LRT: An Efficient Low-Light Restoration Transformer for Dark Light Field Images

Shansi Zhang, Nan Meng and Edmund Y. Lam

***Abstract*—Light field (LF) images with the multi-view property have many applications, which can be severely affected by the low-light imaging. Recent learning-based methods for low-light enhancement have their own disadvantages, such as no noise suppression, complex training process and poor performance in extremely low-light conditions. Targeted on solving these deficiencies while fully utilizing the multi-view information, we propose an efficient Low-light Restoration Transformer (LRT) for LF images, with multiple heads to perform specific intermediate tasks, including denoising, luminance adjustment, refinement and detail enhancement, within a single network, achieving progressive restoration from small scale to full scale. We design an angular transformer block with a view-token scheme to model the global angular relationship efficiently, and a multi-scale window-based transformer block to encode the multi-scale local and global spatial information. To solve the problem of insufficient training data, we formulate a synthesis pipeline by simulating the major noise with the estimated noise parameters of LF camera. Experimental results demonstrate that our method can achieve superior performance on the restoration of extremely low-light and noisy LF images with high efficiency.**

***Index Terms*—Dark light field, angular transformer, multi-scale window-based transformer, noise parameters.**

I. Introduction

Light field (LF) cameras can record both the intensities and directions of the light rays, which enables multi-view imaging and brings many applications, such as post-capture refocusing [1], [2], depth estimation [3], [4], de-occlusion [5], [6] and salient detection [7], [8]. However, these applications are susceptible to the degraded LF images caused by low-light imaging, which leads to invisible contents and serious noise. Simply increasing ISO and exposure time are not helpful since they may promote the noise level and introduce blueness, respectively. Therefore, low-light enhancement algorithms are expected to recover visibility and suppress noise for better LF applications.

Most existing methods are designed for low-light single images, while there is not a lot of work dealing with LF images. LF images have the unique property of capturing multiple views of the scene to provide rich geometric cues. The complementary information across different views should contribute to better LF restoration. Thus, some earlier work [9], [10] utilizes multiple surrounding views to restore the central view. Nevertheless, efficiency is compromised to some extent, as they can only restore one view at each forward process. Meanwhile, others [11], [12] apply convolutions on the macro-pixels to extract angular features and restore all the views synchronously. However, small kernel size only incorporates partial angular information at each step while large kernel size obviously increases parameters.

To address the above deficiencies, we propose an efficient Low-light Restoration Transformer (LRT) for LF images, which incorporates global angular self-attention to fully utilize the information of all the views for restoring the individual views, and multi-scale local and global self-attention within each view to encode rich spatial information. Our LRT contains multiple heads to perform specific intermediate tasks by completely taking into account denoising, luminance enhancement, and detail preservation within a single network. Moreover, paired low-light/normal-light LF images are required to train a network. However, there is no sufficient such dataset available, and it is difficult and costly to collect a large dataset with aligned low-light LFs and ground truths. To solve this problem, we create a synthetic dataset by modeling the sensor noise and estimating the noise parameters of LF camera to synthesize more realistic low-light LF images.

Our main contributions are as follows:

- We develop a transformer-based network with multiple heads to progressively perform denoising, luminance adjustment, refinement and detail enhancement from small scale to full scale, which separates the intermediate tasks in different branches for better restoration while guaranteeing high inference efficiency. An adaptive ratio adjustment module is introduced to adjust the light level of input LF for better prediction of high-frequency details. All the tasks can be learned simultaneously through an end-to-end training.
- We propose an angular transformer block to selectively fuse the information of all the views for restoring each individual view. It adopts an efficient view-token scheme to learn the long-range angular dependencies.
- We propose a multi-scale window-based transformer block to encode multi-scale local and global spatial features. Window partition and spatial reduction are employed to reduce the computational cost significantly.
- We evaluate our LRT on the real LF images under extremely low-light conditions and with serious noise. The experimental results demonstrate that our method can outperform the other state-of-the-art low-light enhancement methods for single images and LF images.

II. Related work

A. Low-light image enhancement

Earlier model-based methods [13]–[16] for low-light image enhancement usually apply the Retinex theory [17] to decompose a low-light image into its illumination and reflectance,

S. Zhang and E.Y. Lam are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong e-mail: sszhang@eee.hku.hk, elam@eee.hku.hk.

N. Meng is with the Li Ka Shing Faculty of Medicine, The University of Hong Kong, Pokfulam, Hong Kong.

and then enlighten the illumination. However, these methods rely on the carefully designed priors and constraints, and their inference speeds are relatively low.

With the prevalence of deep learning, more and more methods improved the performance and efficiency by training a deep convolutional neural network (CNN). Lv et al. [18] proposed a multi-branch network that fuses the output of multiple subnets to obtain the enhanced images or videos. Jiang et al. [19] developed EnlightenGAN, with a UNet-based generator and a global-local discriminator. Wang et al. [20] proposed a lightening network that learns the residual between the low-light and normal-light images with iterative lightening and darkening processes. These methods learn a direct mapping from the low-light image to the normal-light image. Even if their training approach is simple, the learning of mapping incorporating both luminance enhancement and denoising is relatively difficult. Hence, their performances are limited under extremely low-light conditions and severe noise levels.

Instead of learning a direct image-to-image mapping, Wang et al. [21] designed a network with intermediate illumination estimation. The reciprocal of the estimated illumination map multiplied with the low-light input yielded the enhanced image. Guo et al. [22] proposed a zero-reference curve estimation network which estimates high-order curves for each pixel to adjust the illumination of input image. Ma et al. [23] developed a self-calibrated illumination learning framework with cascaded illumination estimation and refinement to achieve fast low-light enhancement. The above methods can only enhance the illumination of dark images without considering the noise, which limits their applications to the real low-light scenarios with non-negligible noise.

There are some Retinex-based methods by training separate networks for decomposition and enhancement. Zhang et al. [24] developed a framework, which contains a decomposition network, a reflectance restoration network and an illumination adjustment network to deal with the noise and low luminance, respectively. They further improved the restoration network by introducing illumination attention in [25]. Wu et al. [26] proposed a deep unfolding framework, with an initialization module for decomposition, an unfolding optimization module to refine the illumination and reflectance iteratively, and an illumination adjustment module to enhance the illumination. The training processes of these methods are complex and their inference speeds are obviously lower than an end-to-end network. Also inspired by the Retinex theory, Liu et al. [27] proposed an unrolling framework with an illumination estimation module and a noise removal module, which unroll the optimization processes with lightweight learnable networks. However, this unrolling approach for denoising is not effective enough when the noise level is high.

B. Low-light LF enhancement

Low-light enhancement for LF images needs to deal with not only the low illumination and noise but also the integration of multi-view information. There is only a few work targeting on it. Lamba [10] et al. proposed a network, which contains a global representation block to encode angular geometry and a view reconstruction block to restore each view by utilizing multiple neighboring views. They further proposed a three-stage network [28], with global embedding, view discrimination and RNN-inspired view restoration, to boost the performance. They adopted the direct image-to-image mapping approach, which could cause learning difficulty for extremely degraded LFs. Ge et al. [29] employed 4D convolution [30] with simultaneous spatial and angular feature extraction to construct a network for dark LF restoration. However, 4D convolution involves large computational cost, resulting in low efficiency. Zhang and Lam [31] proposed a two-stage framework, which contains a multi-to-one network to restore the individual views by fusing the information from other views, and an all-to-all network to refine all the views synchronously with alternate spatial-angular feature extraction. However, this framework was trained only on the gray images, and the two-stage architecture is not efficient. To target on the RGB images, they further proposed a Retinex-based framework [32], which separates the noise suppression and illumination enhancement, and incorporates interaction and fusion of the spatial and angular information. However, its training process is tedious due to the multiple separate networks, and the inference efficiency is still relatively low. Moreover, both the two frameworks were trained by the synthetic data with random noise, which have obvious deviation with the real noise distribution of LF imaging, resulting in degraded performance when generalizing to the real low-light scenes.

