Single-shot Phase Retrieval from Linear Canonical Transform with Deep Physical Neural Networks

Wei Lin and Yulong Mao

Abstract—**Phase retrieval is to reconstruct signals (images) from their Fourier magnitude measurements, and notably applied in the computational imaging realm. However, conventional phase retrieval methods rely on redundant sampling, which entails significantly higher computational costs during computational imaging processes. Hence, single-shot phase retrieval has garnered significant attention due to its ability of a single exposure from Fourier intensity measurement, thereby eliminating the need for any additional constraints on the measured data. To improve the quality of reconstruction images, we propose a method for phase retrieval from the single linear canonical transform (LCT) magnitude measurement with physical neural network, namely SPRLCT-PNN. Deep physical neural networks (PNN) include the LCT-FPB module, upsampling and downsampling module, unified physics-aware attention network module, and post-processing module. Furthermore, PNN priors can effectively integrate LCT measurements and alleviate the ambiguities of phase retrieval. Through numerical simulations, we demonstrate that the proposed method achieves significantly improved reconstruction performance from a single LCT measurement compared to the Fourier transform magnitude. Empirical results demonstrate the superior performance of the proposed method on the Cell 8 and Fashion-MNIST datasets.**

Index Terms—**Single-shot, phase retrieval, linear canonical transform, physical prior, neural network.**

I. Introduction

Phase retrieval problem aims to recover the phase information only from the intensity measurements [1]. Phase retrieval plays a crucial role in various applications, including computational imaging [2], X-ray crystallography [3], computational microscopy [4], optical transient imaging, and biological dynamic imaging [5].

Multiple measurement phase retrieval problems are addressed by Fourier domain intensity differences. Gerchberg-Saxton (GS) algorithm [6] and hybrid input-output (HIO) [7] algorithm are both from the Fourier transform domain. These iterative techniques project between subspaces with their Fourier magnitude information and the desired constraints, achieving widespread practical use. However, their limited accuracy in addressing the phase retrieval problem may result in artifacts during the alternating projection process. However, alternate projection method faces several challenges, particularly in terms of computational complexity and convergence speed. One major drawback is that each iteration requires projecting the current estimate onto multiple subspaces, which can be computationally expensive, especially for high-dimensional data. Candes et al. [8] present a nonconvex formulation of the phase retrieval problem as well as a concrete solution algorithm to tackle the high computational costs. The method guarantees near-optimal recovery with minimal measurements. Horisaki et al. [9] utilize a coded mask to sieve the propagating field from an object under partially coherent light, capturing it on a sensor with support constraint. While Candes et al. [10] propose PhaseLift, which modulates the signal of interest and captures the intensity of its diffraction pattern. To address the inefficiency, Li et al. [11] propose the sparse reweighted thresholded Wirtinger flow algorithm, which exhibits a faster convergence rate and shorter running time. [12] introduces a variational model for phase retrieval that uses total variation regularization as an image prior and applies maximum a posteriori estimation for a Poisson noise model. [13] utilizes a distributed amplitude and phase conditional generative adversarial network to achieve high-quality phase and amplitude images simultaneously. But these methods have limitations and complexities in practice. In particular, under fixed sensor conditions, adjusting the aperture near the sample to capture multiple intensity images complicates the phase-retrieval iterative process.

Despite these successes, the need for overlapped and oversample measurements significantly increases computational costs, making it unsuitable for dynamic ground truth. Additionally, in 3D imaging applications, collecting multiple observations is extremely challenging, highlighting the urgent need for reliable single-shot phase retrieval methods [14]. With the recent advancements in deep learning techniques for computational imaging, there has been a growing interest in using data-driven methods for single-shot phase retrieval. The goal of this approach is to reconstruct the object from a single diffraction pattern, bypassing the need for multiple measurements by utilizing labeled data to train a neural network that learns the inverse mapping of the single-shot measurement function. [15] utilizes deep neural networks (DNNs) to solve inverse problems in computational imaging, and also shows that a DNN can recover a phase object using a lensless imaging system from a raw intensity image captured at a distance.

Ye et al. [16] proposes PPRNet, a feedforward neural network for phase retrieval (PR), which achieves faster and more accurate performance compared to traditional physics-driven methods. Additionally, the self-supervised reconstruction approach leverages the fast discrete FrFT algorithm and untrained neural network priors, achieving superior results. Traditional deep learning relies heavily on large datasets to learn this map, lacking an in-depth understanding of physical processes, which affects the generalization and stability of these methods.

Manuscript received March 18, 2025; revised July 25, 2025.

This work was supported in part by the Natural Science Basic Program of Shaanxi (Program No.2025JC-YBMS-078), and the Graduate Innovation and Practical Ability Training Project of Xi'an Shiyou University (YCS23213184).

Wei Lin is an engineer of the Network and Data Center, Xi'an Shiyou University, Xian, China, 710065 (e-mail: linwei@xsyu.edu.cn).

Yulong Mao is a postgraduate student of the School of Science, Xi'an Shiyou University, Xian, China, 710065 (e-mail:1141628827@qq.com).

Specifically, measurements from LCT retain the amplitude and phase information of the ground truth signal, providing a generalized spatial frequency representation. This facilitates overcoming spatial blurring issues in phase retrieval. The LCT provides the feasibility of recovering the original signal from a single measurement. The contributions of this paper are summarized as follows:

We present the method for single-shot phase retrieval from linear canonical transform via deep physics neural network, called SPRLCT-PNN.

To enhance the reconstruction result, we introduce a deep PNN that includes the LCT-FPB module, upsampling and downsampling modules, a unified physics-aware attention network module, and a post-processing module.

We conduct experiments on various datasets, and the empirical results demonstrate the competitive performance of our proposed method. An ablation study is presented to validate the effectiveness of the PNN modules.

The rest of this paper is as follows. Section 2 introduces the PRSLCT model and presents the PNN framework. Section 3 shows the experimental setup and training strategy. Section 4 gives an ablation study. Section 5 demonstrates the superiority of the proposed method on Cell 8 and Fashion-MNIST datasets. Section 6 draws a conclusion and future research work.

II. Proposed method

In this section, we will briefly show linear canonical transform of signal and the overall framework.

A. Phase retrieval from linear canonical transform

The phase retrieval problem involves recovering images from intensity measurements, complicated by the inability to directly measure phase. In mathematical terms, it requires recovering an n-dimensional vector x from measurement value y:

$$y = |Ax| \tag{1}$$

where A is a matrix of size $m \times n$, m is the number of measurement values, and n is the dimension of the vector x. Furthermore, the amplitude and phase of signal $F(Y)$ are usually expressed as:

$$F(Y) = |F(Y)| e^{j \cdot \text{phase}(F(Y))} \tag{2}$$

where the amplitude $|F(Y)|$ can be directly measured using CCD or CMOS camera sensors.

For signal $f(t)$ its Linear Canonical Transform $b \neq 0$, we have

$$F_{(a,b,c,d)}(u) = L_{(a,b,c,d)}(f(t)) \tag{3}$$

and

$$L_{(a,b,c,d)}(f(t)) = \sqrt{\frac{1}{j2\pi b}} \int_{-\infty}^{+\infty} e^{j\left(\frac{a}{2b}t^2 - \frac{1}{b}tu + \frac{d}{2b}u^2\right)} f(t)dt \tag{4}$$

where j represents the imaginary unit, satisfying $j^2 = -1$, Especially, if $b = 0$, the LCT becomes a pure phase transform:

$$F_{(a,0,c,d)}(u) = \sqrt{d} e^{\frac{jcd}{2}u^2} f(du) \tag{5}$$

Parameters a and d are pivotal in governing phase modulation, which can emulate the phase alterations of light waves as they traverse optical media characterized by varying refractive indices. Concurrently, it addresses the phase response of signals navigating through systems that are subject to temporal changes within the field of signal processing. Parameter b modulates the scale of the transformation, which can be likened to modifying the focal length of a light beam to adjust the size of an image. Similarly, in signal processing, it enables the compression or expansion of signals in the time domain to align with varying sampling frequencies. Parameter c influences the resolution and the local properties of the transform. In the context of time-frequency analysis, adjusting the parameter c allows for fine control over the precision and locality of the analysis.

If the parameters a, b, c, d satisfy $ad - bc = 1$, then the transform is unitary and invertible.

