designed to obtain maximum-difference information from LF views. In MDISB, the features of a reference view and four auxiliary views are collected from a reservoir based on the maximum-difference angular positions. The maximum-difference feature is used to guide reconstruction of the reference view. Through these designs, the complementary information across LF views can be effectively utilized to reconstruct SR LF images to a certain extent.

Extensive experimental results using real-world and synthetic LF datasets demonstrate that the proposed method achieves quantitatively and qualitatively better results than state-of-the-art methods. Our contributions are summarized as follows:

1. We propose an RCSCAM to better exploit correlation cues for LF complementary-view pairs and generate effectively fused complementary-view features by introducing an attention mechanism. The RCSCAM consists of two types of attention: channel attention and spatial attention. Channel attention enhances the global perception of feature channels, while spatial attention strengthens the interaction of spatial information between complementary views.
2. We develop an MDISB to guide supple-mentation of the most informative difference for SR views by treating each perspective unequally. The

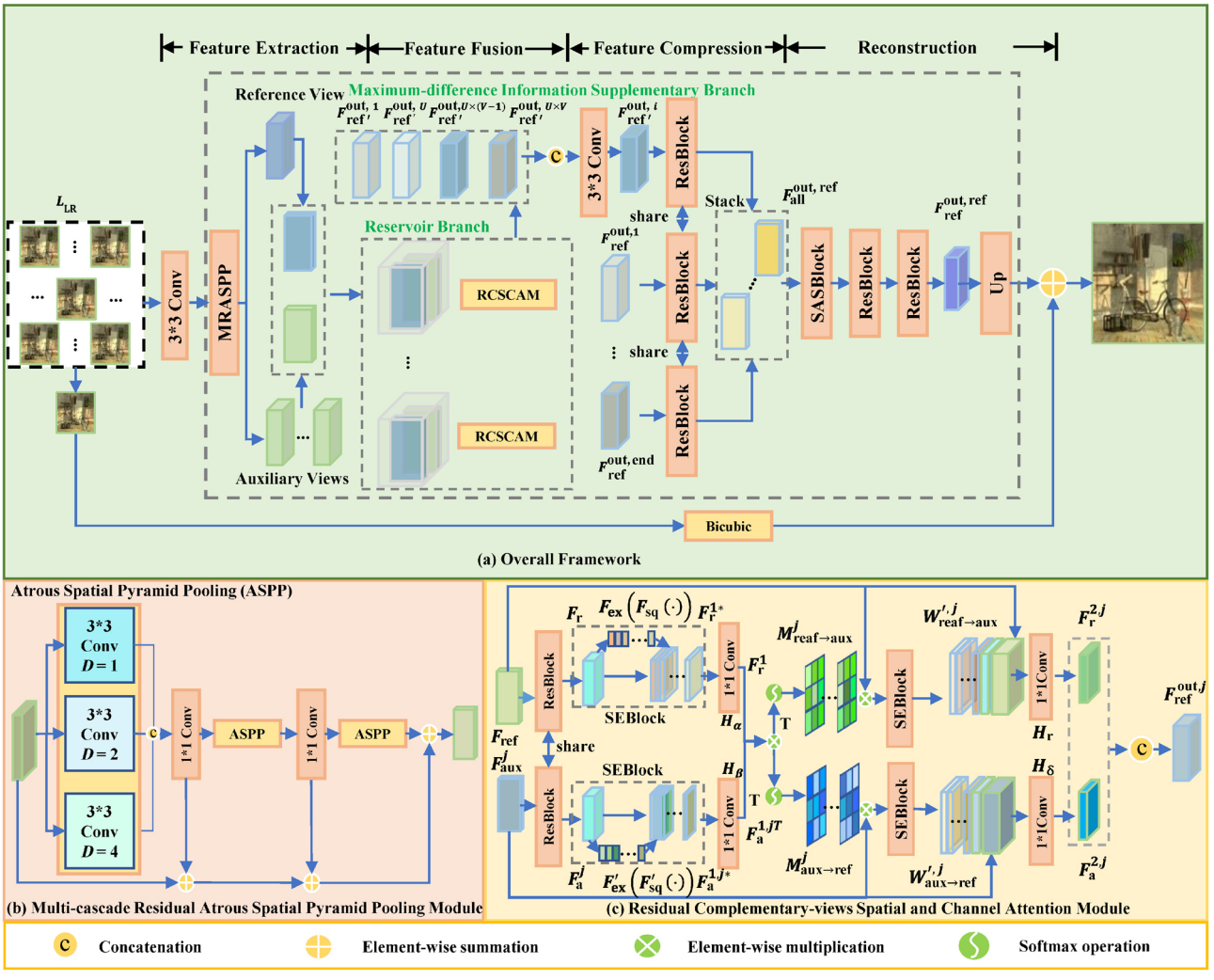

Fig. 2 Network architecture of the proposed LF-CFANet, comprising four parts: feature extraction, feature fusion, feature compression, and reconstruction. The input of our network is made up of sub-aperture images (SAIs). One view is randomly selected from the SAIs as a reference view, and the remaining images are taken as auxiliary view images. The output is an enhanced resolution version of the reference view. C denotes concatenation.

information is provided from a reservoir by concatenating two feature pairs consisting of four maximum-difference fused features based on the angular position of LF images.

3. Our LF-CFANet can exploit the complementary information and the local similarity information from different auxiliary views at the pixel level based on the geometric characteristics of LF images. It uses attention maps to fuse these information for enhancing the spatial resolution. Extensive experiments demonstrate the design effectiveness and improved results compared to state-of-the-art methods.

The rest of this paper is organized in the following sections. Section 2 gives a brief review of related work. Light fields and the architecture of our LF-CFANet are outlined in Section 3. We provide extensive analysis and experiments in Section 4, using synthetic and real-world datasets. Finally, Section 5 concludes the paper.

2 Related work

In this section, we review related work on both single image super-resolution (SISR) and LFSR.

2.1 Single image super-resolution

SISR is a reconstruction technology from fuzzy low-resolution (LR) images. This technology plays an important role in the field of surveillance, satellite imaging, microscope imaging, etc. Several studies

have reviewed SISR in detail [23, 24]. Here, we give a review of several recent advances. Nowadays, deep learning has gradually become a dominant approach for SISR, and has significantly improved the reconstruction quality over traditional methods. Dong et al. [25, 26] proposed a milestone study with an SR deep convolutional neural network (SRCNN), which was a seminal method in the field of SR. This simple and shallow model outperformed earlier work. Kim et al. [27] proposed a very deep convolutional network (VDSR) combined with residual learning, which was more efficient and achieved higher quality than Dong et al.'s work [25, 26]. Note that, VDSR obtained a larger receptive field by stacking filters, and the problem of slow convergence was solved by applying global residual learning. To better exploit intra-view information, more powerful models have been developed based on deep networks. Lim et al. [28] proposed an enhanced deep SR network (EDSR), which achieved extraordinarily better results than previous methods by revising the residual module and multi-scale model [29]. Zhang et al. [30, 31] proposed a residual dense network (RDN), which fully utilized all hierarchical features in all convolutional layers and provided better feature extraction than EDSR. Applying an attention mechanism, Zhang et al. [32] proposed a residual channel attention network (RCAN), which worked by inserting a channel attention module to consider the interdependence of channels. Recently, Dai et al. [33] proposed a second-order attention network (SAN) by applying a trainable second-order attention module to capture spatial information. Both RCAN and SAN have achieved promising results in SISR reconstruction.

As shown in the above review, SISR methods efficiently and effectively reconstruct the spatial information for single images. However, these methods cannot directly handle correlations between multiple views, so cannot be applied to the field of LFSR.

2.2 LF super-resolution

For LFSR, a straightforward approach is to fine-tune the network parameters of SISR. However, LFSR requires using complementary information from multiple LF images of one scene to reconstruct a high-resolution image. Existing LFSR methods can be mainly divided into optimization-based approaches and learning-based approaches.

Optimization-based approaches reconstruct SR images based on the estimated disparities between different views. Bishop and Favaro [10] first used a Bayesian framework for LFSR. Wanner and Goldluecke [11, 12] proposed a variational method for SR by introducing disparity maps obtained from EPIs. Mitra and Veeraraghavan [13] proposed a patch-based approach modeled by a Gaussian mixture model to solve LF problems. The framework of this method could handle many different processing tasks. Zhang et al. [34] proposed the PlenoPatch method based on patches to generate more realistic results than previous methods. To better supplement complementary information and avoid costly disparity estimation, Rossi and Frossard [35] proposed an LFSR framework for homogeneous reconstruction of all views in the LF by using a graph-based regularizer. Later, Alain and Smolic [36] proposed a method to convert the inverse problem of LFSR into an optimization problem based on prior sparsity. Although these methods could well encode the complex 4D LF, optimization-based methods are not effective in combining the spatial information from different views. Moreover, most of these methods are based on handcrafted image priors, which limit the quality of reconstruction.

