Fast mode decomposition for few-mode fiber based on lightweight neural network

Jiajia Zhao (赵佳佳)[1], Guohui Chen (陈国辉)[1], Xuan Bi (毕 轩)[1], Wangyang Cai (蔡汪洋)[1], Lei Yue (岳 磊)[1], and Ming Tang (唐 明)[2*]

[1] School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China

[2] Wuhan National Laboratory for Optoelectronics (WNLO) and National Engineering Laboratory for Next Generation Internet Access System, School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan 430074, China

*Corresponding author: tangming@mail.hust.edu.cn

In this paper, we present a fast mode decomposition method for few-mode fibers, utilizing a lightweight neural network called MobileNetV3-Light. This method can quickly and accurately predict the amplitude and phase information of different modes, enabling us to fully characterize the optical field without the need for expensive experimental equipment. We train the MobileNetV3-Light using simulated near-field optical field maps, and evaluate its performance using both simulated and reconstructed near-field optical field maps. To validate the effectiveness of this method, we conduct mode decomposition experiments on a few-mode fiber supporting six linear polarization (LP) modes (LP01, LP11e, LP11o, LP21e, LP21o, LP02). The results demonstrate a remarkable average correlation of 0.9995 between our simulated and reconstructed near-field light-field maps. And the mode decomposition speed is about 6 ms per frame, indicating its powerful real-time processing capability. In addition, the proposed network model is compact, with a size of only 6.5 MB, making it well suited for deployment on portable mobile devices.

Keywords: deep learning; lightweight neural network; few-mode fiber; mode decomposition.
DOI: 10.3788/COL202422.020604

1. Introduction

In recent years, few-mode fibers (FMFs) have received increasing attention due to their potential applications in high-power fiber lasers[1], space-division multiplexing transmission[2,3], and imaging. Moreover, FMFs are regarded as ideal platforms for the study of spatiotemporal mode-locking mechanisms[4–6] and Kurr nonlinear beam cleaning[7]. However, the mode-coupling phenomenon in FMFs is inevitable, which significantly impacts their performance. Therefore, it is crucial to understand the mode properties of FMFs to suppress higher-order mode generation or optimize their design. The mode-decomposition (MD) technique is a fundamental measurement method that allows for the decomposition of the amplitude and phase information of each eigenmode in FMFs. It plays a critical role in studying mode properties and transmission characteristics in FMFs. Currently, MD techniques are commonly used for measuring fiber mode transfer matrices[8], implementing adaptive mode control[9], analyzing fiber mode coupling[10], studying fiber bending losses[11], and measuring beam quality[12].

Early MD methods were primarily based on experimental techniques[13–17], where the complete distribution of the optical field was directly measured by using sophisticated experimental devices. However, these methods suffer from high equipment costs, high accuracy requirements, complex experimental procedures, heavy workloads, and vulnerability to environmental influences. Subsequently, numerically based MD techniques[18–22] were proposed, which effectively reduce the cost and equipment requirements and only require simple experiments. However, these methods are susceptible to problems such as initial value sensitivity, convergence to local minima, high computational effort, and long convergence time during a large number of iterations. To solve these problems, some non-iterative numerical decomposition methods, such as fractional-level Fourier systems[23] and matrix-inversion methods[24], have emerged, which avoid the above problems and show excellent performance.

Recently, neural network-based MD methods have shown feasibility and are emerging as a significant research direction. An *et al.* achieved the first high-precision real-time MD of five modes using VGG-16 convolutional neural networks in 2019[25]. Subsequently, Fan *et al.* improved the convolutional neural network in 2020 by adding additional loss functions associated with near-field and far-field spot maps to achieve