B. Network architecture

The overall network architecture of the proposed method is shown in Figure 1. The framework of the proposed method consists of four modules to enhance the phase retrieval reconstruction results. The LCT-FPB module preprocesses input data in the LCT domain, transforming it into an efficient feature representation for convolution. The downsampling and upsampling modules ensure effective feature extraction and reconstruction. At the core of the model, the Unified Physics-Aware Attention Network (UPAN) integrates physical constraints with deep learning, improving the model's robustness. Within UPAN, the Physics-Aware Module (PAM) is employed to capture essential physics-driven features, while the Attention-Based Fusion Module (AFM) facilitates the integration of multi-scale information. The Feature Refinement Module (FRM) enhances local details like edges and textures while reducing redundancy. It consists of three ConvB blocks, each containing convolution, batch normalization, and activation. Finally, a post-processing module is applied to refine the output, further enhancing the accuracy and reliability of the predictions. The role and implementation details of each module are further elaborated in the following subsections.

1) The LCT feature processing block: The LCT Feature Processing Block (LCT-FPB) integrates preprocessing, decomposition, and fusion steps into a unified framework.

First, the input data $X \in R^{H \times W}$ from linear canonical transform is separated the magnitude and phase . The magnitude can be expressed as:

$$|X|_{\text{norm}} = \frac{|X| - \min(|X|)}{\max(|X|) - \min(|X|)} \tag{6}$$

where $|X|$ is the magnitude of input data X from linear canonical transform. The original distribution of the phase data $\arg(X)$ remains unchanged within the range $[-\pi, \pi]$ simultaneously. The preprocessed feature representation is $X_{\text{pre}} = \left\{ |X|_{\text{norm}}, \arg(X) \right\}$.

The frequency domain data X_{pre} is decomposed into multi-scale high-frequency and low-frequency components with discrete wavelet transform. The low-frequency features are extracted from the low-frequency approximation components

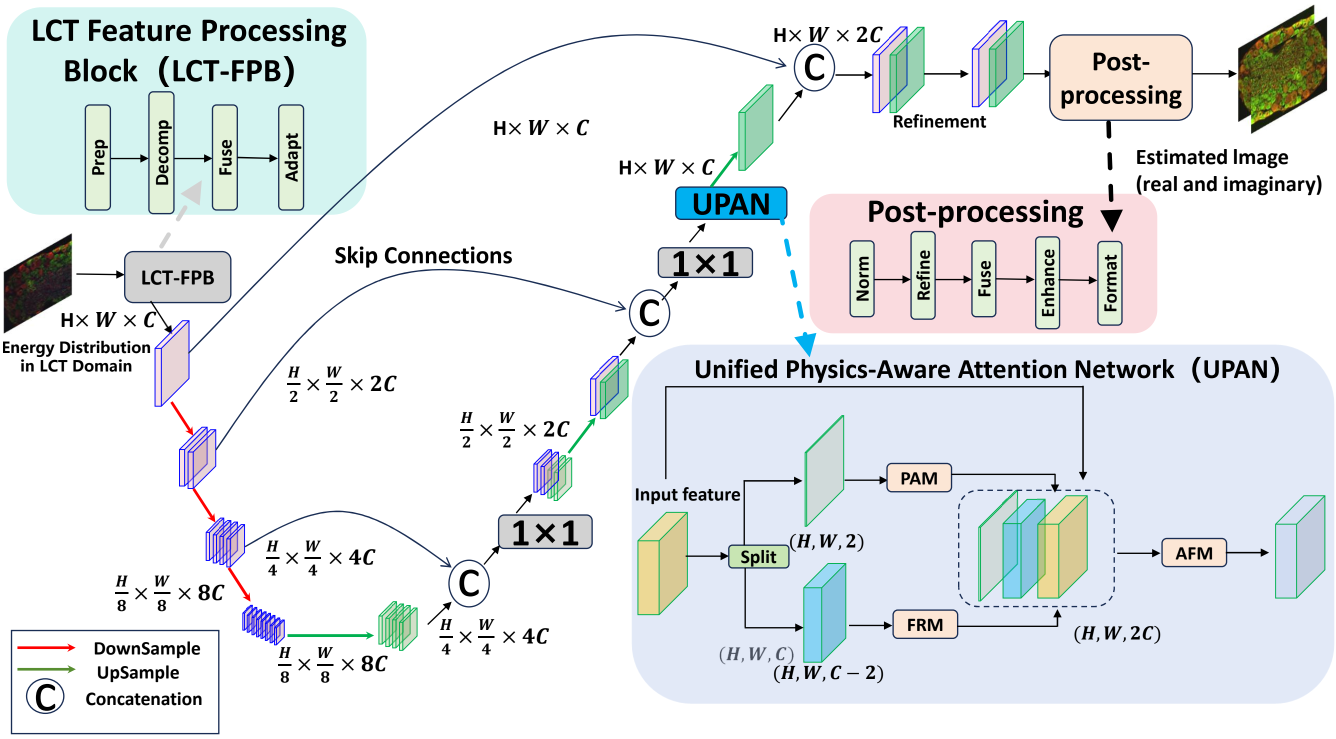

Fig. 1 The overall framework of the proposed method.

of the wavelet transform, expressed as:

$$X_{\text{low},j} = W_{\text{low},j}(X_{\text{pre}}) = \sum_{m=1}^{H}\sum_{n=1}^{W} X_{\text{pre}}[m,n]\cdot\psi_{\text{low},j}[m,n] \tag{7}$$

where $W_{\text{low},j}(\cdot)$ represents the low-frequency wavelet coefficient extraction operation at the j-th level, and $\psi_{\text{low},j}[m,n]$ is the corresponding low-frequency approximation kernel of the wavelet. The low-frequency components mainly provide global structural information, representing the smooth part of the signal. The high-frequency features are extracted from the high-frequency detail components of the wavelet transform, containing multi-directional details in the horizontal, vertical, and diagonal directions, expressed as:

$$X_{\text{high}},j^{\text{dir}} = W^{\text{dir}}_{\text{high},j}(X_{\text{pre}}) = \sum_{m=1}^{H}\sum_{n=1}^{W} X_{\text{pre}}[m,n]\cdot\psi^{\text{dir}}_{\text{high},j}[m,n] \tag{8}$$

where $W^{\text{dir}}_{\text{high},j}(\cdot)$ is the high-frequency wavelet coefficient extraction operation in the specific direction at the j-th level, and $\psi^{\text{dir}}_{\text{high},j}[m,n]$ is the corresponding directional wavelet kernel, with dir $\in \{H,V,D\}$ representing the horizontal, vertical, and diagonal directions, respectively. By combining the high-frequency and low-frequency components, the input signal can be represented as

$$X_{\text{pre}} = X_{\text{low},J} + \sum_{j=1}^{J}\left(X^{H}_{\text{high},j} + X^{V}_{\text{high},j} + X^{D}_{\text{high},j}\right) \tag{9}$$

where $X_{\text{low},J}$ is the highest-level low-frequency component, primarily capturing global information, and $X^{\text{dir}}_{\text{high},j}$ represents local details at different scales and directions. To fully utilize the high-frequency and low-frequency features obtained from the decomposition, the high-frequency $H(X_{\text{pre}})$ and low-frequency features $L(X_{\text{pre}})$ are combined into a joint feature representation by concatenating along the channel dimension, it is formulated as

$$X_{\text{concat}} = \text{Concat}\left(L(X_{\text{pre}}), H(X_{\text{pre}})\right) \tag{10}$$

Subsequently, a 1×1 convolution operation is utilized to the concatenated features for fusion, which helps reduce redundant information and enhance the correlation between features. The fused features reduce computational complexity while maintaining the collaborative effects of global and local characteristics. The fused frequency domain features are transformed by the mapping function $T(\cdot)$ to generate a spatial domain feature representation. We have

$$T_1 = T(X_{\text{fused}}) \tag{11}$$

where $T(\cdot)$ is responsible for adjusting the dimensions and distribution of the features to yield the input requirements of the convolution operation. Thus, the overall processing flow of the LCT-FPB module can be expressed as:

$$T_1 = T\left(\text{Conv}_{1\times1}\left(\text{Concat}\left(L(X_{\text{pre}}), H(X_{\text{pre}})\right)\right)\right) \tag{12}$$

2) Downsampling and upsampling module: In the downsampling stage, the spatial dimensions of the feature map are progressively compressed to extract deeper semantic features of the signal. The process is as follows:

$$T_{i+1}[x',y',c] = \sum_{k_x}\sum_{k_y}\sum_{c'} T_i[x'+k_x, y'+k_y, c'] \cdot W[c',k_x,k_y,c] \tag{13}$$

where T_i is the input feature map at the current layer, T_{i+1} is the downsampled output feature map, and W represents the convolution kernel weights. (k_x, k_y) are the spatial dimensions of the kernel, and c' and c are the indices of the input and output channels, respectively. Through three convolution operations with a stride of 2, the feature map resolution is progressively compressed from $H \times W \times C$ to $H/8 \times W/8 \times 8C$. Each operation also increases the number of channels ($C \rightarrow 2C \rightarrow 4C \rightarrow 8C$), thereby extracting deeper semantic features. During downsampled process, dynamic range normalization, frequency band feature decomposition, and efficient multi-frequency band fusion from the LCT domain input data are achieved.