Learning-based approaches show superiority over optimization-based approaches in using complementary information from different views. Complementary information can improve the quality of LFSR. Yoon et al. [15, 16] introduced CNNs to the field of LF (LFCNN), while Yuan et al. [14] proposed an SR method that fully exploited the structure of the LF with an SISR module and an EPI enhancement module. These modules captured the structural characteristics of LF well. By extending BRCN [37], Wang et al. [38] proposed a bidirectional recurrent convolutional neural network, LFNet, and stacked generalization techniques to synthesize the final sub-aperture images. In this structure, the recurrent neural network was improved to handle horizontal and vertical structures. In this network, spatial correlations between neighboring views could be modified to be more effective and flexible. Inspired by the residual network, Zhang et al. [18] proposed a multi-branch residual network (resLF) to handle image stacks with consistent sub-pixel offsets; each

branch could extract high-frequency details from LF images. In order to preserve the parallax structure, Jin et al. [39] proposed a method with two-step LF spatial resolution by introducing a perspective feature fusion module and a structural consistency regularization loss (LF-ATO). More recently, Wang et al. [40] proposed an LF-InterNet to extract and incorporate spatial and angular information. This network could gradually combine the spatial and angular information. Mo et al. [41] proposed a dense dual-attention network (DDAN), with spatial attention within LF views and channel attention within different channels. This method achieved high-quality LF reconstruction.

In summary, these methods implicitly learn the internal correspondences of the LF structure, and they are gradually improving LFSR. However, due to the design of the network structure, complementary information is still not fully utilized. For example, LFNet uses a bidirectional recurrent network to fuse angular information among SAIs. This information only considers row and column directions, and it cannot be efficiently used to reconstruct LF images. Instead, we propose a complementary-view feature attention approach that uses the information from all auxiliary views to reconstruct the reference view.

3 Architecture of LF-CFANet

In this section, we introduce the 4D LF representation, and propose a many-to-one LFSR network. The architecture of our LF-CFANet is shown in Fig. 2. The feature fusion part is composed of two branches, MDISB and a reservoir branch.

3.1 Problem formulation

A 4D LF can be parameterized by two parallel planes. As shown in Fig. 1(b), the spatial plane, $\Pi = (h, w)$, and the angular plane, $\Omega = (u, v)$, are used to describe the structure of a 4D LF. These planes can accurately represent the light images $\mathcal{L}(\Pi, \Omega)$. Following existing LFSR methods [39], we only use Y channel images as input, which are obtained by converting input RGB images to YCbCr images, and retaining only the Y channel [39]. The input can be denoted $L_{\mathrm{LR}} \in \mathbb{R}^{U\times V\times H\times W}$, ignoring the channel dimension. The goal of the LFSR task can be described as generating an SR LF from the LR input LF. The reconstructed LF images are $L_{\mathrm{SR}} \in \mathbb{R}^{U\times V\times \alpha H\times \alpha W}$, where α is the upsampling rate.

3.2 Feature extraction

The quality of discriminative features with rich contextual information is very useful to SR reconstruction. This information can be obtained by using a multi-scale receptive field and feature learning. Therefore, the feature-extraction module of our LF-CFANet follows [21, 48] and uses atrous spatial pyramid pooling (ASPP) module to extract the LF image features.

Figure 2 shows the overall network architecture of the proposed LF-CFANet. The input L_{LR} is composed of SAIs. The initial features (with 64 channels) of L_{LR} are extracted by a 3×3 convolution shown in Fig. 2(a), and then we use the multi-cascaded residual ASPP (MRASPP) module in Fig. 2(b) for multi-scale feature extraction to support downstream processing. Specifically, the initial features of the LF views are first fed to the ASPP blocks, which share weights for each view. Each ASPP block consists of three different dilated convolutions with a leaky ReLU layer. These dilated convolutions, with dilation rates (D) 1, 2, and 4, are used to extract L_{LR} features with different receptive fields. After a leaky ReLU layer, we concatenate the three output features and compress the number of channels through 1×1 convolution to make them more compact. These ASPP blocks not only obtain multi-receptive fields without changing the size of the feature maps, but also enrich the diversity of the convolutions. After three cascaded residual ASPP blocks, the feature of each view is extracted. These features can be expressed as

$$\left\{F^i_{\mathrm{each}} \mid i = 1, \ldots, n\right\} = f_0\left(L_{\mathrm{LR}}\right) \tag{1}$$

where f_0 represents the MRASPPBlock and n is the number of SAIs.

For the output of MRASPPBlock, F^n_{each}, the reference feature is randomly selected from the n output features, and the auxiliary features are the remaining features. These two types of features can be specifically expressed as

$$\begin{cases} F_{\mathrm{ref}} = F^i_{\mathrm{each}} \\ F^j_{\mathrm{aux}} = F^j_{\mathrm{each}} \end{cases} \tag{2}$$

where i, j ($1 \leqslant i, j \leqslant U\times V$, $i \neq j$, $i+j = n$) represent the angular positions. There is one i-indexed feature, and there are $(U \times V - 1)$ j-indexed features.

As shown in Fig. 2(a), F_{ref} and each F_{aux}^{j} are concatenated to form a feature pair $\{F_{\text{ref}}, F_{\text{aux}}^{j}\}$. Selecting the complementary-view pairs makes the model more compatible with all views and increases the generalization performance of the networks.

3.3 RCSCAM in reservoir branch

The feature fusion part includes two branches. The first branch is a reservoir branch, and the second branch is MDISB. The reservoir branch is the key to fusing auxiliary-view information with reference-view information, using the RCSCAM (Residual Complementary-View Spatial and Channel Attention Module). Inspired by the stereo-attention mechanisms [22, 49], we develop an RCSCAM to supplement the sub-pixel information of the reference view.

As Fig. 2(c) shows, the input pair of features $\{F_{\text{ref}}, F_{\text{aux}}^{j}\}$ are separately fed to two ResBlocks, f_1, with 64 channels. These two ResBlocks share weights. The output features of f_1 are $F_{\text{r}}, F_{\text{a}}^{j} \in \mathbb{R}^{H\times W\times 64}$.

To explore the correlation between feature channels, we introduce SEBlocks following Ref. [20]. Pseudo-code to capture the channel attention is provided in Algorithm 1. This block processes the input feature in three steps: squeeze, excite, and reweight. F_{r} and F_{a}^{j} are respectively fed to globally adaptive pooling $(\boldsymbol{F}_{\text{sq}}^{1}, \boldsymbol{F}_{\text{sq}}^{1,j})$ to obtain feature channels with 1×64 aggregated information. To capture channel-wise dependencies, two fully-connected (FC) layers are used. The output weights of the excitation process represent the importance of the feature channel. They are applied to each channel by multiplication. These processes are denoted $(\boldsymbol{F}_{\text{ex}}^{1}, \boldsymbol{F}_{\text{ex}}^{1,j})$. Then, the outputs are separately fed to 1×1 convolutions to generate the feature maps $(F_{\text{r}}^{1}, F_{\text{a}}^{1,j})$. These outputs can be specifically expressed as

$$\begin{cases} F_{\text{r}}^{1} = H_{\alpha}\left(f_{\text{SE}_1}(F_{\text{r}})\right) \\ F_{\text{a}}^{1,j} = H_{\beta}\left(f_{\text{SE}_2}(F_{\text{a}}^{j})\right) \end{cases} \tag{3}$$

where f_{SE_1} and f_{SE_2} represent the SEBlocks, and H_{α} and H_{β} represent the 1×1 convolutions.

To generate a reference-auxiliary attention map, $F_{\text{a}}^{1,j}$ is first transposed to $(F_{\text{a}}^{1,j})^{\text{T}}$, and then the geometry-aware matrix is multiplied by matrix F_{r}^{1}. The multiplying output of these two matrices is processed by softmax to produce the final attention maps, $\mathcal{M}_{\text{aux}\to\text{ref}}^{j} \in \mathbb{R}^{H\times W\times W}$. Similarly, $\mathcal{M}_{\text{ref}\to\text{aux}}^{j}$ is generated. This process can be expressed as Eq. (4):

$$\begin{cases} \mathcal{M}_{\text{aux}\to\text{ref}}^{j} = \text{Softmax}\left(F_{\text{r}}^{1} \otimes (F_{\text{a}}^{1,j})^{\text{T}}\right) \\ \mathcal{M}_{\text{ref}\to\text{aux}}^{j} = \text{Softmax}\left(F_{\text{a}}^{1,j} \otimes (F_{\text{r}}^{1})^{\text{T}}\right) \end{cases} \tag{4}$$

where $\otimes$ represents batch-wise matrix multiplication.