high-precision MD in the case of six-mode superposition[26]. Zhu *et al.* successfully achieved high-precision MD with six modes using a ResNet-18 convolutional neural network in 2021[27]. Rothe *et al.* utilized Dense-Net convolutional neural network with up to 121 layers to achieve high-precision MD with eight modes superimposed[28]. Also artificially designed neural network-based methods[29] and multitask deep-learning methods[30] to achieve MD have been proposed successively and have shown better performance. However, all of the above methods use traditional convolutional neural networks for MD, which have problems such as long training time, high computer equipment requirements, and excessive computing resources consumption. In addition, the large number of parameters in traditional convolutional neural network models makes them unsuitable for deployment on portable devices, like popular Android smartphones. To address this challenge, lightweight convolutional neural networks have been proposed and shown promising results in image classification[31]. Among them, Google proposed the lightweight neural network model MobileNetV3 in 2019[32]. MobileNetV3 employs the neural architecture search (NAS) parametric search method and redesigns the time-consuming layer structure and activation function. This greatly reduced the number of training parameters while maintaining high accuracy, leading to significantly reduced training time for the entire network. Overall, MobileNetV3 makes the whole image classification network more lightweight and efficient.

In this paper, we propose a fast MD method based on the improvement of MobileNetV3. The proposed algorithm uses depth-separable convolution instead of conventional convolution, redesigns the activation function, and reduces the repetitive layer structure without any pretraining process. The method can quickly and accurately predict the mode weights of the eigenmodes and the phase differences between the fundamental and higher-order modes. Simulation test results show that the average mode weight error of modes is less than 0.56%, the average relative phase error is less than 0.85%, and the average correlation between simulated and reconstructed near-field optical field maps is as high as 0.9995 under the condition of FMFs supporting six LP modes (LP01, LP11e, LP11o, LP21e, LP21o, LP02). The MD of this method speed is about 6 ms per frame, the real-time processing is very strong, and the network model size is merely 6.5 MB, which has the advantages of fast decomposition speed, low experimental equipment requirement, and easy deployment compared with other deep-learning methods. Most importantly, this lightweight model facilitates the reduction of the need for storage devices and computational resources and is easy to deploy on portable devices such as cell phones and sensors.

2. Implementation Method

The propagation field within the FMFs can be expressed as a linear superposition of several eigenmodes, as shown in Eq. (1)[33],

$$U(x,y) = \sum_{n=1}^{n_{\max}} A_n e^{i\theta_n} \psi_n(x,y), \quad \sum_{n=1}^{n_{\max}} A_n^2 = 1, \quad \theta_n \in [-\pi, \pi],$$

(1)

where $\psi_n(x,y)$ denotes the nth eigenmode of the propagation field in the fiber, A_n^2 is the nth mode weight, and θ_n is the relative phase of the mode. $n_{\max}$ denotes the maximum number of modes supported in the fiber. Since the refractive index difference between the core and cladding of an FMF is small enough to qualify for the weak-conductance approximation, in this paper we use the LP mode to uniformly describe each eigenmode[33]. The MD technique is a measurement technique to obtain A_n^2 and θ_n from near-field optical field intensity images.

Figure 1 illustrates the entire MD process using MobileNetV3_Light. First, the eigenmode is calculated based on the known fiber structure parameters, and the mode weights along with relative phase coefficients are randomly generated. Then, the near-field optical field image is generated by eigenmode superposition theory simulation. It is worth noting that, although the phase sign cannot be determined from the near-field optical field image alone, it is feasible to use only the near-field optical field image for MD in fiber laser studies, where in most cases only the mode ratios of the individual eigenmodes are of interest[25]. During the training phase of the neural network, we take the near-field light-field image as input and process the generated mode weights and relative phase coefficients as label vectors. The label vector consists of n mode weights and $n-1$ relative phases, set into a set of $2n-1$ column vectors, where n denotes the number of supported eigenmodes in the FMF. The n mode weights are denoted as $\{A_i^2 \mid i=1,2,\ldots,n\}$, A_i^2 is randomly generated between 0 and 1; and the $n-1$ relative phases between higher-order modes and fundamental modes are denoted as $\{\theta_i | i=1,2,\ldots,n-1\}$. θ_i is randomly generated between $-\pi$ and π. Since the relative phase is ambiguous[25], meaning that a near-field optical field image may have multiple phase labels, directly using the relative phase as the label vector might cause the neural network to be unable to converge. To address this issue, we use the cosine of the relative phase instead of the true value of the label vector and linearly scale the range of the cosine from $[-1, 1]$ to the range of $[0, 1]$. The preprocessed near-field light-field images are then passed through the MobileNetV3_Light neural network model, which includes convolutional layers, pooling layers, fully connected layers, and activation functions, ultimately producing the prediction vector as the output. Finally, the mean square
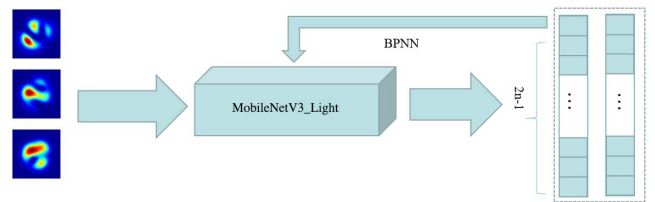