In the upsampling stage, the module progressively increases the spatial resolution of the feature map while effectively preserving high-frequency detail information through skip connections.

For any target resolution coordinates (x', y') in the output feature map, the scaling factors $s_x = H'/H$ and $s_y = W'/W$ map it to the floating-point position (x, y) in the input feature map according to bilinear interpolation. The process is formulated as

$$x = \frac{x'}{s_x}, \quad y = \frac{y'}{s_y} \tag{14}$$

The four nearest pixels to the floating-point position (x, y) in the input feature map $(x_1, y_1), (x_2, y_1), (x_1, y_2), (x_2, y_2)$ are determined, and their pixel values are denoted as $T[x_1, y_1], T[x_2, y_1], T[x_1, y_2], T[x_2, y_2]$, respectively. Finally, the output value is calculated using the weighted formula

$$\begin{aligned} T_{\text{output}}[x', y'] = {} & w_1 T[x_1, y_1] + w_2 T[x_2, y_1] \\ & + w_3 T[x_1, y_2] + w_4 T[x_2, y_2] \end{aligned} \tag{15}$$

where the weights w_1, w_2, w_3, w_4 represent the relative distance of the floating-point position to the four pixels and satisfy $w_1 + w_2 + w_3 + w_4 = 1$. The skip connection in the network directly transmits high-resolution features from the downsampling stage to the corresponding upsampling stage, enabling effective multi-scale feature fusion and the preservation of detail information. And we have

$$T_{i-1} = \text{Upsample}(T_i) + T_{\text{skip}, i-1} \tag{16}$$

where Upsample(T_i) denotes the deep features restored through bilinear interpolation, and $T_{\text{skip}, i-1}$ represents the high-resolution shallow features provided by the skip connection.

3) Unified physics-aware attention network module: UPAN network module combines physical constraints with data-driven features to generate physically reasonable feature representations through global consistency constraints and local feature fusion. The feature processing in UPAN starts with the separation of input features, dividing the input feature $x \in R^{H \times W \times C}$ into two paths: the physically related feature path and the local detail feature path. The physically related features are processed by the PAM to enhance global consistency and ensure the features adhere to physical laws, while the local detail features are extracted and enhanced by the FRM to capture fine-grained local information. The results from both paths are concatenated along the channel dimension to form the fused feature representation:

$$f_{\text{concat}} \in R^{H \times W \times 2C} \tag{17}$$

To further optimize the fused features, UPAN performs global pooling to extract the importance weights of each channel. On a per-channel basis, the features are weighted to highlight key features while suppressing irrelevant or redundant ones. Then, the fused features undergo convolutional operations to compress the channel dimension from $2C$ to the original C, resulting in the optimized output feature

$$f_{\text{out}} \in R^{H \times W \times C} \tag{18}$$

Physics-aware module In single-phase measurement phase retrieval, the intensity measurements from LCT provide essential physical constraints, while the spatial domain features capture better local detail information. The structure of Physics-Aware Module is illustrated in Figure 2. The design goal of PAM is to incorporate these physical constraints into the learning process, while also enhancing feature representation through an attention mechanism.

PAM maps the input feature map u_k into the LCT domain to generate the complex feature $U_k = |U_k| e^{j\varphi_k}$. In the LCT domain, the magnitude $|U_k|$ represents intensity information, and the phase φ_k captures high-frequency details. To introduce physical constraints, we update the magnitude information using the measured intensity $S(X)$ as follows

$$U_k' = \sqrt{S(X)} e^{j\varphi_k} \tag{19}$$

The updated LCT domain feature U_k' is then mapped back to the spatial domain through an inverse LCT operation, generating a new feature map u_k'. However, even though the physical constraints are explicitly applied, the feature map may still lack accurate representation of local details. Thus, we design a convolutional neural network $g_k(\cdot)$ to learn the residual information. After extracting local features from u_k' through the convolutional network, the updated feature is obtained by weighting the initial feature map u_k,

$$u_{k+1} = g_k(u_k') + \beta_k u_k \tag{20}$$

This design retains the global consistency of the physical constraints while enhancing the expression of local details. The multi-layer design of PAM further improves feature learning.

Feature-refinement module The FRM extracts local detail features, such as edge and texture details, from the input features $F_{\text{FRM}} \in R^{H \times W \times (C-2)}$, while suppressing redundant features. FRM employs a shallow structure consisting of three consecutive convolution blocks (ConvB), each including a convolution operation, a batch normalization layer, and a non-linear activation function. The refined feature $F_{\text{FRM}}' \in R^{H \times W \times (C-2)}$ can be described as follows

$$F_{\text{FRM}}' = \sigma\left(\text{Conv}_3\left(\sigma\left(\text{Conv}_2\left(\sigma\left(\text{Conv}_1(F_{\text{FRM}})\right)\right)\right)\right)\right) \tag{21}$$

where Conv$_k$ represents the k-th layer convolution operation and $\sigma(\cdot)$ is the activation function (ReLU). This refinement step enhances the expression of detailed features at each layer, ensuring that the output features provide rich local information for subsequent fusion, complementing the global information extracted by UPAN.

Attention-based fusion module The structure of Attention-Based Fusion Module is shown in Figure 3. It dynamically fuses the global context features $F_{\text{PAM}}' \in R^{H \times W \times 2}$ extracted by PAM with the local features $F_{\text{FRM}}' \in$

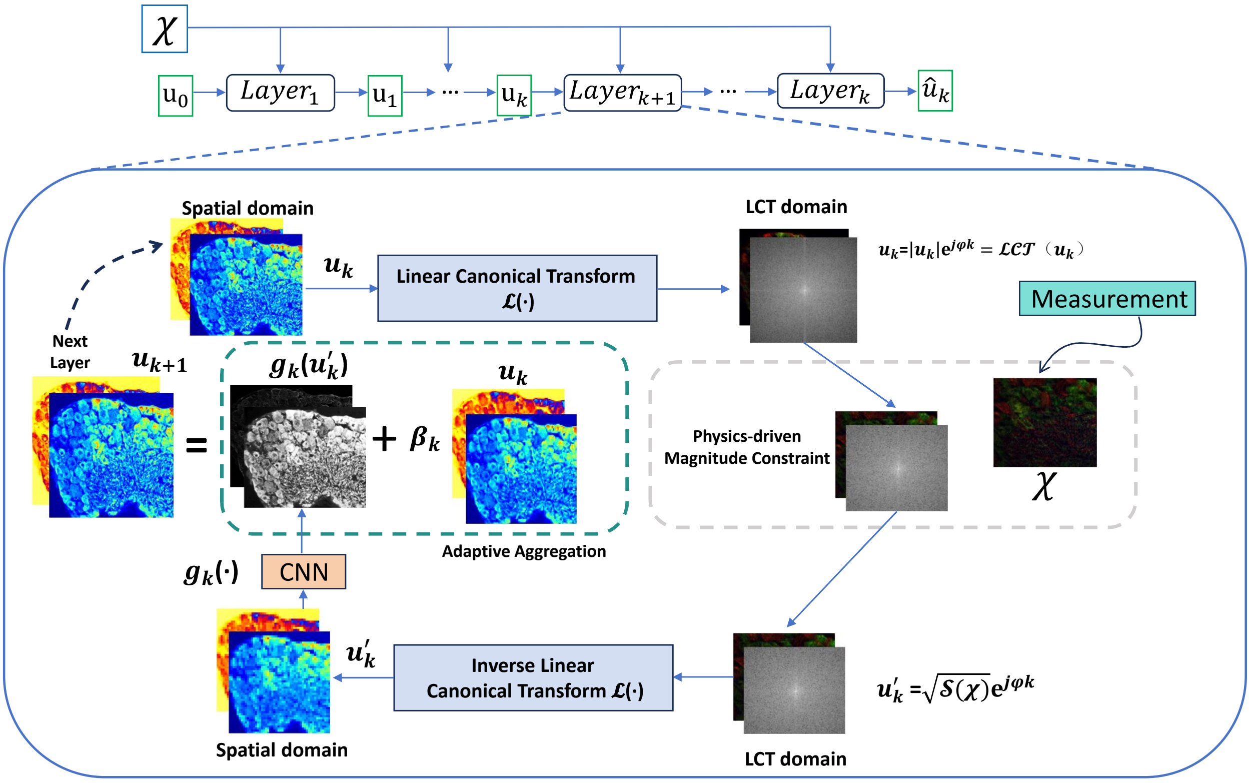

Fig. 2 Structure of Physics-Aware Module.