To achieve feature information combination between the reference view and auxiliary view, $\mathcal{W}_{\text{ref}\to\text{aux}}^{j}$ and $\mathcal{W}_{\text{aux}\to\text{ref}}^{j}$ are generated by multiplying the input pair of features $(F_{\text{ref}}, F_{\text{aux}}^{j})$ and the attention maps $(\mathcal{M}_{\text{ref}\to\text{aux}}^{j}, \mathcal{M}_{\text{aux}\to\text{ref}}^{j})$, respectively. Both $\mathcal{W}_{\text{ref}\to\text{aux}}^{j}$ and $\mathcal{W}_{\text{aux}\to\text{ref}}^{j}$ contain the reference-view and auxiliary-view information. They can be computed using

$$\begin{cases} \mathcal{W}_{\text{ref}\to\text{aux}}^{j} = \mathcal{M}_{\text{ref}\to\text{aux}}^{j} \otimes F_{\text{ref}} \\ \mathcal{W}_{\text{aux}\to\text{ref}}^{j} = \mathcal{M}_{\text{aux}\to\text{ref}}^{j} \otimes F_{\text{aux}}^{j} \end{cases} \tag{5}$$

As Fig. 2 shows, these two features $(\mathcal{W}_{\text{ref}\to\text{aux}}^{j}, \mathcal{W}_{\text{aux}\to\text{ref}}^{j})$ are fed into two new SEBlocks to generate new features $(\mathcal{W}_{\text{ref}\to\text{aux}}^{\prime j}, \mathcal{W}_{\text{aux}\to\text{ref}}^{\prime j})$, respectively.

To retain the original features of the reference and auxiliary views, the input pair of features $(F_{\text{ref}}, F_{\text{aux}}^{j})$ is concatenated with $(\mathcal{W}_{\text{ref}\to\text{aux}}^{\prime j}, \mathcal{W}_{\text{aux}\to\text{ref}}^{\prime j})$, and fed into another 1×1 convolution. This process can be expressed as

$$\begin{cases} F_{\text{r}}^{2,j} = H_{\gamma}\left(\text{Cat}(F_{\text{ref}}, \mathcal{W}_{\text{ref}\to\text{aux}}^{\prime j})\right) \\ F_{\text{a}}^{2,j} = H_{\delta}\left(\text{Cat}(F_{\text{aux}}^{j}, \mathcal{W}_{\text{aux}\to\text{ref}}^{\prime j})\right) \end{cases} \tag{6}$$

where Cat is the concatenation operator, and H_{γ} and H_{δ} represent the 1×1 convolutions to fuse these two

Algorithm 1 Squeeze and excite blocks

Input: Feature pair $\{F_{\text{r}}, F_{\text{a}}^{j}\} \in \mathbb{R}^{h\times w\times 64}$

1: **Squeeze**: Feature $(F_{\text{r}}, F_{\text{a}}^{j})$ is compressed in the spatial dimension.
For each channel, compute
$\boldsymbol{F}_{\text{sq}}^{1}(F_{\text{r}}) = (1/WH)\sum_{w=1}^{W}\sum_{h=1}^{H} F_{\text{r}}(w,h)$;
For each channel, compute
$\boldsymbol{F}_{\text{sq}}^{1,j}\left(F_{\text{a}}^{j}\right) = (1/WH)\sum_{i=w}^{W}\sum_{h=1}^{H} F_{\text{a}}^{i}(w,h)$;
Each two-dimensional (H, W) feature channel becomes a number, which has a global receptive field.

2: **Excite and Reweight**: Each feature channel generates a weight to represent its importance. The weight of the output of Excite is regarded as the importance of each feature channel, and is applied to each channel by multiplication.
Compute $F_{\text{r}}^{1\star} = \boldsymbol{F}_{\text{ex}}^{1}\left(\boldsymbol{F}_{\text{sq}}^{1}(F_{\text{r}})\right)$;
Compute $F_{\text{a}}^{1,j\star} = \boldsymbol{F}_{\text{ex}}^{1,j}\left(\boldsymbol{F}_{\text{sq}}^{1,j}(F_{\text{a}}^{j})\right)$;

Output:
$\{F_{\text{r}}^{1\star}, F_{\text{a}}^{1,j\star}\} \in \mathbb{R}^{h\times w\times 64}$

types of features. $F_{\mathrm{r}}^{2,j}$ and $F_{\mathrm{a}}^{2,j}$ represent the fully RCSCAM fused features of each pair.

The combined features of complementary views are generated in this process. These four SEBlocks can express valid information for reconstruction. The result $F_{\mathrm{ref}}^{\mathrm{out},j}$, fully integrating the complementary information, can be expressed as

$$F_{\mathrm{ref}}^{\mathrm{out},j} = \mathrm{Cat}\left(F_{\mathrm{a}}^{2,j}, F_{\mathrm{r}}^{2,j}\right) \tag{7}$$

In the training process, the reference view feature F_{ref} is generated by randomly selecting from the initial features. Due to the complex geometric structure of LF images, the fusion features $F_{\mathrm{ref}}^{\mathrm{out},j}$ obtained by RCSCAMs contain complementary information and local similarity information from different auxiliary views. The principle of RCSCAM is to obtain the feature similarities for all possible disparities between each pixel in the reference view and auxiliary view to generate an attention map. By introducing the attention mechanism, we can fully fuse the complementary information through feature-level information for reconstructing SR. The effectiveness of RCSCAM is demonstrated in Section 4.3.

3.4 MDISB

As the second branch of our feature fusion, MDISB (Maximum-Difference Information Supplementary Branch) is used to select four maximum-difference fusion features to guide reference view reconstruction. This branch chooses the four fusion features with maximum-difference information relative to the reference view from the reservoir. After RCSCAM, each pair of the reference view and an auxiliary view generates one fusion feature. The total number of fusion features is $n_1 = U \times V - 1$. Due to the parallax structure of LF, the difference information in each auxiliary view varies, and supplements the reference-view information. The four angular-position initial features $[F_{\mathrm{each}}^{1}, F_{\mathrm{each}}^{U}, F_{\mathrm{each}}^{U\times(V-1)}, F_{\mathrm{each}}^{U\times V}]$ generated by the MRASPP block have maximum-difference information compared to the reference view. These four features are concatenated with the reference-view feature and fed into RCSCAM. The output of these four features through RCSCAM is $\left[F_{\mathrm{ref}'}^{\mathrm{out},1}, F_{\mathrm{ref}'}^{\mathrm{out},U}, F_{\mathrm{ref}'}^{\mathrm{out},U\times(V-1)}, F_{\mathrm{ref}'}^{\mathrm{out},U\times V}\right]$. We then use the concatenation operator Cat to combine the output from RCSCAM. This MDISB process can be expressed as

$$F_{\mathrm{ref}'}^{\mathrm{out},i} = \mathrm{Cat}\begin{pmatrix} F_{\mathrm{ref}'}^{\mathrm{out},1} & F_{\mathrm{ref}'}^{\mathrm{out},U} \\ F_{\mathrm{ref}'}^{\mathrm{out},U\times(V-1)} & F_{\mathrm{ref}'}^{\mathrm{out},U\times V} \end{pmatrix} \tag{8}$$

where $F_{\mathrm{ref}'}^{\mathrm{out},i}$ represents the output of our MDISB for the reference-view position, while the input $[F_{\mathrm{ref}'}^{\mathrm{out},1}, F_{\mathrm{ref}'}^{\mathrm{out},U}, F_{\mathrm{ref}'}^{\mathrm{out},U\times(V-1)}, F_{\mathrm{ref}'}^{\mathrm{out},U\times V}]$ represents the fusion features that supplement the complementary-view information to F_{ref} by using RCSCAM. As Fig. 2(a) shows, we concatenate these four features and compress them using a 3×3 convolution. The depth of the final feature is 64.