Fig. 1. Pattern decomposition based on MobileNetV3_Light neural network.

error MSE between the prediction vector and the label vector is used as the network loss function[25], as shown in Eq. (2),

$$\text{MSE} = \frac{1}{K} \cdot \sum_{i=1}^{K} \sum_{j=1}^{2n-1} (x_o^{(i)}[j] - x_l^{(i)}[j])^2. \tag{2}$$

Here, K represents the number of training samples, x_o represents the prediction vector, and x_l represents the label vector. The parameters of the neural network are updated using a back-propagation neural network (BPNN) until the network converges, resulting in a network model capable of directly predicting the mode information. Subsequently, with the trained MobileNetV3_Light model, the mode weights and phase information of the FMF can be directly predicted from the simulated near-field optical field intensity image, and the complete mode coefficients can be obtained by selecting the appropriate phase combinations without any iterative process.

3. Neural Network Model Design

Convolutional neural network models face two major challenges in applications: one is the storage issue, as hundreds of network layers contain a large number of parameters, leading to high storage requirements for the device. The other is the speed issue, where prediction is usually required to be done within milliseconds in order to meet the practical standards of mobile applications. To address these performance issues, model compression is a common solution. It involves retraining an already trained model to reduce the number of parameters in the network, thus solving the storage problem. Unlike dealing with already trained models, lightweight models are designed using a more efficient "network computation method" (mainly used in convolutional methods) to reduce network parameters without affecting performance. Representative examples in this regard include Squeeze-Net, Mobile-Net, Shuffle-Net, and Xception.

In this paper, MobileNetV3 is used as the initial model, and MobileNetV3_Light is obtained by fine-tuning it. The performance improvement of this model can be mainly attributed to the use of depth-separable convolution instead of the traditional convolutional computation. As described in the literature[34], the depth-separable convolution decomposes the traditional convolution into two parts: a depth-wise convolution and a 1×1 convolution. As shown in Fig. 2, Fig. 2(a) illustrates the conventional convolution, while Figs. 2(b) and 2(c) represent the depth convolution and the 1×1 convolution, respectively.

The computation effort of the conventional convolution is $HWNK^2M$, whereas the total computation of the depth-separable convolution can be calculated by Eq. (3)[34],
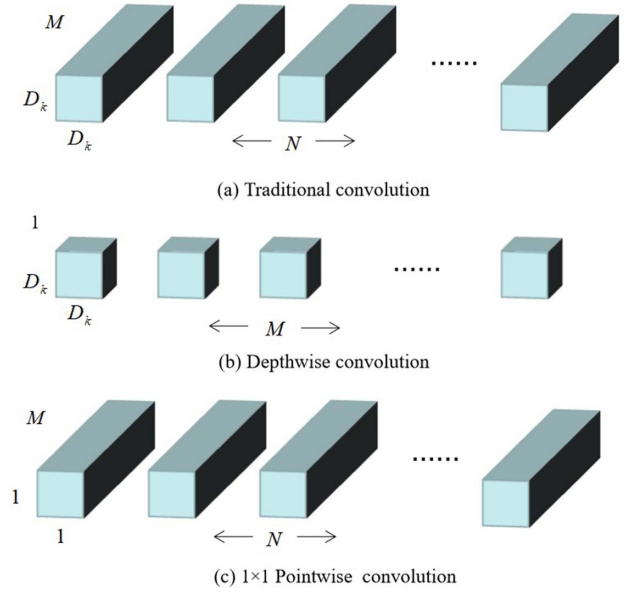