$R^{H\times W\times(C-2)}$ extracted by FRM, resulting in a more expressive fused feature.

The output of PAM and FRM is concatenated along the channel dimension to obtain the fused feature:

$$F_{\text{concat}} = \text{Concat}(F'_{\text{PAM}}, F'_{\text{FRM}}) \in R^{H\times W\times 2C} \tag{22}$$

The global average pooling operation is applied to the fused features F_{concat} to compress them into a channel description vector $F_{\text{global}} \in R^{1\times 1\times 2C}$:

$$F_{\text{global}}(c) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_{\text{concat}}(i,j,c) \tag{23}$$

The channel description vector is then passed through two fully connected layers (FC), with a non-linear activation function to generate the channel attention weights $A \in R^{1\times 1\times 2C}$:

$$A = \text{Softmax}\left(\text{FC}_2\left(\sigma\left(\text{FC}_1\left(F_{\text{global}}\right)\right)\right)\right) \tag{24}$$

Finally, the attention weights are multiplied with the fused features per channel to generate the final fused feature output $F_{\text{AFM}} \in R^{H\times W\times C}$:

$$F_{\text{AFM}}(i,j,c) = A(c)\cdot F_{\text{concat}}(i,j,c) \tag{25}$$

This allows dynamic weight allocation between global and local features, enabling improved feature representation.

4) Post-processing module: In the final stage of image recovery, a series of steps such as normalization, feature optimization, dynamic fusion, detail enhancement, and output formatting are combined to gradually optimize the feature distribution and generate high-quality recovered images.

The input feature map is standardized in terms of mean and standard deviation. The specific process is given by:

$$F_{\text{norm}}(i,j) = \frac{F(i,j)-\mu_F}{\sigma_F} \tag{26}$$

where $F(i,j)$ represents the pixel value of the input feature map, and μ_F and σ_F are the mean and standard deviation of the feature map, respectively, and

$$\mu_F = \frac{1}{H\cdot W}\sum_{i=1}^{H}\sum_{j=1}^{W} F(i,j) \tag{27}$$

$$\sigma_F = \sqrt{\frac{1}{H\cdot W}\sum_{i=1}^{H}\sum_{j=1}^{W}(F(i,j)-\mu_F)^2} \tag{28}$$

The feature map is enhanced by combining local convolution with a channel attention mechanism, which improves the local detail representation ability of the features. This process is obtained as $F_{\text{refine}} = \text{Conv}_{\text{local}}(F_{\text{norm}}) + \text{Attention}_{\text{channel}}(F_{\text{norm}})$, where $\text{Conv}_{\text{local}}$ captures local features such as edges and corners through convolution, and $\text{Attention}_{\text{channel}}$ emphasizes the importance of different feature channels through adaptive weight calculation. The channel attention mechanism extracts the global response for each channel using global average pooling, and then uses a two-layer fully connected network for nonlinear mapping

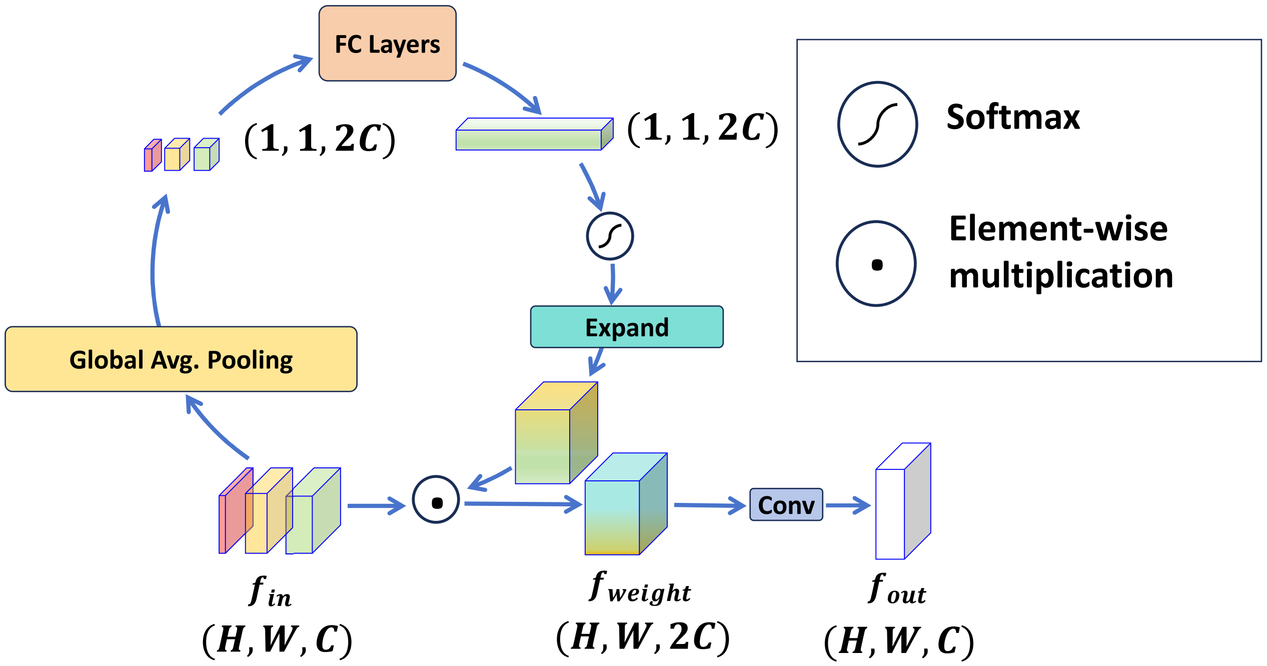

Fig. 3 The structure of Attention-Based Fusion Module.

to generate the channel weights as

$$w_c = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot g_c)) \tag{29}$$

In the feature fusion stage, features from different sources are integrated using dynamic weight adjustment. The fusion operation calculates the weight matrix through the attention mechanism and combines the features as $F_{\text{fuse}} = \text{Softmax}(W) \cdot \begin{bmatrix} F_{\text{refine}} \\ F_{\text{global}} \end{bmatrix}$, where $\text{Softmax}(W)$ denotes the normalization of the weight matrix, and $\begin{bmatrix} F_{\text{refine}} \\ F_{\text{global}} \end{bmatrix}$ represents the joint representation of the locally optimized and global features.

In the detail enhancement stage, convolution operations combined with a nonlinear activation function are applied as follows

$$F_{\text{enhance}} = \sigma(\text{Conv}_{\text{enhance}}(F_{\text{fuse}})) \tag{30}$$

where $\text{Conv}_{\text{enhance}}$ is a high-resolution convolution layer that captures subtle texture features, and $\sigma(\cdot)$ represents the ReLU activation function, which further enhances the visual clarity and detail expression.

Finally, in the output formatting stage, the enhanced feature map is adjusted to fit the target output range. The process is as follows

$$\hat{I} = \text{Clamp}(F_{\text{enhance}} \cdot (\gamma + \delta)), \tag{31}$$

where γ and δ are the scaling factor and the offset for feature stretching, respectively, and $\text{Clamp}(\cdot)$ ensures that the pixel values are within the range $[0, 1]$.

III. Experimental setup and Training strategy

A. Experimental setup

The proposed reconstruction method is based on the PyTorch version 2.0.0 platform, using Python 3.9 and leveraging an NVIDIA RTX 3060 GPU for computational acceleration. We conducted simulations on two different datasets: the Cell8 dataset [24] and the Fashion-MNIST dataset [25]. The training datasets are converted into grayscale images and resized to 128 × 128. Then, two forms of complex-valued images are generated using the preprocessed images $x \in [0, 1]$: pure phase images and pure magnitude images. To generate pure phase images, we mapped x to the 2π-phase domain using the exponential function $\exp(2\pi i x)$, keeping the magnitude as 1. On the other hand, we use the preprocessed Fashion-MNIST images $x \in [0, 1]$ as the magnitude part of the image $x_{\text{mag}} = x$.

During training, the model was optimized using the Adam optimizer, and $\beta_1 = 0.9, \beta_2 = 0.999$, weight decay 1×10^{-4}, with a total of 300,000 iterations. The learning rate follows a cosine annealing strategy, decaying gradually from 3×10^{-4} to 1×10^{-6}. In addition, our proposed method employs a progressive training strategy, starting with 128 × 128 small image patches and a batch size of 64, and then gradually increasing the image patch size and reducing the batch size to improve the model's adaptation to large-scale images. It is noteworthy that, apart from the input measurement data, no additional external training data is required for our proposed method.