3.5 Feature compression

Feature compression can compress the feature depth to adapt to the part of the reconstruction. We use ResBlocks to process each feature, which are $F_{\mathrm{ref}}^{\mathrm{out},1}, \ldots, F_{\mathrm{ref}}^{\mathrm{out},j}, F_{\mathrm{ref}'}^{\mathrm{out},i}, F_{\mathrm{ref}}^{\mathrm{out},j+1}, \ldots, F_{\mathrm{ref}}^{\mathrm{out,end}}$ from two branches. All ResBlocks share the same parameters. We stack these features from all auxiliary views and train them to integrate the complementary information from RCSCAM, and the maximum-difference information from MDISB. The output of feature compression can be written as

$$F_{\mathrm{all}}^{\mathrm{out,ref}} = \mathrm{Stack}\begin{pmatrix} F_{\mathrm{ref}}^{\mathrm{out},1} & \cdots & F_{\mathrm{ref}}^{\mathrm{out},j} \\ F_{\mathrm{ref}'}^{\mathrm{out},i} & & \\ F_{\mathrm{ref}}^{\mathrm{out},j+1} & \cdots & F_{\mathrm{ref}}^{\mathrm{out,end}} \end{pmatrix} \tag{9}$$

where Stack represents feature stacking.

3.6 Reconstruction

Inspired by the architecture of Ref. [50] for SISR, we use a similar structure to reconstruct the SR images. Following the method of Ref. [19], the feature $F_{\mathrm{all}}^{\mathrm{out,ref}}$ from the compression module is first reshaped and processed by a SASBlock. The SASBlock is repeated 3 times to integrate angular and spatial domain information. The output feature is fed into two ResBlocks with 64 channels. One ResBlock (with two residual blocks) provides channel-wise view fusion. The other ResBlock (with three residual blocks) provides channel fusion to generate the final reference-view feature $F_{\mathrm{ref}}^{\mathrm{out,ref}}$.

To save memory and computation, we utilize an up-sampling block $\mathrm{Up}(\cdot)$ to increase the resolution of the reference-view image $L_{\mathrm{SR}}^{\mathrm{ref}}$. This block, inspired by Ref. [18], is composed of a convolution layer, a shuffle layer, and a convolution layer in order. Finally, $L_{\mathrm{SR}}^{\mathrm{ref}}$ is generated by adding the residual map to the

up-sampled image. The reconstruction image for one angular position can be expressed as

$$L_{\mathrm{SR}}^{\mathrm{ref}} = \mathrm{Up}\left(F_{\mathrm{ref}}^{\mathrm{out,ref}}\right) \tag{10}$$

where Up represents the process of reconstruction.

To simplify our LFSR network, we follow the approach in Fig. 3. We randomly choose a view as the reference view and feed all views into our network. Through feature extraction, feature fusion, feature compression, and reconstruction, our network can fully learn the differential sub-pixel information from the auxiliary views. The information can be added to the reference view for reconstruction.

4 Experiments

In this section, we first introduce the datasets used and implementation details. Then, we compare our LF-CFANet with several state-of-the-art SISR and LFSR methods. Finally, we conduct ablation studies to evaluate the contribution of individual component modules in our network.

4.1 Experimental setup

4.1.1 Datasets

Our LF images come from both synthetic datasets and real-world datasets. The real-world datasets were captured by various devices with different baseline lengths. Therefore, LF algorithms should be able to adapt to different datasets. As listed in Table 1, 6 public LF datasets (EPFL [42], HCInew [43], HCIold [44], INRIA [45], STFgantry [46], and STFlytro [47]) were used for training and testing in our experiments, which include a total of 394 LF scenes for training

Table 1 Public LF datasets used in our experiments. R = real-world scene, S = synthetic scene

Dataset	Training	Test	Disparity	Kind
EPFL [42]	70	10	[−1, 1]	R
HCInew [43]	20	4	[−4, 4]	S
HCIold [44]	10	2	[−3, 3]	S
INRIA [45]	35	5	[−1, 1]	R
STFgantry [46]	9	2	[−7, 7]	R
STFlytro [47]	250	50	—	R
Total	394	73		

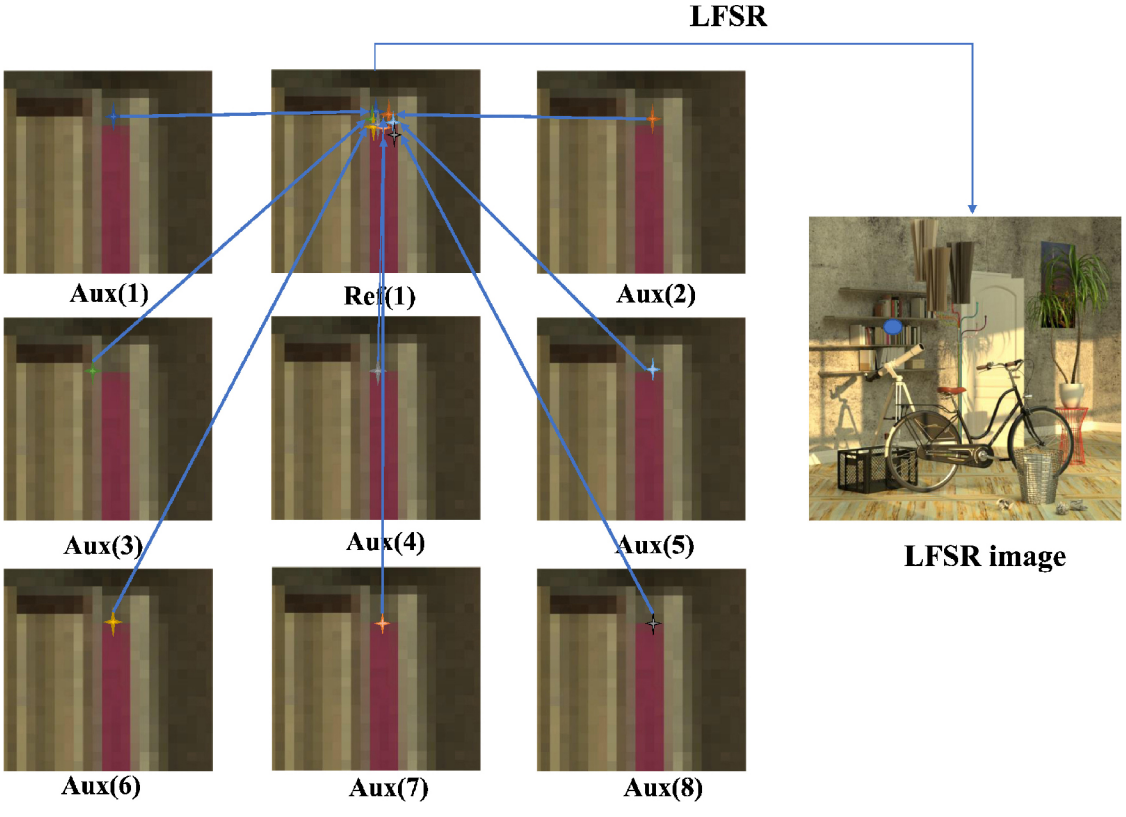

Fig. 3 Supplementing sub-pixel information. Here, an L_{LR} ($\mathbb{R}^{3\times3\times w\times h}$) is used as an example. We randomly choose a reference view ($U = 2, V = 1$), and the remaining views are used as auxiliary views. Sub-pixel information from different auxiliary views is visually represented as stars with different colors for clarity. The information is added to the blue dot in the LFSR image.

and 73 LF scenes for testing. The EPFL, INRIA, and STFlytro data contain rich outdoor-scenes captured with a Lytro Illum camera, while the HCInew, HCIold, and STFgantry data contain indoor LF images. The LF disparity of these datasets varies and the angular resolution is 9×9 for all LF datasets. We generate LR LF images by bicubic interpolation for both training and testing.

4.1.2 Implementation details

Our network has two types of convolutional layers, which are 3×3 and 1×1. All the 3×3 convolutional layers were zero-padded to retain the spatial resolution, and we set the number of Resblocks to 2, 2, and 3 residual blocks in order. The feature depths of residual blocks were all 64.

In the training stage, we randomly cropped the input LF images to a spatial size of 64×64 and randomly processed by flipping the images horizontally or vertically and rotating them by 90°. The upscaling factor r was 2 or 4, and we respectively trained the network with different factors. We train our network with the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$). The initial learning rate was set to 10^{-4} and decreased by a factor of 0.5 every 250 epochs. Training of the full LF-CFANet was stopped after 600 epochs.