III. Low-light restoration transformer for LF images

A. Overall architecture

The overall architecture of LRT is depicted in Fig. 1, where the left part is summarized as local and global feature extraction, and the right part describes multiple heads to achieve the intermediate tasks. Our LRT restores the dark LF images from small scale to full scale progressively. Spatial residual blocks (ResBlocks) are used to extract the local features within each view, and angular transformer blocks are designed to model the dependencies among all the views. The input degraded LF L_{in} is first fed to the spatial ResBlocks and angular transformer blocks alternately to encode the spatial-angular information. $\frac{1}{2}$-scale, $\frac{1}{4}$-scale and $\frac{1}{8}$-scale features are obtained during the contracting path after progressive downsampling by the patch merging layers (convolution with stride 2). The $\frac{1}{8}$-scale features further pass through several multi-scale window-based transformer blocks to model the long-range dependencies within each view. During the expanding path, the features are gradually upsampled through the patch expanding layers [33] and fuse with the corresponding features in the contracting path with skip connections. The fused features at multiple scales are further processed by the spatial ResBlocks and angular transformer blocks, followed by different heads, each of which deals with a specific intermediate task.

According to the Retinex theory, an image can be expressed as the multiplication of its reflectance and illumination. Thus, the low-light image divided by its illumination yields the

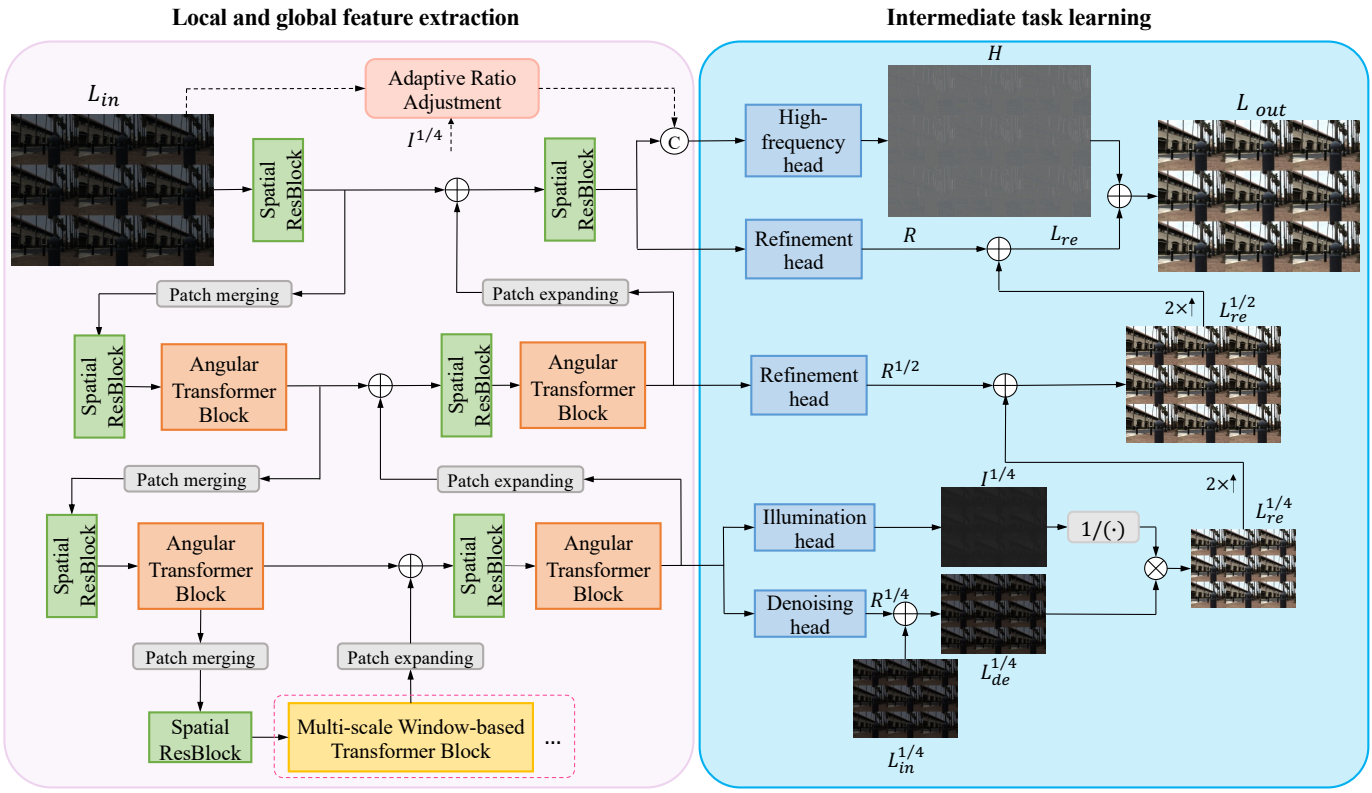


Fig. 1. Our LRT contains spatial ResBlocks and multi-scale window-based transformer blocks to extract the local and global features within each view, and angular transformer blocks to explore the dependencies among all the views. It has multiple heads to achieve different intermediate tasks, with denoising and illumination estimation in the $\frac{1}{4}$-scale branch, refinement and detail enhancement in the $\frac{1}{2}$-scale and full-scale branches.

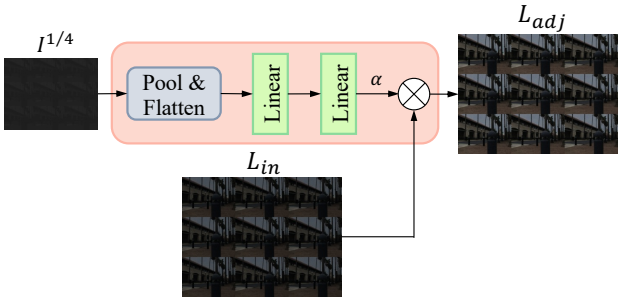


Fig. 2. Adaptive ratio adjustment module. It outputs a ratio α by taking the estimated illumination map as input to adaptively adjust the light level of input LF for better detail prediction.

normal-light reflectance, which usually suffers from noise. Given this, we introduce an illumination head and a denoising head explicitly performing illumination estimation and noise removal to obtain the normal-light clean reflectance. The illumination head consisting of a convolution layer with sigmoid activation estimates the illumination of input LF image in the $\frac{1}{4}$-scale branch. The denoising head consisting of a convolution layer with tanh activation is to remove noise from the $\frac{1}{4}$-scale input LF $L_{\text{in}}^{1/4}$, which is obtained by downsampling L_{in}. The denoised LF $L_{\text{de}}^{1/4}$ is divided by the estimated illumination $I^{1/4}$ in an element-wise manner to yield the $\frac{1}{4}$-scale restored LF $L_{\text{re}}^{1/4}$. This configuration separates the denoising and luminance enhancement to better handle each one, and significantly reduces the color distortion of output LF, compared with the direct mapping from the low-light noisy image to the normal-light clean image. These two heads are performed at a small scale, which can decrease the computational cost while leaving space for refinement in larger scales.

The $\frac{1}{2}$-scale branch contains a refinement head that comprises of a convolution layer with tanh activation to output the $\frac{1}{2}$-scale residual map $R^{1/2}$. $L_{\text{re}}^{1/4}$ is upsampled and added to $R^{1/2}$ to yield the $\frac{1}{2}$-scale restored LF $L_{\text{re}}^{1/2}$. The full-scale branch also has a refinement head to output the residual map R. Similarly, $L_{\text{re}}^{1/2}$ is further upsampled and added to R to obtain the full-scale restored LF L_{re}. Moreover, it has a high-frequency head to predict the high-frequency components for enhancing the local details. Considering the extremely low-light inputs with nearly invisible high-frequency details, we introduce a adaptive ratio adjustment module to adjust the light level of input LF adaptively. It (Fig. 2) takes the estimated illumination map $I^{1/4}$ as input to serve as a luminance clue. $I^{1/4}$ is first pooled and flattened for integrating the key information, and then fed to two linear layers to output a factor α, which is multiplied with L_{in} to obtain the adjusted LF L_{adj}. The concatenation of L_{adj} and full-scale features passes through the high-frequency head to output the high-frequency map H. This adaptive learning approach aims to find a proper ratio to make L_{adj} provide clearer cues for predicting the local details while avoiding over-amplifying the noise that interfere

the identification of object details. H plus L_{re} yields the final output LF image L_{out}. The overall flow is expressed as

$$\begin{aligned} L_{\text{de}}^{1/4} &= R^{1/4} + L_{\text{in}}^{1/4}, \\ L_{\text{re}}^{1/4} &= L_{\text{de}}^{1/4} \oslash I^{1/4}, \\ L_{\text{re}}^{1/2} &= \text{UP}(L_{\text{re}}^{1/4}) + R^{1/2}, \\ L_{\text{re}} &= \text{UP}(L_{\text{re}}^{1/2}) + R, \\ L_{\text{out}} &= L_{\text{re}} + H, \end{aligned} \tag{1}$$

where $R^{1/4}$ is the output of denoising head, $\text{UP}(\cdot)$ denotes the $2\times$ upsampling operation, and $\oslash$ denotes the element-wise division.