B. Progressive training strategy

For a single-shot phase retrieval task, the proposed method typically needs to extract contextual information and local fine-grained features from inputs of different scales. However, due to the attention model's potential inability to effectively capture global image when processing small-sized image patches, the model's performance in recovering full-resolution images during testing may be poor. Moreover, training directly on large-sized inputs significantly increases computational costs and affects training efficiency. To fully leverage the potential of this architecture, we propose a progressive training strategy.

In the early stages of training, we select smaller image patches $P_1 \in R^{h_1 \times w_1 \times C}$ for training, quickly iterating to improve the network's ability to model shallow features (such as edges and textures). As the training progresses through stage t, the image patch size gradually increases to $P_t \in R^{h_t \times w_t \times C}$, where $h_t > h_{t-1}$ and $w_t > w_{t-1}$. In this process, the loss function for each stage is defined as:

$$\mathcal{L}_t = \frac{1}{N_t} \sum_{i=1}^{N_t} \left\| \hat{I}_i^{(t)} - I_i^{(t)} \right\|^2 + \lambda \cdot \mathcal{R}(W) \tag{32}$$

where $\hat{I}_i^{(t)}$ is the reconstructed image at stage t, $I_i^{(t)}$ is the reference image, $R(W)$ is the regularization term, and λ is the balancing parameter. After each image patch expansion, the core modules, including PAM, FRM, and refinement modules, adaptively optimize their feature representations. PAM captures a larger range of physical features, ensuring the consistency of physical constraints. FRM further refines local feature details, supplementing multi-scale information, and the refinement module enhances feature fusion quality through convolution and attention mechanisms, producing features with global consistency and enhanced local details.

As the image patch size increases, the computational cost for training also increases, so we dynamically adjust the batch size B_t to satisfy:

$$B_t \propto \frac{1}{h_t \cdot w_t} \omega \tag{33}$$

to ensure that the computational time for each optimization step remains stable while fully utilizing hardware resources.

IV. Ablation study

A. Ablation study of the LCT-FPB module

In order to analyze the impact of the LCT-FPB module on model performance, we designed a series of ablation experiments by removing only the LCT-FPB module while keeping other architecture and training parameters unchanged. This allowed us to quantify the contribution of this module to reconstruction quality, structural similarity, and computational complexity. The experiment mainly includes two settings: **Baseline** is the model without the LCT-FPB module, and **Full Model** is the complete architecture with the LCT-FPB module.

Both configurations use the same training scheme. We adopt a progressive learning strategy, initially training with small image blocks (128 × 128, batch size 64), and gradually increasing the image block size while reducing the batch size to improve the model's adaptability to large-scale data. The specific adjustment scheme is as follows: (160×160, 40), (192×192, 32), (256×256, 16), (320×320, 8), (384×384, 8). The loss function uses L1 loss to minimize reconstruction error, and data augmentation (horizontal/vertical flipping) is applied to enhance generalization ability.

To quantitatively analyze the contribution of the LCT-FPB module, we adopted several evaluation metrics: Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) for assessing reconstruction quality, Mean Absolute Error (MAE) for quantifying pixel-level errors, and Floating Point Operations (FLOPs), measured in GigaFLOPs, along with parameter count (M) for evaluating computational complexity.

Visual comparison of the contribution of FPB module results are provided in Figure 4. In terms of image reconstruction quality, the Baseline model without LCT-FPB module exhibits blurring in high-frequency detail regions and significant loss of edge structures. In contrast, the complete model (with LCT-FPB) effectively preserves texture details and reduces noise artifacts, producing reconstructed images that are visually closer to the original ones. This further validates the effectiveness of LCT-FPB in enhancing feature representation and improving reconstruction quality.

The ablation study of the LCT-FPB module is shown in Table I. The experimental results show that, without the LCT-FPB module, the model's PSNR is 32.50 dB, whereas the PSNR of the full model increases to 33.92 dB. This indicates that the LCT-FPB module plays a significant role in feature propagation and information enhancement. Similarly, the SSIM index increases from 0.921 to 0.938, suggesting that the LCT-FPB module has a positive effect on enhancing image structural details. Furthermore, the MAE decreases from 0.018 to 0.015, further indicating that the module effectively reduces pixel-level errors and improves reconstruction quality.

Although the LCT-FPB module improves the model's performance, the increase in computational cost is limited. In terms of computational complexity, FLOPs only increase from 85.7G to 88.1G, with an increase of about 2.8%. The parameter count also increases by only 1.2M, indicating that the LCT-FPB module improves performance while maintaining high computational efficiency. To further evaluate its computational overhead, we performed inference speed tests on an NVIDIA RTX 3060 GPU. The results show that the inference time for the Baseline model without LCT-FPB on 256×256 resolution images is 78.6 ms, while the inference time for the full model is 81.2 ms, with only a 3.3% increase. This demonstrates that the LCT-FPB module provides performance improvement with a minimal impact on computational overhead.

B. Contribution of PAM and FRM attention mechanisms

In order to investigate the role of different attention mechanisms in image reconstruction tasks, a series of ablation experiments are designed to systematically analyze the contributions of PAM and FRM to model performance. We constructed four experimental configurations:

- (1) **Baseline** (no attention mechanism), where all attention modules are removed, leaving only the backbone network;
- (2) **PAM-only**, where only PAM is used to evaluate its impact on global feature modeling capability;
- (3) **FRM-only**, where only FRM is used to analyze its role in local feature enhancement;
- (4) **Full Model** (PAM + FRM), which is the complete model containing both attention mechanisms.

The ablation study of PAM and FRM is illustrated in Table II. In the initial stage, 128×128 small image blocks are used with a batch size of 64, and the image block size is gradually increased to 384×384 while reducing the batch size to improve the model's adaptability to large-scale images.

Experimental results show that the Baseline model significantly lags behind models with attention mechanisms in

Table I ABLATION STUDY OF PAM AND FRM

Model	PSNR (dB) ↑	SSIM ↑	MAE ↓	FLOPs (G) ↓	Params (M) ↓
Baseline (w/o LCT-FPB)	32.50	0.921	0.018	85.7	24.5
Full Model (w/ LCT-FPB)	**33.92**	**0.938**	**0.015**	**88.1**	**25.7**

Table II ABLATION STUDY OF LCT-FPB MODULE

Model	PSNR (dB) ↑	SSIM ↑	MAE ↓	FLOPs (G) ↓	Params (M) ↓
Baseline (No Attention)	32.15	0.918	0.019	80.2	23.5
PAM-only	33.42	0.932	0.016	87.5	25.1
FRM-only	33.01	0.927	0.017	85.3	24.8
Full Model (PAM + FRM)	**34.05**	**0.940**	**0.015**	**90.1**	**26.3**

both PSNR and SSIM metrics. Among them, PAM excels in capturing global features, improving the PSNR to 33.42 dB, while FRM is more advantageous in local feature enhancement and detail recovery, reducing the MAE to 0.017. The full model with PAM and FRM performs the best across all metrics, with the PSNR reaching 34.05 dB and SSIM improving to 0.940, verifying the effectiveness of their synergistic effect.

Additionally, the computational complexity analysis shows that compared to the Baseline (80.2G FLOPs), the FLOPs of PAM-only and FRM-only increase to 87.5G and 85.3G, respectively, while the FLOPs of the full model reach 90.1G. Although the computational overhead increases slightly, the performance improvements justify its efficiency.

Figure 5 illustrates visual comparisons that highlight the strength of integrating PAM and FRM for enhanced performance. The baseline model shows a significant loss of high-frequency information, while the PAM-only version improves global consistency but still lacks detailed information. The FRM-only version presents richer local details but slightly weaker overall structure. In contrast, the combination of PAM and FRM performs effectively in both global and local feature extraction. It is evident that the baseline model suffers from blurring in high-frequency detail regions and considerable loss of edge structures. The PAM-only model enhances global consistency, yet still lacks optimization for local details. The FRM-only model demonstrates advantages in local feature enhancement but still shows issues with coherent global structure. The complete model (PAM + FRM) effectively preserves texture details and reduces noise artifacts, making the reconstructed image visually closer to the original. This further confirms the effectiveness of LCT-FPB in enhancing feature representation and improving reconstruction quality.