Fig. 2. Traditional convolution and depth-separable convolution.

$$HWNK^2(\text{Depth Wise}) + HWNM(\text{Point Wise})$$

$$= HWN(K^2 + M),$$

$$\frac{\text{Depth Wise} + \text{Point Wise}}{\text{Conv}} = \frac{1}{M} + \frac{1}{K^2}. \tag{3}$$

It can be seen that the number of parameters in the lightweight neural network model is 2.5 million and the number of parameters in the traditional network model is 21.875 million. Deep-separable convolution requires only 1/8 to 1/9 of the computational effort of traditional convolution. It achieves this by decomposing the traditional convolutional factorization into a deep convolution and a point-by-point convolution, which significantly reduces the computational effort of the neural network model.

In addition, MobileNetV3 further reduces the computational cost of the model by employing the hard-sigmoid function (instead of the sigmoid function) and by simplifying the repetitive folding operation. Experiments in the literature[35] demonstrate that hard-sigmoid functions play almost the same role as sigmoid functions in mobile devices, but are typically less computationally intensive. The activation functions used in this paper include Hard-Swish and ReLU functions, whose expressions are shown in Eqs. (4) and (5), respectively,

$$\text{Hard-Swish}(x) = x\frac{\text{ReLU6}(x+3)}{6}, \tag{4}$$

$$\text{ReLU}(x) = \max(x, 0). \tag{5}$$

The $\text{ReLU6}(x) = \min(\max(x, 0), 6)$ and ReLU6 functions output zero for negative values and values greater than six, and output itself for other values; the ReLU function outputs zero for negative values and outputs itself for other values. The overall architecture of the MobileNetV3_Light network

designed in this paper is shown in Fig. 3, which consists of three modules.

The first module consists of a traditional convolutional layer, followed by the use of a Hard-Swish activation function. The second module comprises nine MobileNetV3 blocks, each structured as depicted in Fig. 4[32]. Within each MobileNetV3 block, the input feature matrix is first up-dimensioned by a 1×1 convolutional layer, followed by a 3×3 depth-wise convolutional layer. Each convolutional operation is then followed by a batch normalization (BN) layer and an activation function (NL). Additionally, a self-attention mechanism (SE) module is applied, where the average pooling of the number of channels of the feature matrix is obtained, and an output vector is derived through two fully connected layers. Here the first fully connected layer comprises a number of nodes equal to one-quarter of the number of feature matrix channels, followed by a ReLU activation function. The second fully connected layer consists of a number of nodes equal to the number of feature matrix channels and employs a hard-sigmoid activation function.

Finally, a 1×1 convolutional layer is utilized to reduce the dimensionality, resulting in the final output feature matrix. It is important to note that the shortcut operation is executed only when the step of depth-wise convolution is equal to 1 and the number of channels of the input feature matrix matches the number of channels of the output feature matrix. In this case, the input and output feature matrices are directly summed in the same dimension. The third module consists of a 7×7 average pooling layer and three 1×1 convolution layers.

The detailed parameters of the whole network model structure are shown in Table 1. The "Input" column denotes the size of the input feature matrix. The "Operator" column represents various operations, where "Conv2d" indicates the convolutional
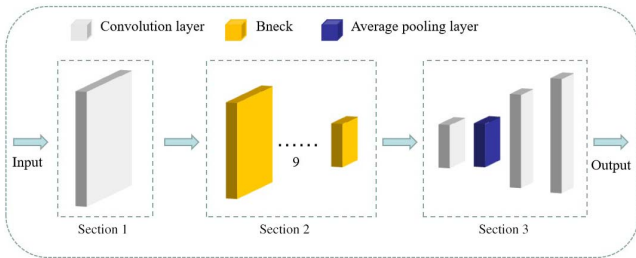
Table 1. Detailed Parameter Settings of MobileNetV3_Light Network Model Structure.