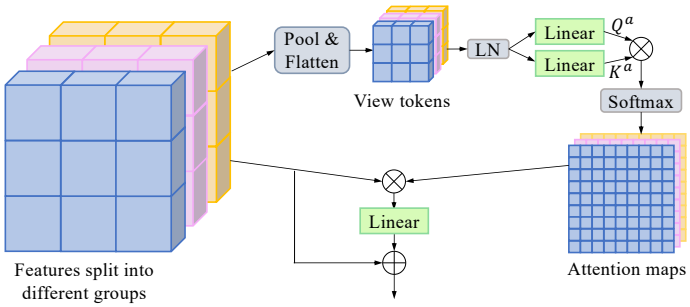


Fig. 3. Angular transformer block. View tokens are obtained by applying pooling and flatten operations to the view features, and they are further processed by linear layers to generate the query and key. The view features are treated as the value. The query, key and value are split into different groups for self-attention computation.

B. Angular transformer block

The angular transformer block aims to explore the long-range angular dependencies among different views. Fig. 3 shows its structure (taking 3×3 views as an example). The feature $F \in \mathbb{R}^{u\times v\times c\times h\times w}$ (u and v for the angular dimension, h and w for the spatial dimension, and c for the channel dimension,) are first split into m groups along the channel dimension, with $\{F_i\}_{i=1}^{m}$ ($F_i \in \mathbb{R}^{u\times v\times \frac{c}{m}\times h\times w}$). Huge computational cost could be caused if all the feature elements of each view are exploited to determine the attention. Thus, for each group, we apply the pooling and flatten operations to all the view features to obtain the view tokens, each of which incorporates the core information of one view. Then, the view tokens pass through a layer normalization (LN) and two linear layers to generate the query $Q_i^a \in \mathbb{R}^{uv\times d}$ and key $K_i^a \in \mathbb{R}^{uv\times d}$, with dimension d. F_i is treated as the value. The procedures of calculating the angular self-attention for group i is as follows

$$\begin{aligned} Q_i^a &= \text{LN}(\text{Pool}(F_i))W_i^Q, \\ K_i^a &= \text{LN}(\text{Pool}(F_i))W_i^K, \\ G_i^a &= \text{softmax}\Big(\frac{Q_i^a (K_i^a)^T}{\sqrt{d}}\Big)F_i, \end{aligned} \tag{2}$$

where $\text{Pool}(\cdot)$ means pooling with flatten operated on each view feature, $\text{LN}(\cdot)$ denotes the layer normalization, W_i^Q and W_i^K are the parameters of the linear layers.

The outputs of all the groups are concatenated together, and then fed to another linear layer to fuse the features of different groups, whose output is added to the input feature to obtain the output feature F' with cross-view information. This process is expressed as

$$F' = [G_1^a, G_2^a, \cdots, G_m^a]W^G + F, \tag{3}$$

where $[\cdot]$ denotes the concatenation operation and W_G denotes the parameters of the linear layer for fusion.

Given the feature with size $u\times v\times c\times h\times w$, the complexity of our angular transformer block is calculated as

$$\Omega(\text{Angular}) = uvc^2\frac{p^2+2}{m} + (uv)^2hwc + (uv)^2c + uvhwc^2, \tag{4}$$

where p is the size of view feature after pooling. Compared with [34], which computes angular self-attention within each macro-pixel, with the complexity $4uvhwc^2 + 2(uv)^2hwc$, our angular transformer block involves less computational cost due to the efficient view-token scheme.

In this way, each view can integrate the complementary information from all the other views according to the attention maps, which indicate the potential contributions of all the views for restoring each individual view. Moreover, it enables synchronous restoration of all the views in each forward process.

C. Multi-scale window-based transformer block

The original transformer architecture performs global self-attention by computing the relationship between each token and all the other tokens, which results in high computational complexity, especially for dense prediction tasks with high-resolution feature maps. Self-attention within local windows is a promising solution for reducing the computation burden. However, it can not model the global dependencies and the relationships across different windows effectively.

For efficient modeling on both the global and local dependencies within each view, we propose a spatial transformer block (Fig. 4) with the multi-scale window-based self-attention, which is divided into four groups to encode features in different scales. The first group (global group) computes self-attention within the overall feature maps, and the other three groups (local groups) compute self-attention within the local windows with different sizes. The $\frac{1}{8}$-scale feature after LN passes through a linear layer to obtain the query Q_1^s for the global group, and Q_2^s, Q_3^s and Q_4^s for the local groups, after being partitioned into 2×2, 4×4 and 8×8 windows, respectively. To further reduce the computational cost, we decrease the lengths of keys and values [35] by using convolution layers with large strides for downsampling the features. Then, LN, GELU non-linearity [36] and linear layers are employed to generate $\{K_j^s\}_{j=1}^4$ and $\{V_j^s\}_{j=1}^4$. For the local groups, stride convolutions are conducted within their respective local windows. The spatial self-attention of each group is calculated by

$$G_j^s = \text{softmax}\Big(\frac{Q_j^s (K_j^s)^T}{\sqrt{d}}\Big)V_j^s. \tag{5}$$

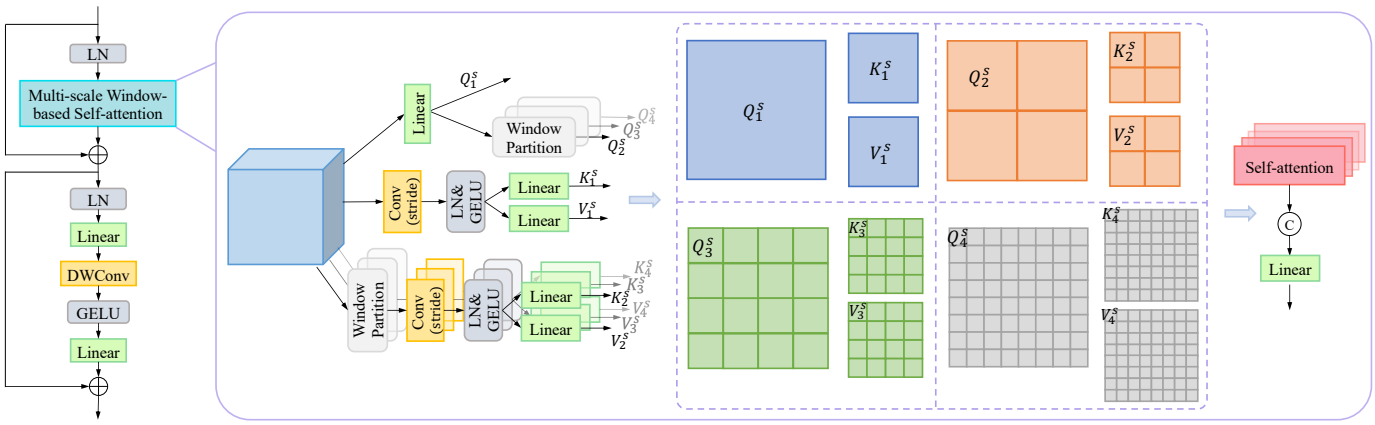


Fig. 4. Spatial transformer block. It adopts multi-scale window-based self-attention, with four groups to encode both the global and local features. The first group computes the global self-attention, and the other three groups compute self-attention within the local windows with different sizes. Stride convolutions are used to reduce the lengths of keys and values for more efficient computation.

The outputs of all the groups are concatenated and then fed to a linear layer for fusion.