4.2 Comparison to state-of-the-art

4.2.1 Setting

We compared our LF-CFANet with recent state-of-the-art SISR and LFSR methods: VDSR [50], EDSR [28], GB [35], RCAN [32], SAN [33], LFBMD5D [36], resLF [18], LFSSR [19], LF-ATO [39], and LF-InterNet [40]. For a fair comparison, these methods were re-trained on the same training dataset as our method. We chose bicubic interpolation as a baseline for comparison. For evaluation, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) were used as quantitative quality metrics, with higher values indicating better LF reconstruction results.

4.2.2 Quantitative comparison results

A quantitative evaluation of PSNR and SSIM at 5×5 angular resolution for the 6 test datasets is given in Table 2. Our method achieved higher PSNR and SSIM than the RCAN [32] SISR method. Specifically, our method had an average PSNR increase of 1.7 dB (×2) and 1.2 dB (×4) in the test. That is because complementary information can be used effectively in

Table 2 PSNR and SSIM values achieved by different methods for 2× and 4× SR. Red: best results. Blue: second best results

Method	Scale	EPFL	HCInew	HCIold	INRIA	STFgantry	STFlytro
Bicubic	×2	29.50/0.935	31.69/0.934	37.46/0.978	31.10/0.956	30.82/0.947	33.02/0.950
VDSR [50]	×2	32.01/0.959	34.37/0.956	40.34/0.985	33.80/0.972	35.80/0.980	35.91/0.970
EDSR [28]	×2	32.86/0.965	35.02/0961	41.11/0.988	34.61/0.977	37.08/0.985	36.84/0.975
GB [35]	×2	31.22/0.959	35.25/0969	40.21/0.988	32.76/0.972	35.44/0.983	35.04/0.956
RCAN [32]	×2	33.46/0.967	35.56/0.963	41.59/0.989	35.18/0.978	38.18/0.988	37.32/0.977
SAN [33]	×2	33.36/0.967	35.51/0.963	41.47/0.989	35.15/0.978	37.98/0.987	37.26/0.976
LFBMD5D [36]	×2	31.15/0.955	33.72/0.955	39.62/0.985	32.85/0.969	33.55/0.972	35.01/0.966
resLF [18]	×2	33.22/0.969	35.79/0.969	42.30/0.991	34.86/0.979	36.28/0.985	35.80/0.970
LFSSR [19]	×2	34.15/0.973	36.98/0.974	43.29/0.993	35.76/0.982	37.67/0.989	37.57/0.978
LF-ATO [39]	×2	34.49/0.976	37.28/0.977	43.76/0.994	36.21/0.984	39.06/0.992	38.27/0.982
LF-InterNet [40]	×2	34.76/0.976	37.20/0.976	44.65/0.995	36.64/0.984	38.48/0.991	38.81/0.983
Ours	×2	34.92/0.976	37.46/0.977	44.16/0.994	36.81/0.985	39.48/0.992	38.91/0.983
Bicubic	×4	25.14/0.831	27.61/0.851	32.42/0.934	2682/0.886	25.93/0.843	27.84/0.855
VDSR [50]	×4	26.82/0.869	29.12/0.876	34.01/0.943	28.87/0.914	28.31/0.893	29.17/0.880
EDSR [28]	×4	27.82/0.892	29.94/0.893	35.53/0.957	29.86/0.931	29.43/0.921	30.29/0.903
GB [35]	×4	26.02/0.863	28.92/0.884	33.74/0.950	27.73/0.909	28.11/0.901	28.37/0.873
RCAN [32]	×4	28.31/0.899	30.25/0.896	35.89/0.959	30.36/0.936	30.25/0.934	30.66/0.909
SAN [33]	×4	28.30/0.899	30.25/0.898	35.88/0.960	30.29/0.936	30.25/0.934	30.66/0.909
LFBMD5D [36]	×4	26.61/0.869	29.13/0.882	34.23/0.951	28.49/0.914	28.30/0.900	29.07/0.881
resLF [18]	×4	27.86/0.899	30.37/0.907	36.12/0.966	29.72/0.936	29.64/0.927	28.94/0.891
LFSSR [19]	×4	29.16/0.915	30.88/0.913	36.90/0.970	31.03/0.944	30.14/0.937	31.21/0.919
LF-ATO [39]	×4	29.16/0.917	31.08/0.917	37.23/0.971	31.21/0.950	30.78/0.944	30.98/0.918
LF-InterNet [40]	×4	29.52/0.917	31.01/0.917	37.23/0.972	31.65/0.950	30.44/0.941	31.84/0.927
Ours	×4	29.58/0.917	31.24/0.918	37.24/0.972	31.89/0.951	31.05/0.948	31.99/0.928

the context of LFs. Moreover, our method achieved the best results on both real-world datasets (EPFL, INRIA, STFgantry) and synthetic datasets (HCInew, HCIold). That is because our LF-CFANet is based on feature fusion driven by the attention mechanism, which is sensitive to the disparity.

Due to different angular resolutions, the PSNR of each view in SAIs is not identical. Figures 5 and 6 compare the PSNR of individual SAIs for LFSSR, ATO, LF-InterNet, and our method. Compared with the same many-to-one approach (LF-ATO), our approach shows significant performance improvements, as shown by Fig. 6. Although LFSSR, LF-ATO, and LF-InterNet can use the angular information from all input views to super-resolve each view, the gap among maximum-difference views of our method is much smaller than for other methods. That is because our method introduces MDISB to reduce the information degradation of maximum-difference views. The reconstruction quality of LF-CFANet is slightly higher than those of other LFSR methods. The computational load of some state-of-the-art methods (EDSR, resLF, LF-ATO, LF-InterNet) is presented in Table 4. Note that, our method consumes less computational resources but achieves the best results compared with SISR and LFSR methods, especially for the reference view. Our LF-CFANet has higher computational efficiency than LF-InterNet, because the structure of LF-InterNet processes with all LR LF simultaneously. However, our method reconstructs the reference view by supplementing the complementary information from auxiliary views. Note that the PSNR for our reference view is much higher than the average PSNR for LF-InterNet: see Fig. 5.

4.2.3 Qualitative comparison

We provide a visual comparison of results of different methods in Fig. 4 for ×2 and ×4 LFSR. Our LF-CFANet can recover fine details and textures, such as the letters in STFgantry_cards. However, the other methods lose most high-frequency details in the reconstructed results. Compared with our method, VDSR and SAN, state-of-the-art SISR methods, produce poor details, because they lack