Input	Operator	Exp size	#out	SE	NL	s
$224^2 \times 3$	Conv2d	–	8	–	HS	2
$112^2 \times 8$	Bneck, 3×3	16	16	✓	RE	2
$56^2 \times 16$	Bneck, 3×3	72	24	–	RE	2
$28^2 \times 24$	Bneck, 3×3	88	24	–	RE	1
$28^2 \times 24$	Bneck, 5×5	96	40	✓	HS	2
$14^2 \times 40$	Bneck, 5×5	240	40	✓	HS	1
$14^2 \times 40$	Bneck, 5×5	120	48	✓	HS	1
$14^2 \times 48$	Bneck, 5×5	144	48	✓	HS	1
$14^2 \times 112$	Bneck, 5×5	288	96	✓	HS	2
$7^2 \times 96$	Bneck, 5×5	576	96	✓	HS	1
$7^2 \times 96$	Conv2d, 1×1	–	576	✓	HS	1
$7^2 \times 576$	AvgPool, 7×7	–	–	–	–	1
$1^2 \times 576$	Conv2d, 1×1, NBN	–	1024	–	HS	1
$1^2 \times 1024$	Conv2d, 1×1, NBN	–	K	–	–	1

layer, "Bneck, 3×3" indicates the size of depth-wise convolution in MobileNetV3 block as 3×3, "AvgPool, 7×7" means the average pooling layer size of 7×7, and "NBN" means no BN layer is used. The "exp size" column represents the size of the first up-dimensioned convolutional layer. The "#out" column shows the size of the output feature matrix. The "SE" column signifies whether the self-attention mechanism is employed, with "✓" indicating its usage. The "NL" column represents different activation functions, where "HS" is the Hard-Swish activation function, and "RE" is the ReLU activation function. In addition, the symbol "s" is the step size of the convolution or pooling operation, while "K" represents the number of output vectors. For instance, in the case of N modes, the decomposition corresponds to a number of output vectors K equal to $2N - 1$. Compared with the initial model, the number of convolution kernels in the first convolution layer is reduced from 16 to 8, and the number of repeated MobileNetV3 blocks is reduced from 11 to 9, which further improves the efficiency of our model.

Fig. 3. MobileNetV3_Light network structure.
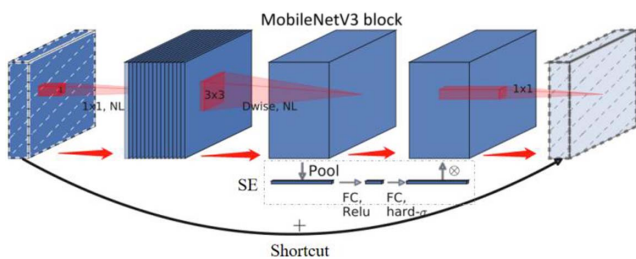
Fig. 4. MobileNetV3 block network structure diagram.

4. Experimental Results and Discussion

All experiments reported in this paper were run on a desktop computer with an AMD R7 5800X CPU and an NVIDIA GeForce RTX 3070 GPU. First, a data set comprising 100,000 near-field light-field maps with a resolution of 224×224 is randomly generated through MATLAB simulation. The data set is then divided into training and validation sets in an 8:2 ratio.

For training, 150 epochs are utilized, with a batch size of 64 for key training parameters. The hyperparameter learning rate is set to 0.01 for the initial 20 epochs and reduced to 0.0001 for the subsequent epochs, with the aim of accelerating the training process. To prevent overfitting, validation checks are used as the termination condition in this work, and the number of validation checks is set to six. Using the validation check as the termination condition means that during the training process, the computation will stop if the error curve of the validation set samples no longer decreases in six consecutive iterations. The network decreases rapidly to a relatively good level after the first 15 training periods and stops training after 100 training periods. The total training time is about 267.5 min. Finally, we combine simulated and reconstructed near-field light-field maps to test the performance of the entire network.