Given the spatial feature with size $c \times h \times w$, the kernel size of stride convolution t and the number of windows $n \times n$ for one group, the complexity of our spatial self-attention, with four groups $\mathcal{S} = \{(1,4),(2,4),(4,2),(8,2)\}$, is calculated as

$$\begin{aligned}\Omega(\text{Spatial}) = &\sum_{(n,t)\in\mathcal{S}} \Big(\frac{hwc^2}{2n^2} + \frac{hwc^2}{4(nt)^2} + \frac{(hw)^2c}{2(nt)^2n^2}\Big)n^2 \\ &+ 2hwc^2 = 4hwc^2 + \frac{5}{32}hwc^2 + \frac{25}{512}(hw)^2c,\end{aligned} \tag{6}$$

which is much less than the complexity of common global self-attention, with $4hwc^2 + 2(hw)^2c$.

After the self-attention layer, the features pass through the feed-forward layers. Similar with PVTv2 [37], we add a depth-wise convolution layer between the two linear layers to complement the local information.

D. Loss function

Our LRT contains multiple heads to implement different intermediate tasks, each of which corresponds to specific loss terms. For the $\frac{1}{4}$-scale denoising head, the denoising loss is

$$\ell_{\text{de}} = \|L_{\text{de}}^{1/4} - L_{\text{low}}^{1/4}\|_1, \tag{7}$$

where $L_{\text{low}}^{1/4}$ is the $\frac{1}{4}$-scale clean low-light LF with the same illumination as the input LF, and $\|\cdot\|_1$ is the ℓ_1 norm.

For the $\frac{1}{4}$-scale illumination head, the estimated illumination map should be smooth while preserving the object structures, which introduces the smoothness loss as

$$\ell_{\text{sm}} = |\nabla I^{1/4}| \times e^{(-\eta|\nabla L_{\text{gt}}^{1/4}|)}, \tag{8}$$

where ∇ means computing the gradients along both the horizontal and vertical directions, η is a hyper-parameter to adjust the structure awareness, and $L_{\text{gt}}^{1/4}$ is the $\frac{1}{4}$-scale ground-truth LF to serve as a structure reference.

Moreover, we propose an illumination reference loss for better structure and contrast preservation by utilizing the Y channel (YUV color space) of low-light LF, defined as

$$\ell_{\text{ref}} = \|\text{Nor}(I^{1/4}) - \text{Nor}(Y^{1/4})\|_1, \tag{9}$$

where $Y^{1/4}$ is the Y channel of $L_{\text{low}}^{1/4}$, $\text{Nor}(\cdot)$ is to normalize the illumination map and Y channel to $0 \sim 1$, with $\text{Nor}(I^{1/4}) = \frac{I^{1/4} - \min(I^{1/4})}{\max(I^{1/4}) - \min(I^{1/4})}$. By introducing this constraint, the illumination head can learn the light distribution among different objects more effectively with the guidance of Y channel.

The high-frequency map is learned by the supervision of its ground truth F_{gt}, which is obtained by applying Gaussian blur to L_{gt}, with

$$\begin{aligned} F_{\text{gt}} &= L_{\text{gt}} - \text{Gau}(L_{\text{gt}}), \\ \ell_{\text{hf}} &= \|F - F_{\text{gt}}\|_1, \end{aligned} \tag{10}$$

where $\text{Gau}(\cdot)$ denotes the Gaussian filter.

Reconstruction loss is applied to the restored LFs with all the scales and the output LF, expressed as

$$\ell_{\text{rec}} = \sum_{z=1,\frac{1}{2},\frac{1}{4}} \|L_{\text{re}}^z - L_{\text{gt}}^z\|_1 + \|L_{\text{out}} - L_{\text{gt}}\|_1. \tag{11}$$

In addition, we impose the structural similarity (SSIM) [38] loss to the final output LF to improve the visual quality, with

$$\ell_{\text{SSIM}} = 1 - \text{SSIM}(L_{\text{out}}, L_{\text{gt}}). \tag{12}$$

Thus, the full loss for training LRT is the combination of the above loss terms with different coefficients manually tuned by us, written as

$$\ell_{\text{full}} = 10\ell_{\text{de}} + 0.1\ell_{\text{sm}} + \ell_{\text{ref}} + \ell_{\text{hf}} + 5\ell_{\text{rec}} + \ell_{\text{SSIM}}. \tag{13}$$

IV. Noise Parameter Estimation for LF Camera

As with the conventional camera, the imaging process of LF camera also introduces various noise sources.

The number of photons hitting a pixel follows a Poisson distribution, which is the source of shot noise. Some of the

incident photons are converted to the electrons, and the number of photoelectrons E_{photon} is proportional to the exposure time τ, luminous flux Φ, and quantum efficiency α, with

$$E_{\text{photon}} \sim \mathcal{P}(\tau\alpha\Phi), \tag{14}$$

where $\mathcal{P}$ denotes the Poisson distribution.

Few electrons are generated when there is no incident photon. The number of electrons E_{dark} follows a Poisson distribution, which introduces the dark noise, with

$$E_{\text{dark}} \sim \mathcal{P}(\tau D), \tag{15}$$

where D is the number of electrons produced per unit time under current temperature.

Therefore, the total number of electrons E_{total} before the amplifier also follows a Poisson distribution, with

$$\begin{aligned} E_{\text{total}} &= E_{\text{photon}} + E_{\text{dark}}, \\ E_{\text{total}} &\sim \mathcal{P}(\tau\alpha\Phi + \tau D). \end{aligned} \tag{16}$$

The collected electrons are converted to the voltage, which passes though the analog amplifier and analog-to-digital converter (ADC) to obtain the pixel value. These stages introduce the read noise N_{read} following a Gaussian distribution $\mathcal{N}(0, \sigma_{\text{read}})$, and the quantization noise N_{quan} following a uniform distribution $\mathcal{U}(-\frac{q}{2}, \frac{q}{2})$ with the quantization step q. Furthermore, banding noise usually exits under low-light imaging. Here, we mainly consider the row noise N_{row}, which is modeled as an offset added to each row and sampled from a Gaussian distribution $\mathcal{N}(0, \sigma_{\text{row}})$. Thus, the raw data from a LF camera can be expressed as

$$L = kE_{\text{total}} + N_{\text{read}} + N_{\text{quan}} + N_{\text{row}}, \tag{17}$$

where k is the system gain relative to ISO.

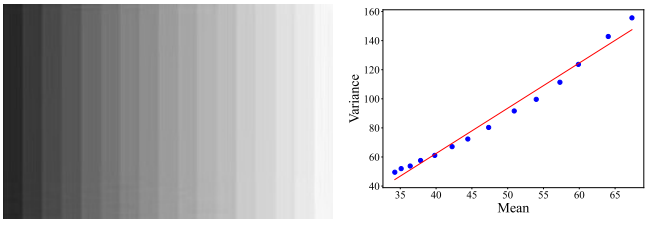
Let $N_{\text{add}} = N_{\text{read}} + N_{\text{quan}} + N_{\text{row}}$. Then, the mean and variance of the raw data are expressed as

$$\mathbb{E}(L) = k(\tau\alpha\Phi + \tau D) \tag{18}$$

$$\text{Var}(L) = k^2(\tau\alpha\Phi + \tau D) + \text{Var}(N_{\text{add}}). \tag{19}$$

Thus, we have

$$\text{Var}(L) = k\mathbb{E}(L) + \text{Var}(N_{\text{add}}). \tag{20}$$



(a) (b)

Fig. 5. (a) Gray-scale image. (b) Estimate k under a specific ISO with linear regression.

To estimate k under a specific ISO, we capture a series of gray-scale images (Fig. 5(a)) by the LF camera. The raw LF images are first rectified by the estimated centroids of micro images to obtain the aligned images, where the mean and variance within a small region of each gray scale are calculated. Then, we apply linear regression to find an optimal line fitting the points of mean and variance (Fig. 5(b)), and the slope of the line is the estimated k.

To estimate the signal-independent noise, we capture a series of dark LF images under each ISO setting. From Eq. 18, we have $\mathbb{E}(L) = k\tau D$ when $\Phi = 0$. Thus, we can calculate the mean of these dark images under a specific ISO and exposure time, and the mean divided by k can be the estimated dark noise. For the row noise, the mean values of each row can be treated as the row noise intensities as the mean of other noise sources approximates to 0 [39]. Then, the standard deviation σ_{row} can be estimated by fitting a normal distribution. After removing the dark noise and row noise from the dark image, we can estimate the standard deviation of read noise σ_{read} by fitting another normal distribution.