V. EXPERIMENTAL RESULTS

This paper compares the proposed method with both traditional and state-of-the-art deep learning-based phase retrieval methods, including HIO [7], a classical iterative optimization technique for obtaining the desired solution, and ResNet [26], a deep residual network that mitigates the vanishing gradient problem via skip connections, thereby improving the stability of phase retrieval. LenlessNet [27], a residual UNet network designed for lensless imaging, performs well in pure phase image recovery tasks. MCNN [28], a multi-scale CNN architecture, enhances phase retrieval performance by extracting features at different scales. PRCGAN [29], based on a Generative Adversarial Network (GAN) framework, optimizes the phase retrieval task through adversarial training strategies. A deep learning model NNPhase [30] recovers complex phase information from Fraunhofer diffraction measurements. LearnInitNet [31], a method based on learned initialization weights, optimizes the quality of the initial solution and accelerates phase retrieval convergence. HIO-Unet [32], a combination of the HIO iterative algorithm and Unet structure, enhances phase retrieval accuracy in complex scenarios through deep learning. CPR-FS [33], a phase retrieval algorithm based on conditional probability modeling, provides optimal recovery under specific LCT transform conditions. The results in the Table III demonstrate the performance of different reconstruction methods, where the best and second-best results are highlighted, showing the effectiveness of the proposed method across various LCT measurements on CELL 8 and Fashion-MNIST datasets, while the visualization results, including amplitude objects and phase results, can be found in Figure 6 and 7. The method proposed in this paper achieves optimal PSNR and SSIM results under different LCT transform measurement conditions through multi-scale feature learning and deep optimization strategies.

The results of all learning-based methods indicate that the reconstructed images appear slightly blurred compared to the ground truth. This can be attributed to the fact that, in the captured intensity images, high-frequency components have much lower magnitudes than low-frequency ones. Moreover, the presence of noise further obscures these high-frequency details, making them harder to recover. Therefore, it is necessary to further investigate the reconstructions obtained under certain sub-optimal LCT parameter settings.

In Table IV, we show a set of results on the CELL8 and Fashion-MNIST datasets under sub-optimal LCT parameters. It can be observed that while all methods experience significant performance degradation, the proposed method consistently achieves the best results, demonstrating its robustness and adaptability.

Based on the experimental results and image reconstruc-

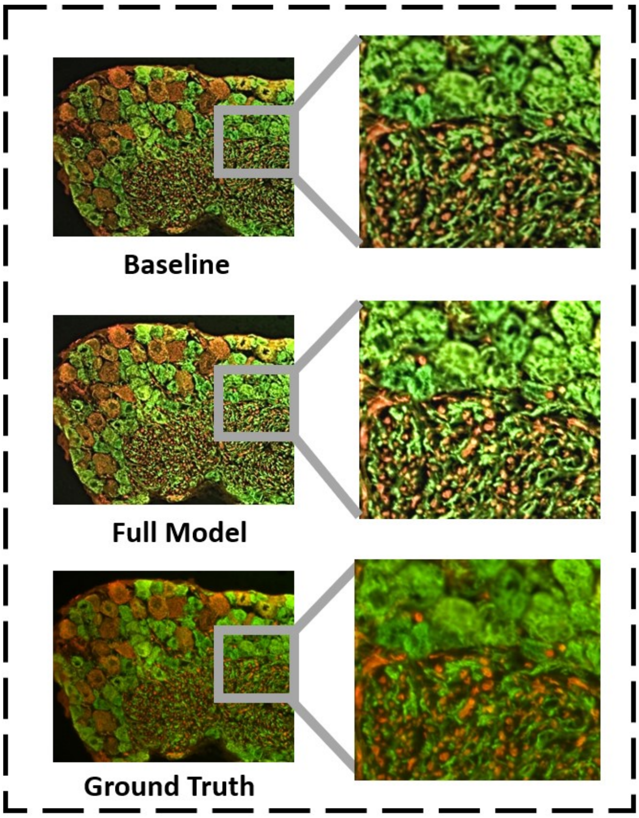

Fig. 4 Visual comparison of the contribution of FPB module.

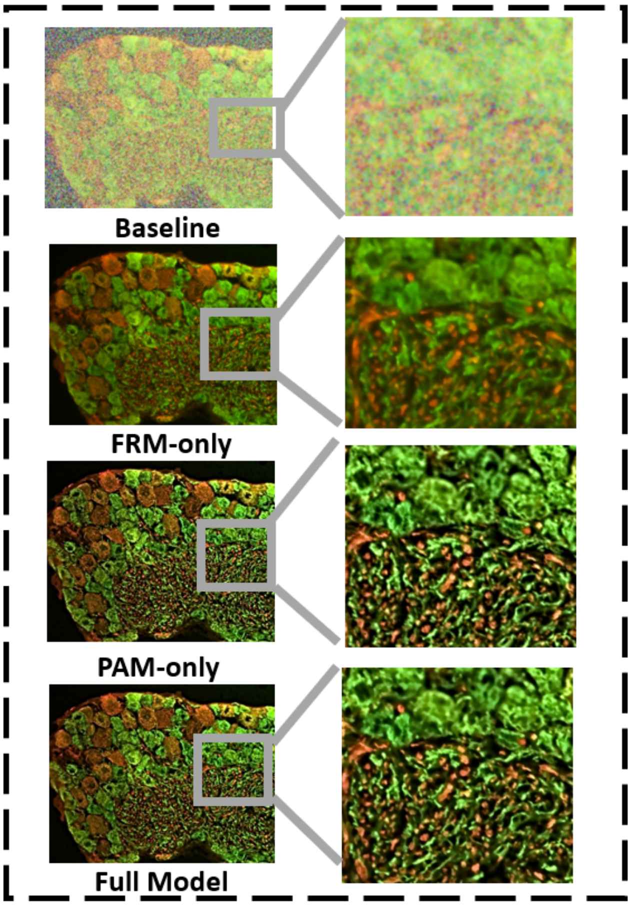

Fig. 5 Visual Comparison of the Contribution of PAM and FRM Attention Mechanisms.

Table III AVERAGE PSNR/SSIM PERFORMANCE COMPARISONS OF VARIOUS RECONSTRUCTION METHODS FOR Different LCT PARAMETERS ON CELL 8 AND FASHION-MNIST DATASETS, WHERE THE BEST RESULTS ARE HIGHLIGHTED IN BOLD AND THE SECOND-BEST RESULTS ARE UNDERLINED

datasets	method	type	$a=1, b=\pi/4,$ $c=0, d=1$	$a=1, b=3\pi/4,$ $c=0, d=1$	$a=\cos(\pi/3), b=\sin(\pi/3),$ $c=-\sin(\pi/3), d=\cos(\pi/3)$
CELL 8	HIO	Amplitude	24.31/0.82	25.14/0.84	26.32/0.86
	ResNet		30.45/0.89	31.78/0.91	32.56/0.92
	LenslessNet		31.12/0.91	32.45/0.92	34.01/0.94
	MCNN		30.87/0.90	32.01/0.91	33.12/0.93
	PRCGAN		32.78/0.93	33.89/0.94	35.12/0.95
	NNPhase		31.89/0.92	34.12/0.95	35.45/0.96
	LearnInitNet		30.98/0.91	32.87/0.92	33.98/0.94
	HIO-UNet		32.12/0.92	33.78/0.94	35.01/0.95
	CPR-FS		30.76/0.90	32.45/0.92	33.89/0.93
	Ours		**33.98/0.95**	**35.12/0.97**	**36.78/0.98**
	HIO	Phase	25.87/0.83	27.12/0.85	28.45/0.87
	ResNet		31.23/0.90	32.76/0.92	34.02/0.94
	LenslessNet		32.78/0.93	34.45/0.94	35.87/0.96
	MCNN		31.98/0.92	33.21/0.93	34.32/0.94
	PRCGAN		33.45/0.94	34.89/0.95	36.12/0.97
	NNPhase		32.34/0.93	34.12/0.94	35.45/0.96
	LearnInitNet		31.87/0.92	33.45/0.93	34.76/0.94
	HIO-UNet		33.12/0.94	34.65/0.95	35.78/0.96
	CPR-FS		31.98/0.91	33.54/0.93	34.98/0.94
	Ours		**34.76/0.96**	**36.12/0.97**	**37.65/0.98**
Fashion-MNIST	HIO	Amplitude	23.14/0.81	24.32/0.83	25.87/0.85
	ResNet		29.12/0.88	30.45/0.90	31.78/0.91
	LenslessNet		30.78/0.90	32.12/0.92	33.45/0.93
	MCNN		29.87/0.89	31.12/0.91	32.34/0.92
	PRCGAN		31.98/0.92	33.45/0.94	34.78/0.95
	NNPhase		30.98/0.91	33.12/0.94	35.01/0.96
	LearnInitNet		29.98/0.90	31.87/0.92	33.54/0.94
	HIO-UNet		32.65/0.94	33.78/0.95	35.12/0.96
	CPR-FS		30.21/0.89	31.54/0.91	33.12/0.93
	Ours		**33.87/0.95**	**35.21/0.97**	**36.78/0.98**
	HIO	Phase	30.12/0.87	32.01/0.89	33.12/0.92
	ResNet		32.34/0.91	34.12/0.93	35.21/0.94
	LenslessNet		33.12/0.93	35.12/0.95	36.12/0.96
	MCNN		32.98/0.92	34.87/0.94	35.78/0.95
	PRCGAN		34.21/0.95	36.21/0.96	36.98/0.97
	NNPhase		33.89/0.94	35.21/0.95	36.12/0.96
	LearnInitNet		32.78/0.92	34.98/0.94	35.98/0.95
	HIO-UNet		35.01/0.97	37.12/0.98	37.98/0.99
	CPR-FS		32.56/0.91	34.54/0.93	36.78/0.96
	Ours		**35.87/0.98**	**37.45/0.98**	**38.45/0.99**