The flow of the whole test is shown in Fig. 5. MobileNetV3_Light can directly predict the weight coefficients of the obtained patterns $\overline{A_1^2}$, $\{\overline{A_i^2} | i = 1, 2, \ldots, n\}$, and the cosine coefficients of the relative phases L_i, $\{L_i | i = 1, 2, \ldots, n-1\}$, where it is also necessary to find all possible matching phase combinations $\overline{\theta_i}$, $\{\overline{\theta_i} | i = 1, 2, \ldots, n-1\}$ by the processed cosine coefficients and finally to evaluate the correlation between the simulated and reconstructed near-field light-field maps by the evaluation function $f(k)$. The expression is shown in Eq. (6)[24],

$$f(k) = \frac{\sum_{x,y}(I_0(x,y) - \bar{I}_0)(I_k(x,y) - \bar{I}_k)}{\sqrt{\sum_{x,y}(I_0(x,y) - \bar{I}_0)^2 \sum_{x,y}(I_k(x,y) - \bar{I}_k)^2}}, \quad (6)$$

where I_0 is the simulated near-field light-field intensity distribution, I_k is the reconstructed near-field light-field intensity distribution, and $\bar{I}_0$ and $\bar{I}_k$ are the average values of I_0 and I_k, respectively. A larger $f(k)$ indicates a higher correlation between the simulated and reconstructed near-field intensity distributions, and the closer the evaluation function is to 1, the closer the simulated and reconstructed near-field light-field images are. In this experiment, ΔP and $\Delta \theta$ defined in Eqs. (7) and (8) are used to numerically describe the mode weights and relative phase errors of the simulated and reconstructed near-field optical field intensities[29],

$$\Delta p_i = |A_i^2 - \overline{A_i^2}|, \quad i = 1, 2, \ldots, n, \quad (7)$$

$$\Delta \theta_i = ||\theta_i| - |\overline{\theta_i}||, \quad i = 1, 2, \ldots, n-1, \quad (8)$$
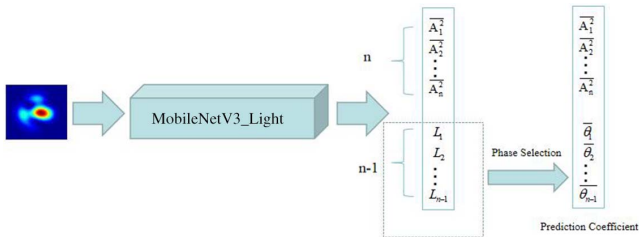
where A_i^2 and θ_i represent the simulated near-field optical field intensity mode weights and the relative phases of the higher-order modes to the fundamental mode, and $\overline{A_i^2}$ and $\overline{\theta_i}$ represent the predicted mode weights and the relative phases of the higher-order modes to the fundamental mode. The closer the error is to 0, the more accurate the predicted mode weights and relative phases are. Finally, our method is evaluated visually by comparing the simulated and reconstructed near-field optical field images and showing the residuals between them. In addition, we evaluate the time spent for the whole decomposition process and the size of the number of parameters of the network model; the shorter the prediction time, the higher the performance of the MD. The number of parameters affects the memory footprint on the one hand and the size of the package directly on the other. If the package is too large, it will make the model deployment impossible.

We have evaluated the performance simulation at 1073 nm using a step refractive index fiber with a core radius of 11.8 μm and NA of 0.064 as an example. The normalized frequency of this fiber is about 4.43, so it can support six modes, which can be sequentially arranged as LP01, LP11e, LP11o, LP21e, LP21o, and LP02 modes. Due to the simplicity of the modes[30], there are three possible cases of modes propagated by this fiber, which are the first three, five, and six modes. As the number of modes increases, the combination of eigenmodes becomes more complex and the number of near-field optical field images with different mode coefficients increases[25–30]. Therefore, under the condition of supporting six modes with FMF, we generated 1000 random near-field light-field images for testing the MobileNetV3_Light network model for different training periods.