V. Experiments

A. Synthesis of dark LF images

With the method introduced in Section IV, we can estimate k, σ_{read} and σ_{row} under different ISO settings for the Lytro Illum LF camera. To synthesize dark noisy LF images, clean images from Kalantari [40], Stanford [41] and EPFL [42] datasets are chosen as the ground truths. Then, we can introduce noise sources to L_{low} with the estimated noise parameters to obtain the low-light noisy LF L_{in}. The synthesis pipeline is formulated as

$$\begin{aligned} L_{\text{low}} &= \gamma L_{\text{gt}}, \\ L_{\text{in}} &= k\mathcal{P}(\frac{L_{\text{low}}}{k} + \hat{E}_{\text{dark}}) + \mathcal{N}(0, \hat{\sigma}_{\text{read}}) \\ &\quad + \mathcal{N}(0, \hat{\sigma}_{\text{row}}) + \mathcal{U}(-\frac{q}{2}, \frac{q}{2}). \end{aligned} \tag{21}$$

The synthesis procedures are implemented on the plenoptic image, as shown in Fig. 6, and the noise and low luminance are allocated to different views after converting the plenoptic image to the sub-aperture image (SAI) array. We only preserve the central 7×7 views, which have very similar illumination and noise levels. Fig. 7 presents the central views of our synthetic LF images with different k values and captured LF images under different ISOs, which are brightened to clearly show the noise distributions. We use these synthetic images instead of real-captured images to train our model.

B. Implementation details

During each training step, the input LF images were synthesized from the ground truths with the randomly selected low-light factors and noise parameters, and cropped to 256×256 patches randomly. The $\frac{1}{2}$-scale and $\frac{1}{4}$-scale images are obtained by downsampling the full-scale images. Gaussian smoothing is applied before each downsampling operation to avoid the aliasing effect [43]. Our LRT was trained end-to-end using the loss in Eq. 13 with Adam optimizer for about 300 epochs. The learning rate was initially set to 5×10^{-4}, and decayed by multiplying 0.8 after every 50 epochs. It was implemented with PyTorch on NVIDIA Tesla P100 GPU.

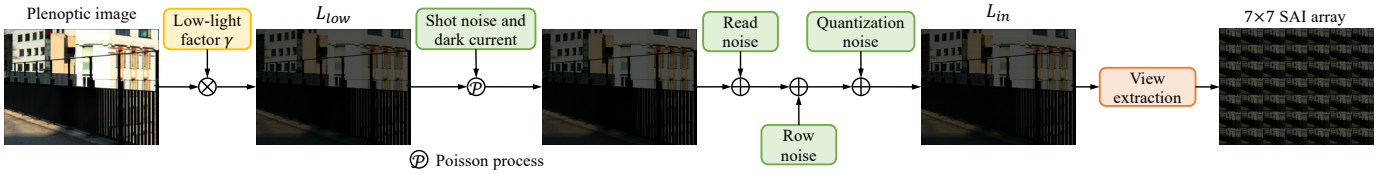


Fig. 6. The plenoptic image is first multiplied by a low-light factor, and then different noise sources are added to obtain the low-light noisy plenoptic image, which is converted to the SAI array with 7×7 views.

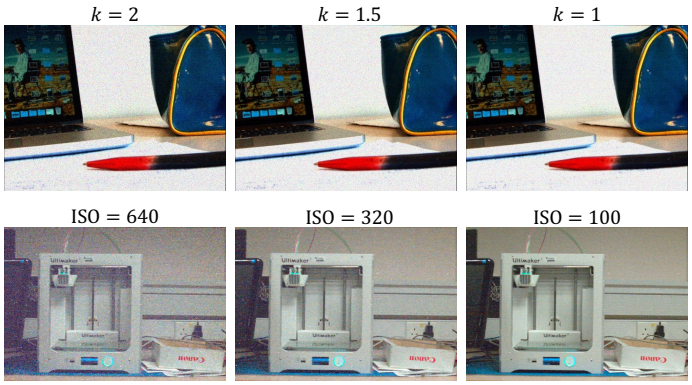


Fig. 7. The first row presents the central views of synthetic LF images with different k values. The second row shows the central views of captured LF images by Lytro Illum under different ISOs.

TABLE I
QUANTITATIVE RESULTS ON THE L3F DATASET. BOLD: BEST.

Method	PSNR↑	SSIM↑	LPIPS↓
Single image method			
DeepUPE [21]	19.093	0.614	0.235
Zero-DCE [22]	18.485	0.521	0.339
RUAS [27]	20.475	0.594	0.259
KinD++ [25]	18.708	0.592	0.270
URetinex-Net [26]	20.162	0.664	0.226
LF image method			
LFRetinex [32]	20.540	0.705	0.177
L3Fnet [10]	24.552	0.803	0.141
Ours	**25.096**	**0.822**	**0.120**

C. Comparison

We compared our method with the other state-of-the-art low-light enhancement methods for the single images and LF images. The single image methods restore each view separately without utilizing the information of other views. We tested all these methods on the real-captured dark LF images to show their performance on the practical low-light restoration. PSNR, SSIM and LPIPS [44] are the metrics to evaluate the model performance.

First, these methods were tested on the L3F dataset [10], which were captured by Lytro Illum camera under extremely low-light conditions. Table I records the corresponding quantitative results of different methods, which suggests that our method can outperform the other single image and LF image methods, with higher PSNR and SSIM, and lower LPIPS.

The visual results of several scenes are presented in Fig. 8, including the central views, epipolar-plane images (EPIs) and zoomed patches. We can observe that the images restored by other methods have some color distortion with residual noise. Our method can achieve better luminance enhancement and noise suppression with little color distortion to obtain LF images with more compelling visual qualities.

We also captured some LF images in the low-light environment and tested different methods using them. The quantitative results are listed in Table II, which again reflects that our method can obviously outperform the others. The visual results under ISO $=$ 250 and 500 are shown in Fig. 9 and Fig. 10, respectively. We can see that DeepUPE [21] and Zero-DCE [22] can only enhance illumination but can not suppress noise caused by the low-light imaging as their models do not incorporate denoising module. RUAS [27] and URetinex-Net [26] both adopt the unfolding optimization approach to suppress noise. However, this approach is not effective to the situation with relatively serious noise. Hence, their restored LF images still suffer from obvious noise, resulting in poor restoration performance. In contrast, our method achieves better performance on the practical low-light restoration to obtain higher-quality LF images.

TABLE II
QUANTITATIVE RESULTS ON OUR CAPTURED LF IMAGES. BOLD: BEST.

Method	PSNR↑	SSIM↑	LPIPS↓
Single image method			
DeepUPE [21]	20.302	0.704	0.174
Zero-DCE [22]	18.814	0.692	0.206
RUAS [27]	21.421	0.730	0.167
KinD++ [25]	18.080	0.731	0.198
URetinex-Net [26]	20.890	0.756	0.157
LF image method			
LFRetinex [32]	21.911	0.833	0.099
L3Fnet [10]	21.433	0.815	0.112
Ours	**26.547**	**0.862**	**0.087**

Table III lists the average run time and model size of different methods. The run time is for restoring a $7 \times 7 \times 384 \times 512$ dark LF image on P100 GPU. We can find that our method is much more efficient and lightweight than the other LF image methods. Even if Zero-DCE [22] and RUAS [27] have very fast inference speed and small model size, their performances on dark LF restoration are worse than ours. Fig. 11 presents the model performance in terms of PSNR versus run time of different methods, which reflects that our method achieves a better balance between the performance and efficiency.