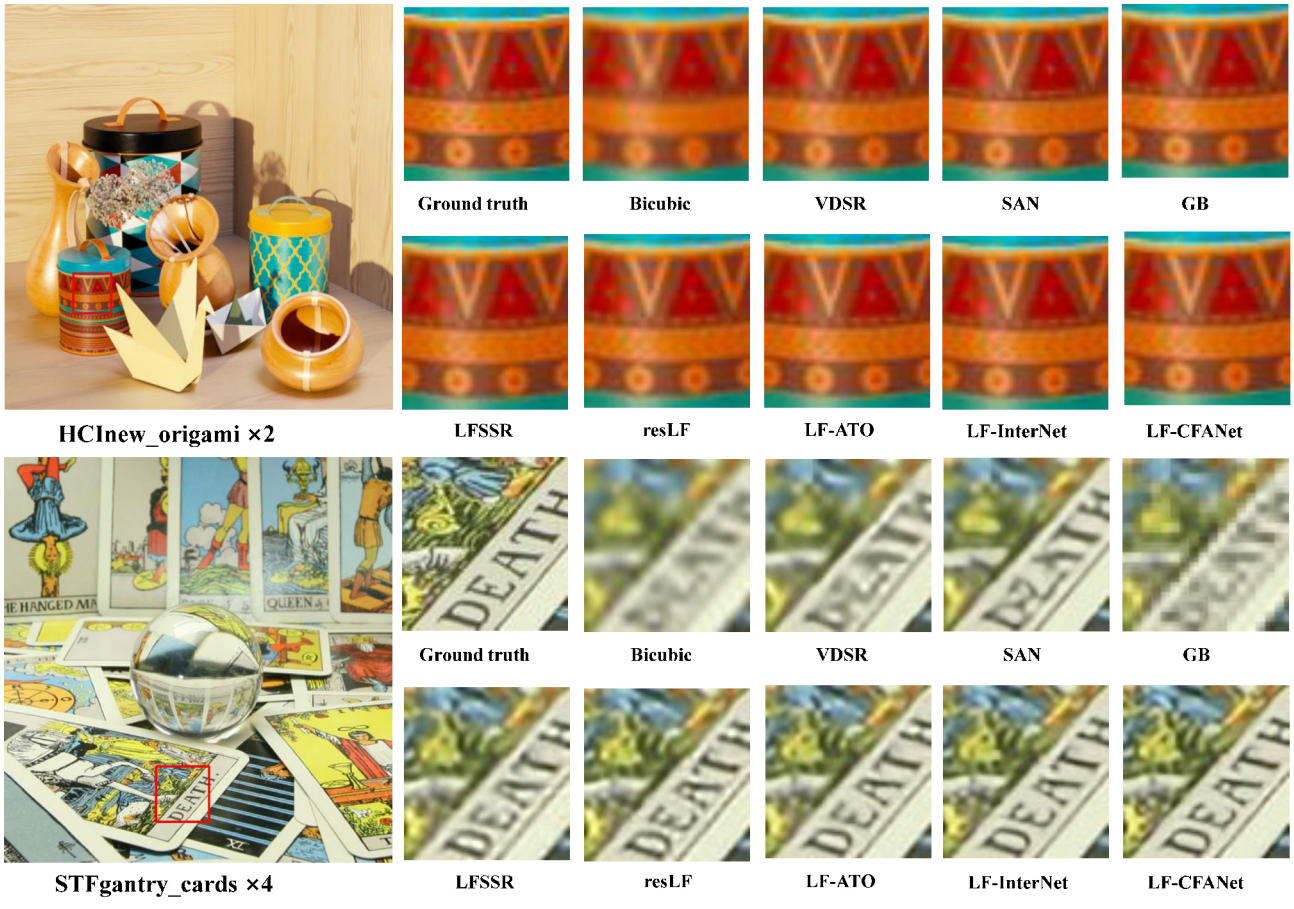

Fig. 4 Results of different methods for 2× and 4× reconstruction.

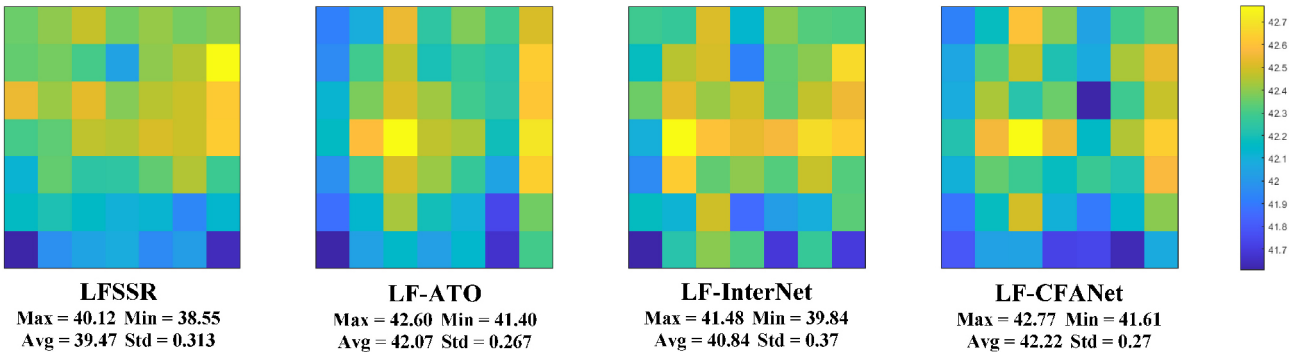

Fig. 5 PSNR comparison for individual SAIs. Here, 7×7 input views are used to perform 2× LFSR. We use standard deviation (Std) to assess their uniformity.

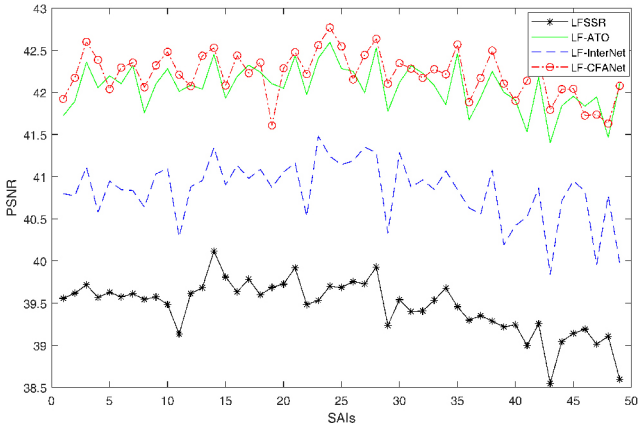

Fig. 6 Comparison of PSNR of individual SAIs.

the complementary information to supplement image reconstruction. Although resLF, LF-SAS, LF-ATO, and LF-InterNet methods generate better results than SISR methods, they do not effectively make use of complementary information in the LFSR process. Our method can effectively and efficiently reconstruct LF images by using a channel and spatial attention mechanism. Figures 7 and 8 further demonstrate the visual comparisons of LF parallax structure of LFSR methods: our method produces clearer and straighter lines than the other LFSR methods. Our method can preserve the structural characteristics of LFs.

4.3 Ablation study

We conducted several experiments to evaluate the results using different architectures.

4.3.1 Effectiveness of MRASPP

The MRASPP is used to extract discriminative features. We used variants of LF-CFANet (onlyMRASPP and rmMRASPP) to show the effectiveness of the MRASPP. The results are given in Table 3. As expected, removing MRASPP caused rmMRASPP to suffer a decrease (of 0.06 dB) in PSNR. That is because MRASPP extracts feature at different scales, which can make the feature representations more robust. Moreover, discriminative features with rich context information can be extracted by using the multiple receptive fields of atrous convolutions. Therefore, our model can obtain accurate features to reconstruct LF.

Table 3 Results for different LF-CFANet architectures on the STFlytro data for 2× upscaling. The bicubic result provides a baseline. onlyMRASPP, onlyMDISB, and onlyRCSCAM denote that only MRASPP, MDISB, and RCSCAM blocks are used in our LF-CFANet, while rmMRASPP, rmMDISB, and rmRCSCAM mean that only MRASPP, MDISB, and RCSCAM blocks are removed from LF-CFANet

Architecture	PSNR	SSIM	Parameters
Bicubic	33.02	0.950	—
onlyMRASPP	38.62	0.980	2.32M
rmMRASPP	38.85	0.982	2.63M
onlyMDISB	38.71	0.982	2.54M
rmMDISB	38.87	0.983	2.41M
onlyRCSCAM	38.71	0.982	2.05M
rmRCSCAM	38.78	0.983	2.91M
LF-CFANet	**38.91**	**0.983**	**3.00M**

Table 4 Number of parameters and FLOPs are provided for 2× SR. Note that, FLOPs is computed with an input LF of size $5 \times 5 \times 32 \times 32$ for an LF dataset

Network	MParams	GFLOPs	PSNR
EDSR	38.62	39.56×25	37.08 dB
resLF	6.35	37.06	37.6 dB
LF-ATO	1.22	28.24×25	39.06 dB
LF-InterNet	4.80	47.46	38.48 dB
LF-CFANet	3.00	47.68×25	**39.48 dB**

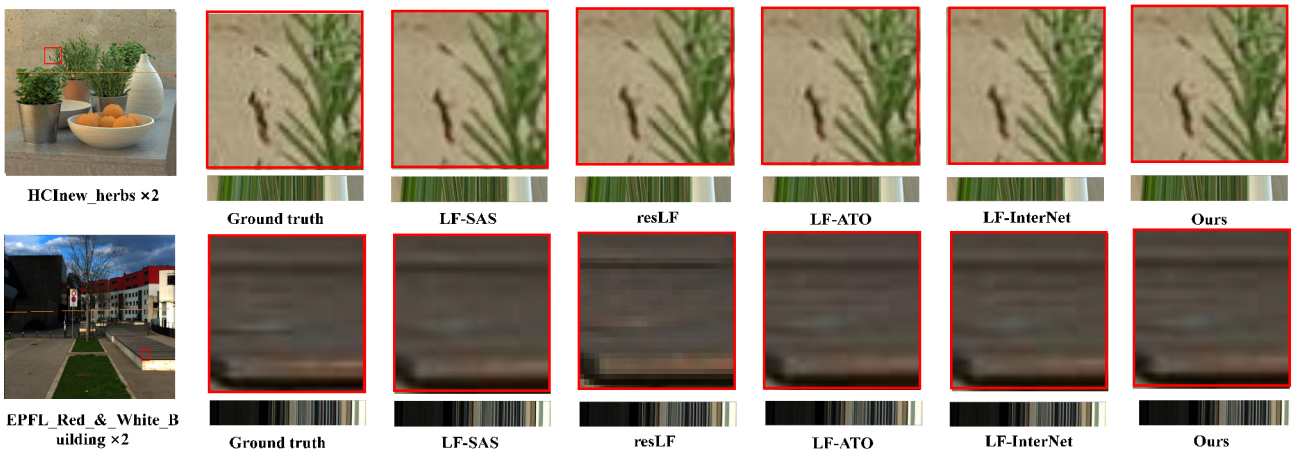

Fig. 7 Results of different methods for 2× reconstruction for both synthetic and real-world scenes, showing predicted central SAIs, close-ups of the framed patches, and EPIs at the colored lines.