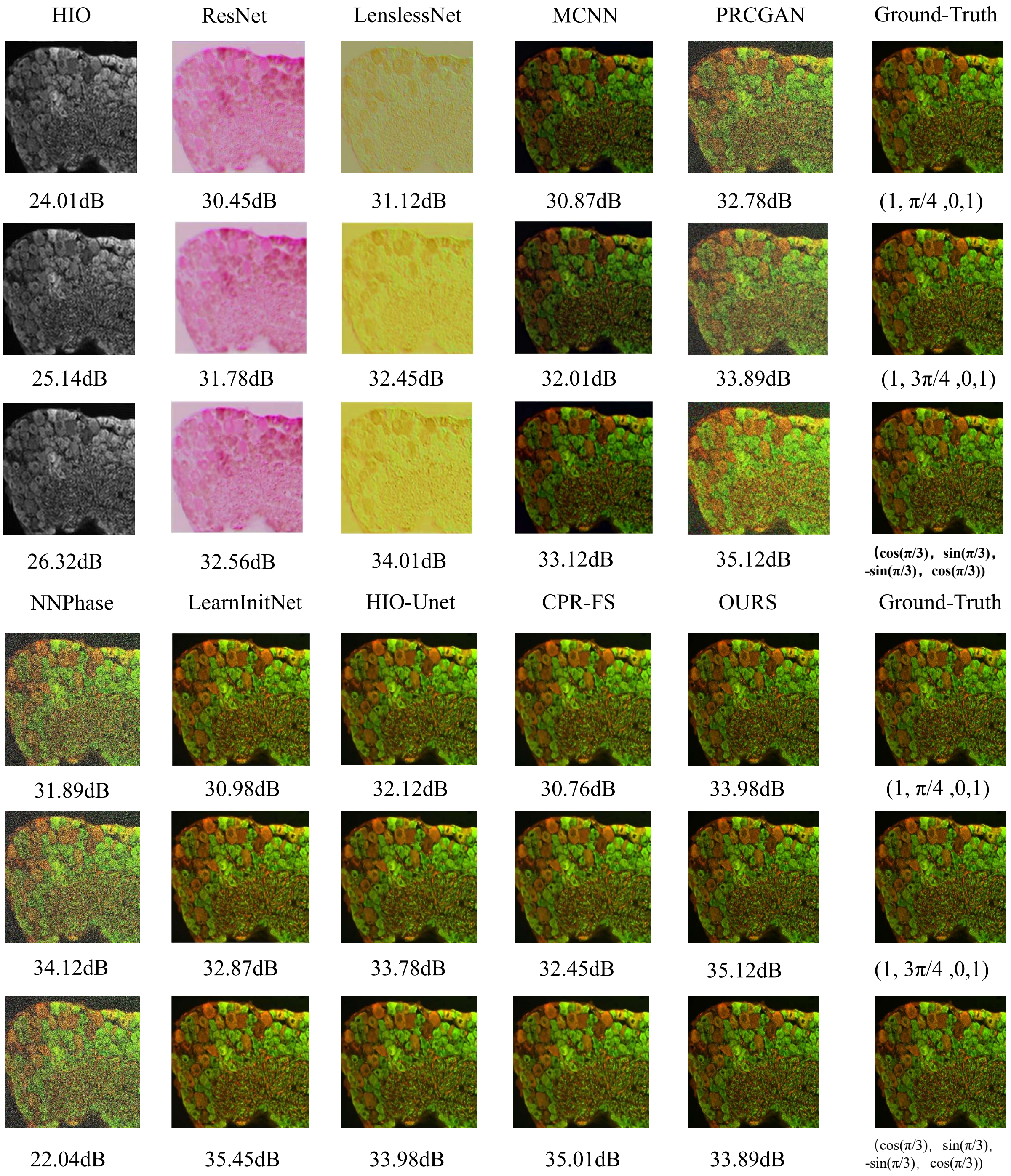

Fig. 6 Reconstruction results (Amplitude objects) of nine PR methods for different LCT measurements, from top to bottom.

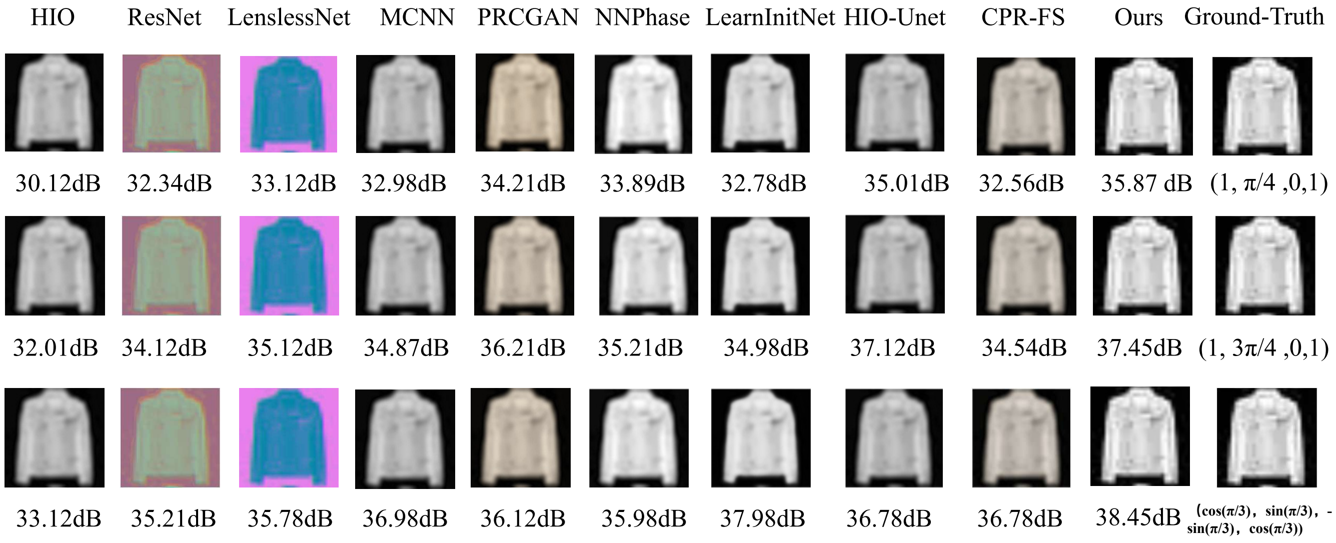

Fig. 7 Reconstruction results (phase objects) of nine PR methods for different LCT measurements, from top to bottom.

Table IV AVERAGE PSNR/SSIM PERFORMANCE COMPARISONS OF VARIOUS RECONSTRUCTION METHODS FOR INFERIOR LCT MEASUREMENTS ON CELL 8 AND FASHION-MNIST DATASETS, WHERE THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

datasets	method	type	$a=1, b=0, c=1, d=1$
CELL8	HIO	Amplitude	8.27 / 0.12
	ResNet		10.47 / 0.13
	LenslessNet		11.32 / 0.24
	MCNN		10.16 / 0.19
	PRCGAN		11.77 / 0.27
	NNPhase		9.86 / 0.23
	LearnInitNet		11.23 / 0.21
	HIO-Unet		10.97 / 0.18
	CPR-FS		12.33 / 0.24
	Ours		**13.41 / 0.25**
	HIO	Phase	9.37 / 0.13
	ResNet		10.32 / 0.20
	LenslessNet		13.66 / 0.26
	MCNN		11.79 / 0.20
	PRCGAN		11.72 / 0.21
	NNPhase		10.32 / 0.25
	LearnInitNet		11.12 / 0.21
	HIO-Unet		11.32 / 0.26
	CPR-FS		12.56 / 0.28
	Ours		**15.70 / 0.31**
Fashion-MNIST	HIO	Amplitude	11.24 / 0.08
	ResNet		9.37 / 0.16
	LenslessNet		11.15 / 0.24
	MCNN		12.64 / 0.23
	PRCGAN		10.35 / 0.26
	NNPhase		9.63 / 0.23
	LearnInitNet		12.57 / 0.26
	HIO-Unet		12.39 / 0.29
	CPR-FS		11.94 / 0.25
	Ours		**13.91 / 0.29**
	HIO	Phase	13.47 / 0.24
	ResNet		11.84 / 0.19
	LenslessNet		12.45 / 0.28
	MCNN		12.36 / 0.27
	PRCGAN		12.57 / 0.21
	NNPhase		11.59 / 0.19
	LearnInitNet		12.97 / 0.22
	HIO-Unet		13.86 / 0.24
	CPR-FS		12.73 / 0.25
	Ours		**14.31 / 0.27**

tion quality analysis, the proposed method demonstrates significant advantages under different LCT measurement conditions. Compared with traditional methods and recent deep learning-based phase retrieval methods, the proposed method achieves the best PSNR and SSIM values under all tested LCT transform conditions. In particular, under different LCT transforms on the CELL 8 and Fashion-MNIST datasets, the proposed method effectively recovers high-frequency details and structures of the image, surpassing existing phase retrieval methods.