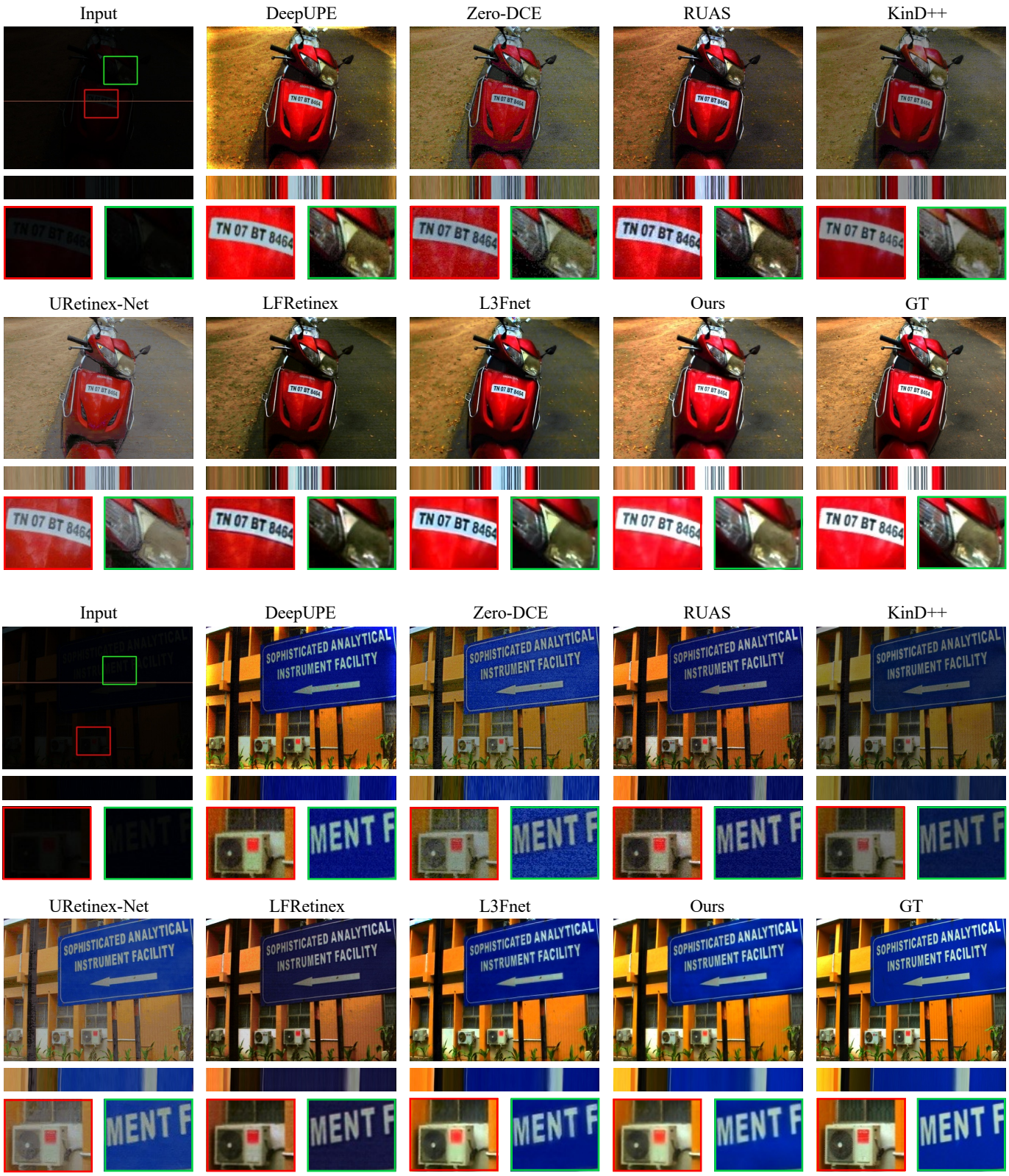


Fig. 8. Visual results of different methods on the L3F dataset [10]. The central view and EPIs are presented, with zoomed patches to show the local details.

D. Ablation studies

We conducted some ablation studies to validate our method design using the synthetic test dataset. The quantitative results of ablation study on the network architecture are recorded in Table IV. First, we validate the different heads for intermediate-task learning. We trained an extra model

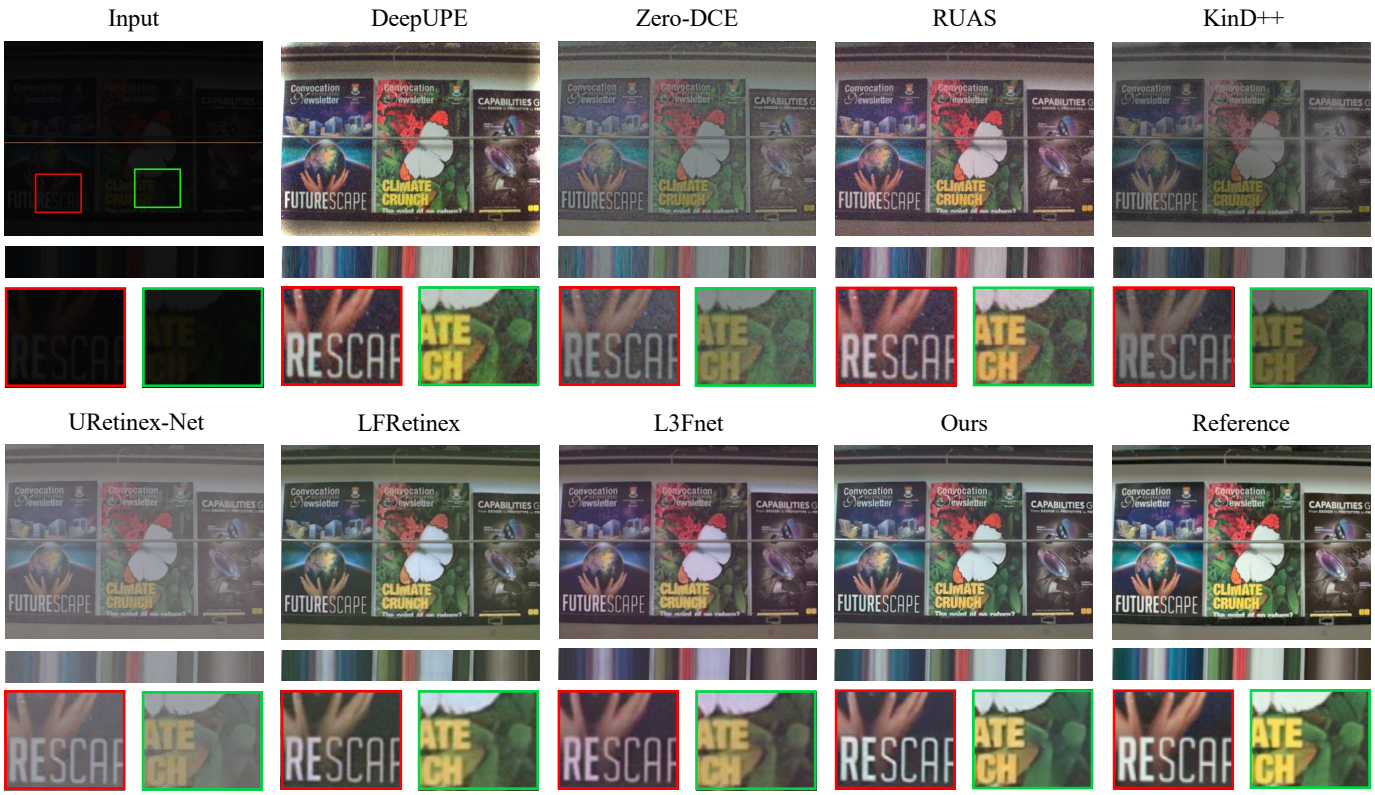


Fig. 9. Visual results of different methods on our captured low-light LF image with ISO = 200. The central view and EPIs are presented, with zoomed patches to show the local details.

TABLE III
RUN TIME AND MODEL SIZE OF DIFFERENT METHODS.

Method	Run time (s)	Parameters (M)
DeepUPE [21]	0.043	0.594
Zero-DCE [22]	0.002	0.079
RUAS [27]	0.007	0.004
KinD++ [25]	0.478	8.017
URetinex-Net [26]	3.609	0.340
LFRetinex [32]	3.214	3.697
L3Fnet [10]	1.638	3.725
Ours	0.017	1.466

without any head, which is equivalent to the full-scale image-to-image mapping. Its results are much worse than our default configuration, verifying the effectiveness of our intermediate-task learning. Moreover, we trained a model by removing the illumination head and denoising head (become direct mapping from $L_{\text{in}}^{1/4}$ to $L_{\text{re}}^{1/4}$), a model without denoising head, a model without high-frequency head, and a model with high-frequency head but without adaptive ratio adjustment. Their results suggests that the performance is degraded significantly if a direct mapping from $L_{\text{in}}^{1/4}$ to $L_{\text{re}}^{1/4}$ is learned, and if the denoising head is not employed. The performance has some decline if the high-frequency head is removed, and also has little decline if the adaptive ratio adjustment module for detail prediction is not incorporated. Then, we validate our angular transformer blocks by removing them and spatial transformer blocks by replacing them with the residual blocks. Their degraded performances prove the validity of modeling the global angular relationship and long-range spatial dependencies.