Fig. 8 Results of different methods for 4× reconstruction for both synthetic and real-world scenes, showing predicted central SAIs, close-ups of the framed patches, and EPIs at the colored lines.

4.3.2 Effectiveness of MDISB

MDISB is used to guide reference view reconstruction. To validate the effectiveness of the MDISB, we removed this block, and again show the results in Table 3. rmMDISB suffered a 0.04 dB PSNR decrease compared to LF-CFANet. That is because this block can enahnce the influence of angular-position features in the process of reconstructing the reference-view image. Recall that in Eq. (8), we select four angular-position features with maximum-difference information according to the structure of the LF to boost the impact of maximum-difference views.

We also evaluated the performance of MDISB with different angular positions for auxiliary views; see Table 5 and Fig. 9. The reconstruction accuracy consistently improved as the degree of differentiation of the information increased. Table 5 shows that MDISB (maximum) had the best result: MDISB (0°), MDISB (45°), MDISB (90°), and MDISB (135°) only provide differentiated information in the same direction, while the difference information for MDISB (minimum) and MDISB (maximum) has four directions. Moreover, the four views in MDISB (maximum) are the furthest from the angular position of the reference view, and the maximum degree of differentiation can be provided for reconstructing the reference view.

Table 5 Different supplementary information is used in MDISB with the STFlytro dataset and 2× upscaling

Architecture	PSNR	SSIM
MDISB (0°)	38.83	**0.983**
MDISB (45°)	38.84	**0.983**
MDISB (90°)	38.81	**0.983**
MDISB (135°)	38.89	**0.983**
MDISB (minimum)	38.87	0.964
MDISB (maximum)	**38.91**	**0.983**

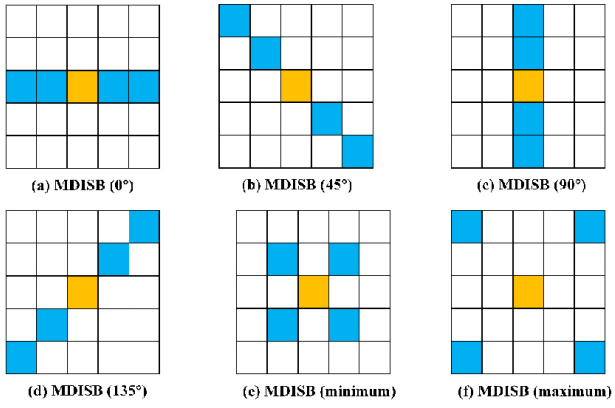

Fig. 9 Different supplementary information is used in MDISB. For a fair test, we fix the angular position of the reference view (yellow). Blue blocks represent different auxiliary views in different angular positions, which are used to supplement different information in MDISB.

4.3.3 Effectiveness of RCSCAM

The RCSCAM plays a key role in our LF-CFANet. This model can enhance the complementary information exploitation capability between the reference view and complementary view by introducing an attention mechanism. For comparison, we replaced our RCSCAM with simple feature concatenation. As shown in Table 5, this block had a significant influence on the result, and the PSNR suffered a 0.13 dB decrease. Without a spatial and channel attention mechanism, complementary information from cross-parallax images cannot be effectively learned to supplement the reference view.

5 Conclusions

In this paper, we propose the complementary-view feature attention network (LF-CFANet) for LFSR. The main contribution of our method is the fusion of complementary-view information, by using RCSCAM and MDISB. For RCSCAM, we use spatial and channel attention to effectively extract complementary-view feature information to supplement the reference view. To guide reference view reconstruction, MDISB is proposed to supplement the most differentiated feature-level information. Our experiments show that MDISB works well in the reconstruction process, allowing the reference view image to be effectively and efficiently reconstructed. Our method achieves state-of-the-art LFSR results in both quantitative and qualitative evaluations, and it is more robust for real-world scenes.

It is worth noting that the quality of the supplementary information from MDISB is crucial and improves the reconstruction accuracy. Therefore, a further study of the maximum-difference views is needed, and we could possibly use fewer views to reconstruct the whole set of LF views. In future work, we aim to use an encoder and decoder framework to improve the quality of feature fusion with fewer LF views, providing a further step toward consumer applications.

Acknowledgements

This study was partially supported by the National Key R&D Program of China (2018YFB2100500), the National Natural Science Foundation of China (61872025), the Science and Technology Development Fund, Macau SAR (0001/2018/AFJ), and the Open Fund of the State Key Laboratory of Software Development Environment (SKLSDE-2021ZX-03). We thank the HAWKEYE Group for their support.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Huang, F.-C.; Luebke, D.; Wetzstein, G. The light field stereoscope. In: Proceedings of the ACM SIGGRAPH Emerging Technologies, Article No. 24, 2015.

[2] Yu, J. A light-field journey to virtual reality. *IEEE MultiMedia* Vol. 24, No. 2, 104–112, 2017.

[3] Sheng, H.; Wang, S.; Zhang, Y.; Yu, D. X.; Cheng, X. Z.; Lyu, W. F.; Xiong, Z. Near-online tracking with co-occurrence constraints in blockchain-based edge computing. *IEEE Internet of Things Journal* Vol. 8, No. 4, 2193–2207, 2021.

[4] Sheng, H.; Zhang, Y.; Wu, Y. B.; Wang, S.; Lyu, W. F.; Ke, W.; Xiong, Z. Hypothesis testing based tracking with spatio-temporal joint interaction modeling. *IEEE Transactions on Circuits and Systems for Video Technology* Vol. 30, No. 9, 2971–2983, 2020.

[5] Wang, S.; Sheng, H.; Zhang, Y.; Wu, Y. B.; Xiong, Z. A general recurrent tracking framework without real data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 13199–13208, 2021.

[6] Wang, S.; Sheng, H.; Zhang, Y.; Yang, D.; Shen, J.; Chen, R. Blockchain-empowered distributed multi-camera multi-target tracking in edge computing. *IEEE Transactions on Industrial Informatics*, doi: 10.1109/TII.2023.3261890, 2023.

[7] Zhu, H.; Wang, Q.; Yu, J. Y. Occlusion-model guided antiocclusion depth estimation in light field. *IEEE Journal of Selected Topics in Signal Processing* Vol. 11, No. 7, 965–978, 2017.

[8] Piao, Y. R.; Li, X.; Zhang, M.; Yu, J. Y.; Lu, H. C. Saliency detection via depth-induced cellular automata on light field. *IEEE Transactions on Image Processing* Vol. 29, 1879–1889, 2020.

[9] Zhang, M.; Li, J.; Ji, W.; Piao, Y.; Lu, H. Memory-oriented decoder for light field salient object detection. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Article No. 81, 898–908, 2019.

[10] Bishop, T. E.; Favaro, P. The light field camera: Extended depth of field, aliasing, and superresolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 34, No. 5, 972–986, 2012.

[11] Wanner, S.; Goldluecke, B. Spatial and angular variational super-resolution of 4D light fields. In: *Computer Vision – ECCV 2012. Lecture Notes in Computer Science, Vol. 7576*. Fitzgibbon, A.; Lazebnik, S.; Perona, P.; Sato, Y.; Schmid, C. Eds. Springer Berlin, Heidelberg, 608–621, 2012.

[12] Wanner, S.; Goldluecke, B. Variational light field analysis for disparity estimation and super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 36, No. 3, 606–619, 2014.

[13] Mitra, K.; Veeraraghavan, A. Light field denoising, light field superresolution and stereo camera based refocussing using a GMM light field patch prior. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 22–28, 2012.

[14] Yuan, Y.; Cao, Z. Q.; Su, L. J. Light-field image superresolution using a combined deep CNN based on EPI. *IEEE Signal Processing Letters* Vol. 25, No. 9, 1359–1363, 2018.

[15] Yoon, Y.; Jeon, H. G.; Yoo, D.; Lee, J. Y.; Kweon, I. S. Learning a deep convolutional network for light-field image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision Workshop, 57–65, 2015.