VI. Conclusion

In this paper, we propose a method of single-shot phase retrieval from linear canonical transform via deep physical neural network, called SPRLCT-PNN. The proposed method leverages the advantages of LCT while incorporating upsampling and downsampling techniques for improvement. The experimental results on the CELL 8 and Fashion-MNIST datasets demonstrate that the proposed method effectively recovers high-frequency details and structures of the image, outperforming existing CNN methods. Future work is needed to explore the potential of the proposed method in preprocessing and its performance across different datasets.

References

[1] Bertero, Mario, Patrizia Boccacci, and Christine De Mol. Introduction to inverse problems in imaging. CRC press, 2021

[2] Jonathan Dong, Lorenzo Valzania, Antoine Maillard, Thanh-an Pham and Sylvain Gigan, Michael Unser, "Phase Retrieval: From Computational Imaging to Machine Learning: A tutorial, IEEE Signal Processing Magazine," vol. 40, no. 1, pp45-57, 2023

[3] Miao Jianwei, David Sayre, and H. N. Chapman. "Phase retrieval from the magnitude of the Fourier transforms of nonperiodic objects, Journal of the Optical Society of America A." vol. 15.no. 6, pp1662-1669, 1998

[4] Guoan Zheng, Roarke Horstmeyer and Changhuei Yang, "Wide-field, high-resolution Fourier ptychographic microscopy," Nature Photonics, vol. 7, no. 9, pp739-745, 2013

[5] Kaiqiang Wang, Li Song, Chutian Wang, Zhenbo Ren, Guangyuan Zhao, Jianglei Di, George Barbastathi, Renjie Zhou, Jianlin Zhao and Edmund Y. Lam, "On the use of deep learning for phase recovery," Light: Science & Applications, vol. 13, no. 1, pp4, 2024

[6] Guozhen Yang, Bizhen Dong, Benyuan Gu, Jieyao Zhuang, and Okan K. Ersoy, "Gerchberg–Saxton and Yang–Gu algorithms for phase retrieval in a nonunitary transform system: a comparison," Applied Optics. vol. 33, pp209-218, 1994

[7] Heinz H. Bauschke, Patrick L. Combettes, and D. Russell Luke, "Hybrid projection–reflection method for phase retrieval," Journal of the Optical Society of America A, vol. 20, pp1025-1034, 2003

[8] Emmanuel J. Candès, Xiaodong Li and Mahdi Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," IEEE Transactions on Information Theory, vol. 61, no. 4, pp1985-2007, 2015

[9] Ryoichi Horisaki, Yusuke Ogura, Masahiko Aino, and Jun Tanida, "Single-shot phase imaging with a coded aperture," Optics Letters, vol. 39, no. 22, pp6466-6469, 2014

[10] Emmanuel J. Candès, Xiaodong Li and Mahdi Soltanolkotabi, "Phase retrieval from coded diffraction patterns," Applied and Computational Harmonic Analysis, vol. 39, no. 2, pp277-299, 2015

[11] Lan Li, Yufeng Ren, and Kai Yu. Sparse Phase Retrieval via Reweighted Thresholded Wirtinger Flow. IAENG International Journal of Applied Mathematics, vol.54, no.4, pp683-689, 2024

[12] Huibin Chang, Yifei Lou, Yuping Duan, and Stefano Marchesini, "Total variation-based phase retrieval for Poisson noise removal," SIAM Journal on Imaging Sciences, vol. 11, no. 1, pp24-55, 2018

[13] Lan Li, Shasha Pu, Mingli Jing, Yulong Mao, Xiaoya Liu and Qiyu Sun, "Phase retrieval based on the distributed conditional generative adversarial network," Journal of the Optical Society of America A, vol. 41, no. 9, pp1702-1712, 2024

[14] Yixiao Yang, Ran Tao, Kaixuan Wei and Jun Shi, "Single-shot phase retrieval from a fractional Fourier transform perspective," IEEE Transactions on Signal Processing, vol. 72, pp3303-3317, 2024

[15] Alice Lucas, Michael Iliadis, Rafael Molina and Aggelos K. Katsaggelos,"Using deep neural networks for inverse problems in imaging: Beyond analytical methods," IEEE Signal Processing Magazine, vol. 35, no. 1, pp20-36, 2018

[16] Qiuliang Ye, LiWen Wang and Daniel P K Lun "Towards practical single-shot phase retrieval with physics-driven deep neural network," Optics Express, vol. 31, no. 22, pp35982-35999, 2023

[17] Qiuliang Ye, LiWen Wang and Daniel P K Lun "SiSPRNet: end-to-end learning for single-shot phase retrieval," Optics Express, vol. 30, no. 18, pp31937-31958, 2022

[18] Tobias Uelwer, Tobias Hoffmann and Stefan Harmeling,"Non-iterative phase retrieval with cascaded neural networks," in Artificial Neural Networks and Machine Learning–ICANN, Bratislava, Slovakia, pp295-306, 2021

[19] Unnikrishnan Gopinathan, John J. Healy, Damien P. Kelly and John T. Sheridan, "Deterministic phase retrieval using the LCT," Linear Canonical Transforms: Theory and Applications, pp331-346, 2016

[20] M. Günhan Ertosun, Haluk Atlı, Haldun M. Ozaktas and Billur Barshan, "Complex signal recovery from two fractional Fourier transform intensities: order and noise dependence," Optics Communications, vol. 244, no. 1-6, pp61-70, 2005

[21] L. J. Allen, M. P. Oxley, "Phase retrieval from series of images obtained by defocus variation," Optics Communications, vol. 199, no. 1-4, pp65-75, 2001

[22] Giancarlo Pedrini, Wolfgang Osten, and Yan Zhang, "Wave-front reconstruction from a sequence of interferograms recorded at different planes," Optics Letters, vol. 30, no. 8, pp833-835, 2005

[23] Alon Greenbaum, Aydogan Ozcan, "Maskless imaging of dense samples using pixel super-resolution based multi-height lensfree on-chip microscopy," Optics Express, vol. 20, no. 3, pp3129-3143, 2012

[24] S. Ito, "CIL:10932, Felis catus, epithelial cell, enterocyte," CIL Dataset, 2011. [Online]. Available: https://doi.org/10.7295/W9CIL10932.

[25] Han Xiao, Kashif Rasul and Roland Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," arXiv preprint arXiv:1708.07747, 2017

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun, "Deep residual learning for image recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778

[27] Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis, "Lensless computational imaging through deep learning," Optica, vol. 4, pp1117, 2017

[28] Wonseok Jeon, Wooyoung Jeong, Kyungchan Son, and Hyunseok Yang "Speckle noise reduction for digital holographic images using multi-scale convolutional neural networks," Optics Letters, vol. 43, pp4240-4243, 2018

[29] Tobias Uelwer, Alexander Oberstraß and Stefan Harmeling, "Phase retrieval using conditional generative adversarial networks," 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, pp731-738, 2021

[30] Longlong Wu, Pavol Juhas, Shinjae Yoo and Ian Robinson, "Complex imaging of phase domains by deep neural networks," IUCrJ, vol. 8, no. 1, pp12-21, 2021

[31] David Morales, Andrés Jerez, and Henry Arguello, "Learning spectral initialization for phase retrieval via deep neural networks," Applied Optics, vol. 61, ppF25-F33, 2022

[32] Çağatay Işıl, Figen S. Oktem, and Aykut Koç, "Deep iterative reconstruction for phase retrieval," Applied Optics, vol. 58, pp5422-5431, 2019

[33] T. Uelwer, T. Hoffmann and S. Harmeling, "Non-iterative phase retrieval with cascaded neural networks," Artificial Neural Networks and Machine Learning–ICANN, vol. 30, pp295-306, 2021