Some visual comparisons are presented in Fig. 12, where we can find that the model without illumination head and denoising head leads to obvious color distortion, the model without denoising head fail to suppress severe noise, and the model without high-frequency head can not preserve clear local details, compared with our default configuration. In addition, Fig. 13 shows the predicted high-frequency maps and their corresponding restoration results with and without the adaptive ratio adjustment module. It can be observed that more and clearer local details can be predicted by introducing adaptive ratio adjustment for the extremely low-light input, therefore obtaining higher-quality output.

Next, we validate our loss design by training extra models without using the illumination reference loss and SSIM loss, respectively, since the other loss terms are indispensable to achieve the intermediate and main tasks. Table V lists their quantitative results, which reflects that these two loss terms can help to improve the performance. Fig. 14 shows the estimated illumination maps and their corresponding restoration results with and without the illumination reference loss. It can be seen that the illumination map with the reference loss preserves clearer object boundaries and achieves better estimation for the light distribution. The restored image without the reference loss has some unexpected colors due to its poor illumination map, as shown in the circled region.

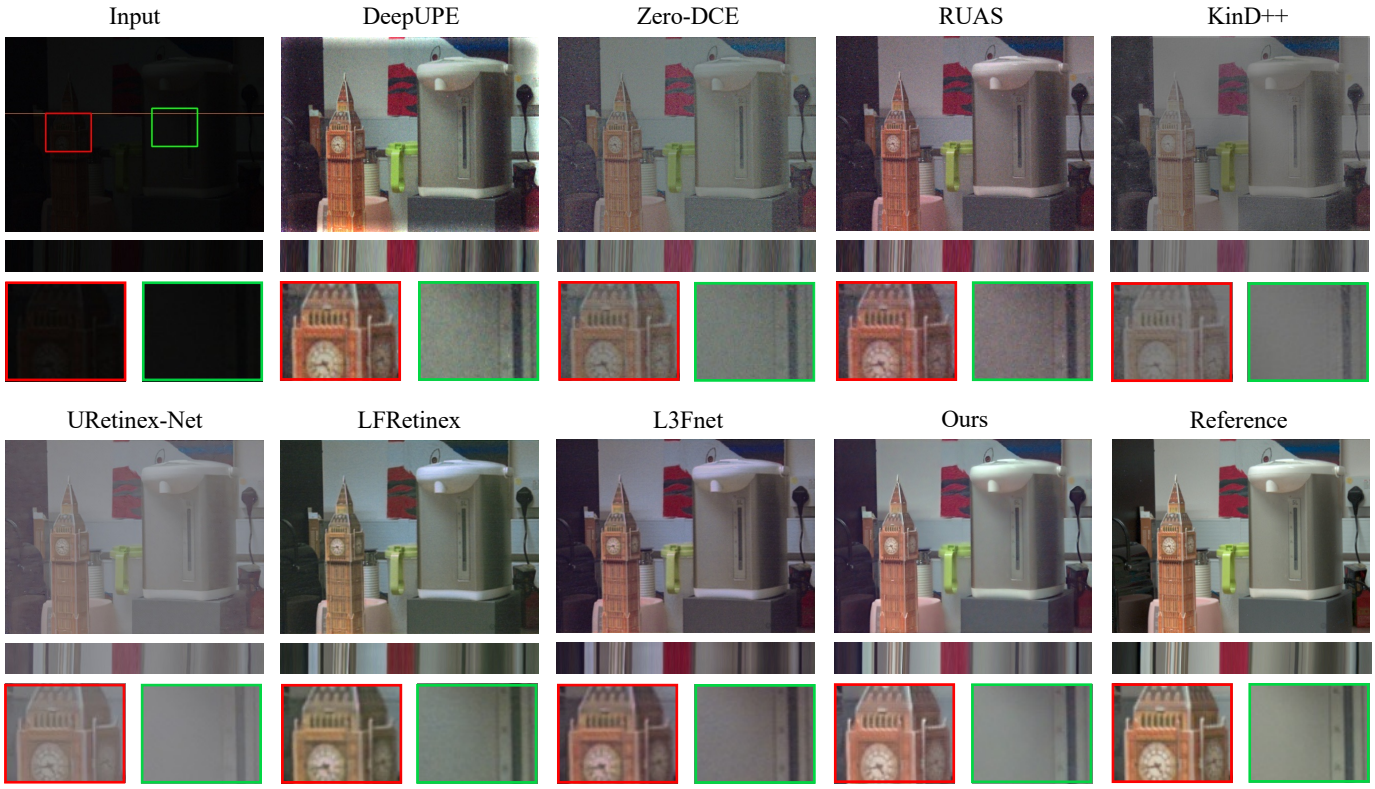


Fig. 10. Visual results of different methods on our captured low-light LF image with ISO = 500. The central view and EPIs are presented, with zoomed patches to show the local details.

TABLE IV
ABLATION STUDY ON THE NETWORK ARCHITECTURE. BOLD: BEST.

Different heads				Transformer blocks		Results		
Illumination	Denoising	High-frequency	Adaptive ratio adjustment	Angular	Spatial	PSNR↑	SSIM↑	LPIPS↓
				✓	✓	21.373	0.733	0.226
		✓	✓	✓	✓	21.625	0.746	0.219
✓		✓	✓	✓	✓	22.039	0.763	0.181
✓	✓			✓	✓	26.702	0.827	0.123
✓	✓	✓		✓	✓	27.526	0.845	0.102
✓	✓	✓	✓		✓	25.154	0.813	0.139
✓	✓	✓	✓	✓		26.230	0.821	0.132
✓	✓	✓	✓	✓	✓	**28.115**	**0.854**	**0.098**

TABLE V
ABLATION STUDY ON THE LOSS TERMS. BOLD: BEST.

Loss terms		Results		
Illumination reference	SSIM	PSNR↑	SSIM↑	LPIPS↓
	✓	27.436	0.830	0.121
✓		27.645	0.847	0.104
✓	✓	**28.115**	**0.854**	**0.098**

VI. CONCLUSION

In this paper, we propose the LRT, an efficient low-light restoration transformer for LF images, which contains multiple heads to implement denoising, luminance adjustment, refinement and detail enhancement, respectively, achieving progressive restoration from small scale to full scale. Moreover, we design an angular transformer block which employs a view-token scheme to model the angular relationship across all the views, and a multi-scale window-based transformer block to extract multi-scale local and global spatial features. In order to synthesize more realistic dark LF images, we estimate the noise parameters of LF camera under different ISOs and use them to simulate the corresponding noise. Our network was trained on the synthetic dataset and can generalize well to the real low-light scenarios. In addition, it can outperform the other state-of-the-art low-light enhancement methods with more effective noise suppression and luminance recovery.

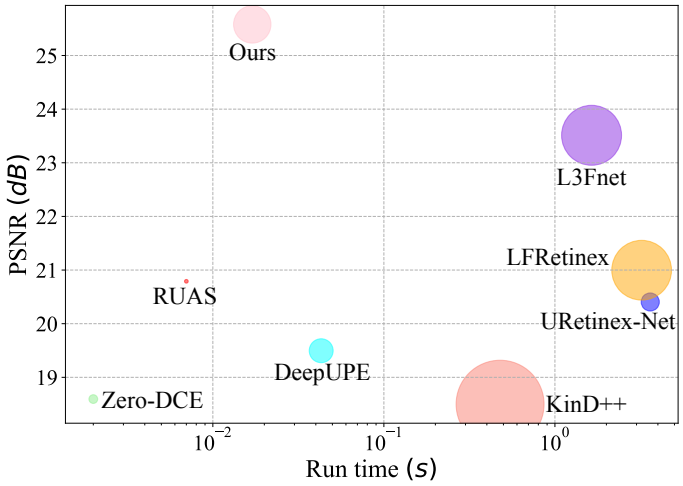


Fig. 11. The model performance (in PSNR) versus run time of different methods. The circle size of each method is proportional to its model size.

ACKNOWLEDGMENT

The work is supported in part by the Research Grants Council of Hong Kong (GRF 17200019, 17201620, 17200321) and by ACCESS — AI Chip Center for Emerging Smart Systems, Hong Kong SAR.