[16] Yoon, Y.; Jeon, H. G.; Yoo, D.; Lee, J. Y.; Kweon, I. S. Light-field image super-resolution using convolutional neural network. *IEEE Signal Processing Letters* Vol. 24, No. 6, 848–852, 2017.

[17] Li, D. L.; Yang, D.; Wang, S. Z.; Sheng, H. Light field super-resolution based on spatial and angular attention. In: *Wireless Algorithms, Systems, and Applications. Lecture Notes in Computer Science, Vol. 12937*. Liu, Z.; Wu, F.; Das, S. K. Eds. Springer Cham, 314–325, 2021.

[18] Zhang, S.; Lin, Y. F.; Sheng, H. Residual networks for light field image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11038–11047, 2019.

[19] Yeung, H. W. F.; Hou, J. H.; Chen, X. M.; Chen, J.; Chen, Z. B.; Chung, Y. Y. Light field spatial super-resolution using deep efficient spatial-angular separable convolution. *IEEE Transactions on Image Processing* Vol. 28, No. 5, 2319–2330, 2019.

[20] Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7132–7141, 2018.

[21] Wang, L. G.; Wang, Y. Q.; Liang, Z. F.; Lin, Z. P.; Yang, J. G.; An, W.; Guo, Y. L. Learning parallax attention for stereo image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12242–12251, 2019.

[22] Wang, Y. Q.; Ying, X. Y.; Wang, L. G.; Yang, J. G.; An, W.; Guo, Y. L. Symmetric parallax attention for stereo image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 766–775, 2021.

[23] Yang, W. M.; Zhang, X. C.; Tian, Y. P.; Wang, W.; Xue, J. H.; Liao, Q. M. Deep learning for single image super-resolution: A brief review. *IEEE Transactions on Multimedia* Vol. 21, No. 12, 3106–3121, 2019.

[24] Wang, Z. H.; Chen, J.; Hoi, S. C. H. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 43, No. 10, 3365–3387, 2021.

[25] Dong, C.; Loy, C. C.; He, K. M.; Tang, X. O. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 38, No. 2, 295–307, 2015.

[26] Dong, C.; Loy, C. C.; He, K. M.; Tang, X. O. Learning a deep convolutional network for image super-resolution. In: *Computer Vision – ECCV 2014. Lecture Notes in Computer Science, Vol. 8692*. Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T. Eds. Springer Cham, 184–199, 2014.

[27] Kim, J.; Lee, J. K.; Lee, K. M. Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1646–1654, 2016.

[28] Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K. M. Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 1132–1140, 2017.

[29] Timofte, R.; Agustsson, E.; Van Gool, L.; Yang, M.-H.; Zhang, L. NTIRE 2017 challenge on single image super-resolution: Methods and results. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 1110–1121, 2017.

[30] Zhang, Y. L.; Tian, Y. P.; Kong, Y.; Zhong, B. N.; Fu, Y. Residual dense network for image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2472–2481, 2018.

[31] Zhang, Y. L.; Tian, Y. P.; Kong, Y.; Zhong, B. N.; Fu, Y. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 43, No. 7, 2480–2495, 2021.

[32] Zhang, Y. L.; Li, K. P.; Li, K.; Wang, L. C.; Zhong, B. N.; Fu, Y. Image super-resolution using very deep residual channel attention networks. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11211*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 294–310, 2018.

[33] Dai, T.; Cai, J. R.; Zhang, Y. B.; Xia, S. T.; Zhang, L. Second-order attention network for single image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11057–11066, 2019.

[34] Zhang, F. L.; Wang, J.; Shechtman, E.; Zhou, Z. Y.; Shi, J. X.; Hu, S. M. PlenoPatch: Patch-based plenoptic image manipulation. *IEEE Transactions on Visualization and Computer Graphics* Vol. 23, No. 5, 1561–1573, 2017.

[35] Rossi, M.; Frossard, P. Geometry-consistent light field super-resolution via graph-based regularization. *IEEE Transactions on Image Processing* Vol. 27, No. 9, 4207–4218, 2018.

[36] Alain, M.; Smolic, A. Light field super-resolution via LFBM5D sparse coding. In: Proceedings of the 25th IEEE International Conference on Image Processing, 2501–2505, 2018.

[37] Huang, Y.; Wang, W.; Wang, L. Bidirectional recurrent convolutional networks for multi-frame super-resolution. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, 235–243, 2015.

[38] Wang, Y. L.; Liu, F.; Zhang, K. B.; Hou, G. Q.; Sun, Z. N.; Tan, T. N. LFNet: A novel bidirectional recurrent convolutional neural network for light-field image super-resolution. *IEEE Transactions on Image Processing* Vol. 27, No. 9, 4274–4286, 2018.

[39] Jin, J.; Hou, J. H.; Chen, J.; Kwong, S. Light field spatial super-resolution via deep combinatorial geometry embedding and structural consistency regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2257–2266, 2020.

[40] Wang, Y. Q.; Wang, L. G.; Yang, J. G.; An, W.; Yu, J. Y.; Guo, Y. L. Spatial-angular interaction for light field image super-resolution. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12368*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 290–308, 2020.

[41] Mo, Y.; Wang, Y.; Xiao, C.; Yang, J.; An, W. Dense dual-attention network for light field image super-resolution. *IEEE Transactions on Circuits and Systems for Video Technology* Vol. 32, No. 7, 4431–4443, 2022.

[42] Rerabek, M.; Ebrahimi, T. New light field image dataset. In: Proceedings of the 8th International Conference on Quality of Multimedia Experience, 2016.

[43] Honauer, K.; Johannsen, O.; Kondermann, D.; Goldluecke, B. A dataset and evaluation methodology for depth estimation on 4D light fields. In: *Computer Vision – ACCV 2016. Lecture Notes in Computer Science, Vol. 10113*. Lai, S. H.; Lepetit, V.; Nishino, K.; Sato, Y. Eds. Springer Cham, 19–34, 2017.

[44] Wanner, S.; Meister, S.; Goldluecke, B. Datasets and benchmarks for densely sampled 4D light fields. In: Proceedings of the Vision, Modeling & Visualization, 225–226, 2013.

[45] Le Pendu, M.; Jiang, X. R.; Guillemot, C. Light field inpainting propagation via low rank matrix completion. *IEEE Transactions on Image Processing* Vol. 27, No. 4, 1981–1993, 2018.

[46] Vaish, V.; Adams, A. The (new) Stanford light field archive. Computer Graphics Laboratory, Stanford University, 2008. Available at http://lightfield.stanford.edu/index.html.

[47] Raj, A. S.; Lowney, M.; Shah, R.; Wetzstein, G. Stanford Lytro light field archive. 2016. Available at http://lightfields.stanford.edu/LF2016.html.

[48] Chen, L. C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. L. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 40, No. 4, 834–848, 2018.

[49] Ying, X. Y.; Wang, Y. Q.; Wang, L. G.; Sheng, W. D.; An, W.; Guo, Y. L. A stereo attention module for stereo image super-resolution. *IEEE Signal Processing Letters* Vol. 27, 496–500, 2020.

[50] Kim, J.; Lee, J. K.; Lee, K. M. Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1646–1654, 2016.

Wei Zhang received his M.S. degree in control engineering from Beijing University of Chemical Technology in 2019. He is currently a Ph.D. student in the Faculty of Applied Sciences, Macao Polytechnic University, China. His research interests include image processing and computer vision.

Wei Ke received his Ph.D. degree from the School of Computer Science and Engineering, Beihang University, China. He is an associate professor of computing, Macao Polytechnic University. His research interests include programming languages, image processing, computer graphics, and component-based engineering and systems. His recent research focuses on the design and implementation of open platforms for applications of computer graphics and pattern recognition.

Da Yang received his B.S. degree from the School of Computer Science and Engineering, Beihang University in 2012. He is currently pursuing a Ph.D. degree with the School of Computer Science and Engineering, Beihang University.

Hao Sheng received his B.S. and Ph.D. degrees from the School of Computer Science and Engineering of Beihang University in 2003 and 2009, respectively. Now he is a professor and Ph.D. supervisor in the School of Computer Science and Engineering, Beihang University. He is working on computer vision, pattern recognition, and machine learning.

Zhang Xiong received his B.S degree from Harbin Engineering University in 1982, and his M.S. degree from Beihang University in 1985. He is a professor and Ph.D. supervisor in the School of Computer Science and Engineering, Beihang University. He is working on computer vision, information security, and data visualization.