REFERENCES

[1] J. Fiss, B. Curless, and R. Szeliski, "Refocusing plenoptic images using depth-adaptive splatting," in *IEEE International Conference on Computational Photography*, 2014, pp. 1–9.

[2] Y. Wang, J. Yang, Y. Guo, C. Xiao, and W. An, "Selective light field refocusing for camera arrays using bokeh rendering and super-resolution," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 204–208, Jan. 2019.

[3] J. Shi, X. Jiang, and C. Guillemot, "A framework for learning depth from a flexible subset of dense and sparse light field views," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 5867–5880, 2019.

[4] J. Jin and J. Hou, "Occlusion-aware unsupervised learning of depth from 4-d light fields," *IEEE Transactions on Image Processing*, vol. 31, pp. 2216–2228, 2022.

[5] Y. Wang, T. Wu, J. Yang, L. Wang, W. An, and Y. Guo, "DeOcc-Net: Learning to see through foreground occlusions in light fields," in *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020.

[6] Y. Li, W. Yang, Z. Xu, Z. Chen, Z. Shi, Y. Zhang, and L. Huang, "Mask4D: 4d convolution network for light field occlusion removal," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 2480–2484.

[7] M. Zhang, W. Ji, Y. Piao, J. Li, Y. Zhang, S. Xu, and H. Lu, "LFNet: Light field fusion network for salient object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 6276–6287, 2020.

[8] J. Zhang, Y. Liu, S. Zhang, R. Poppe, and M. Wang, "Light field saliency detection with deep convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 4421–4434, 2020.

[9] J. Jin, J. Hou, J. Chen, and S. Kwong, "Light field spatial super-resolution via deep combinatorial geometry embedding and structural consistency regularization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2260–2269.

[10] M. Lamba, K. K. Rachavarapu, and K. Mitra, "Harnessing multi-view perspective of light fields for low-light imaging," *IEEE Transactions on Image Processing*, vol. 30, pp. 1501–1513, 2021.

[11] 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 Transactions on Image Processing*, vol. 28, no. 5, pp. 2319–2330, 2019.

[12] Y. Wang, L. Wang, J. Yang, W. An, J. Yu, and Y. Guo, "Spatial-angular interaction for light field image super-resolution," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 290–308.

[13] X. Fu, D. Zeng, Y. Huang, X.-P. Zhang, and X. Ding, "A weighted variational model for simultaneous reflectance and illumination estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] X. Guo, Y. Li, and H. Ling, "LIME: Low-light image enhancement via illumination map estimation," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 982–993, 2017.

[15] M. Li, J. Liu, W. Yang, X. Sun, and Z. Guo, "Structure-revealing low-light image enhancement via robust retinex model," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2828–2841, 2018.

[16] X. Ren, M. Li, W.-H. Cheng, and J. Liu, "Joint enhancement and denoising method via sequential decomposition," in *IEEE International Symposium on Circuits and Systems*, 2018, pp. 1–5.

[17] E. H. Land, "The retinex theory of color vision," *Sci. Amer*, vol. 237, no. 6, pp. 108–129, 1977.

[18] F. Lv, F. Lu, J. Wu, and C. Lim, "MBLLEN: Low-light image/video enhancement using CNNs," in *British Machine Vision Conference*, 2018.

[19] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang, "EnlightenGAN: Deep light enhancement without paired supervision," *IEEE Transactions on Image Processing*, vol. 30, pp. 2340–2349, 2021.

[20] L.-W. Wang, Z.-S. Liu, W.-C. Siu, and D. P. K. Lun, "Lightening network for low-light image enhancement," *IEEE Transactions on Image Processing*, vol. 29, pp. 7984–7996, 2020.

[21] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, and J. Jia, "Underexposed photo enhancement using deep illumination estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[22] C. Guo, C. Li, J. Guo, C. C. Loy, J. Hou, S. Kwong, and R. Cong, "Zero-reference deep curve estimation for low-light image enhancement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[23] L. Ma, T. Ma, R. Liu, X. Fan, and Z. Luo, "Toward fast, flexible, and robust low-light image enhancement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[24] Y. Zhang, J. Zhang, and X. Guo, "Kindling the darkness: A practical low-light image enhancer," in *ACM International Conference on Multimedia*, 2019, pp. 1632–1640.

[25] Y. Zhang, X. Guo, J. Ma, W. Liu, and J. Zhang, "Beyond brightening low-light images," *International Journal of Computer Vision*, vol. 129, pp. 1013–1037, 2021.

[26] W. Wu, J. Weng, P. Zhang, X. Wang, W. Yang, and J. Jiang, "URetinex-Net: Retinex-based deep unfolding network for low-light image enhancement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 5901–5910.

[27] R. Liu, L. Ma, J. Zhang, X. Fan, and Z. Luo, "Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 10 561–10 570.

[28] M. Lamba and K. Mitra, "Fast and efficient restoration of extremely dark light fields," in *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, pp. 1361–1370.

[29] Z. Ge, L. Song, and E. Y. Lam, "Light field image restoration in low-light environment," in *Future Sensing Technologies*, ser. Proceedings of the SPIE, vol. 11525, 2020, p. 115251H.

[30] N. Meng, H. K.-H. So, X. Sun, and E. Y. Lam, "High-dimensional dense residual convolutional neural network for light field reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 873–886, 2021.

[31] S. Zhang and E. Y. Lam, "Learning to restore light fields under low-light imaging," *Neurocomputing*, vol. 456, pp. 76–87, 2021.

[32] S. Zhang and E. Y. Lam, "An effective decomposition-enhancement method to restore light field images captured in the dark," *Signal Processing*, vol. 189, p. 108279, 2021.

[33] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-Unet: Unet-like pure transformer for medical image segmentation," *arXiv preprint*, 2021.

[34] Z. Liang, Y. Wang, L. Wang, J. Yang, and S. Zhou, "Light field image super-resolution with transformers," *IEEE Signal Processing Letters*, vol. 29, pp. 563–567, 2022.

[35] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 568–578.

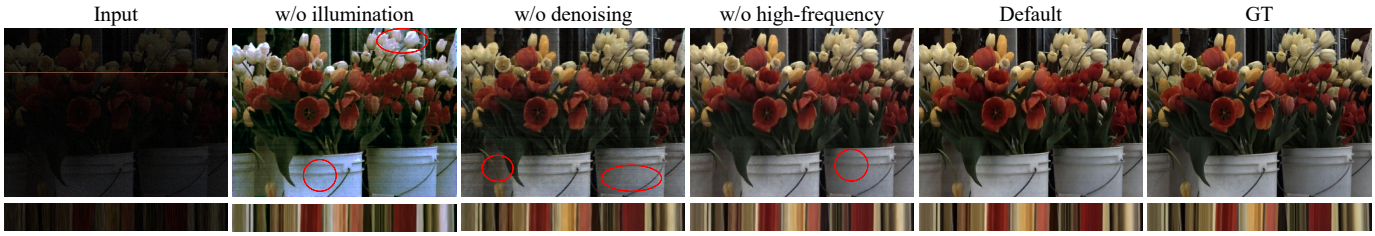


Fig. 12. Visual results of different configurations on the network architecture. The circled regions reflect poor performance of the other configurations.

Fig. 13. The high-frequency maps and restoration results without and with the adaptive ratio adjustment module (ARAM). The circled regions indicates blurry details compared with our default output.

Fig. 14. The illumination maps and restoration results without and with the illumination reference loss.

[36] K. G. Dan Hendrycks, "Gaussian error linear units (GELUs)," *arXiv preprint*, 2016.

[37] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "PVT v2: Improved baselines with pyramid vision transformer," *Computational Visual Media*, vol. 8, p. 415–424, 2022.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[39] K. Wei, Y. Fu, Y. Zheng, and J. Yang, "Physics-based noise modeling for extreme low-light photography," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[40] N. K. Kalantari, T. C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM Transactions on Graphics*, vol. 35, no. 6, 2016.

[41] R. Shah, G. Wetzstein, A. S. Raj, and M. Lowney, "Stanford lytro light field archive," 2016.

[42] M. Rerabek and T. Ebrahimi, "New light field image dataset," in *International Conference on Quality of Multimedia Experience*, 2016.

[43] S. Zhang and E. Y. Lam, "An effective image restorer: Denoising and luminance adjustment for low-photon-count imaging," *arXiv preprint*, 2021.

[44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.