We then calculated the average correlation between the simulated and reconstructed near-field light-field intensities for the test samples after each period of training; the results are shown in Fig. 6. It can be found that the correlation only increases rapidly to above 0.9910 within the first 15 periods, and then starts to
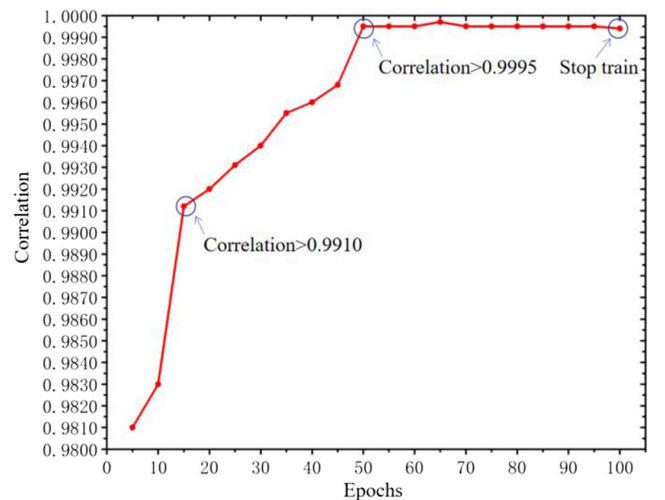

Fig. 5. Test flow chart.

Fig. 6. Average correlations across training periods for the six model cases.

converge by the 50th period. We chose to stop training at 100 training periods, when the correlation approaches 0.9995. In addition, we tested the network after 100 training periods, comparing the pattern weights and the relative phase errors. The average errors of the individual pattern weights and relative phases are shown in Tables 2 and 3. It can be seen that the average mode weight error is less than 0.56%, and the average relative phase error is less than 0.85% for all six modes. Compared with the literature[25–29], the scheme proposed in this paper can achieve similar decomposition accuracy. It can be concluded that the trained MobileNetV3_Light can learn the relationship between pattern coefficients and near-field light-field intensity images. It should be emphasized that the mode weight errors and relative phase errors described in this experiment are the average of 1000 samples.

To visually evaluate the method, simulated near-field light-field maps, reconstructed near-field light-field maps, residual maps, and their correlation are collected for multiple sets of samples in Fig. 7. It can be found that the reconstructed near-field light-field maps and the simulated near-field light-field maps are highly similar with very small residuals, which further confirms the accuracy and effectiveness of the method described in this work.

In order to measure the size of the memory footprint of the devices required for the MD technique and the speed of decomposition, we computed the time taken by the network to perform each stage of MD using GPUs and the size of the number of parameters modeled by the network, under the condition of 1000 test legends. The time spent in each stage is reported in

detail in Table 4. In Table 4, T_1 indicates the training time of the MobileNetV3_Light network, T_2 indicates the time of image preprocessing, T_3 indicates the calculation time of pattern weights and relative phases, and T_4 indicates the time of picking the most suitable phase combination. From Table 4, we can find that it takes 6.27 s to complete MD for 1000 samples using the trained MobileNetV3_Light neural network, of which 2.41 s is required for image preprocessing, and 3.86 s is required for modal weight and relative phase calculation. Note that a single sample takes only 6.27 ms to perform MD. We can see that a near-field optical field image takes only about 6 ms to complete MD, which is much lower than that of the method in the literature[25] that uses the VGG-16 model for MD, which demonstrates that the method has high performance. In addition, such high decomposition efficiency makes the method also capable of potential real-time MD. In Table 5, the sizes of some of the proposed neural network models are compared, where "Parameters" is the number of parameters. It can be seen from Table 5 that the number of parameters of our proposed MobileNetV3_Light network model is only 2.5 million and the size of the network model is 6.5 MB. Compared to the neural network model size of some of the proposed pattern decompositions[36,37], the decomposition scheme proposed in this paper has obvious advantages. Currently proposed neural network methods for MD generally have the problem of large model size, which we avoid better by designing a lightweight neural network. This gives our model a greater advantage on a portable mobile device.

To access the feasibility of our method in a case with more modes, we extended our investigations to train network models for 8 and 10 modes. As the number of modes increases, the combination of eigenmodes becomes more complex and the number of similar near-field light-field images with different mode coefficients increases. Therefore, to ensure the accuracy of decomposition, we optimized the network by stacking the number of MobileNetV3 blocks in the MobileNetV3_Light network in order to improve the ability of network learning. Furthermore, we augmented the training data set size and improved the resolution of the near-field light-field images, which benefited the fitting process.

In our work, the number of MobileNetV3 blocks corresponding to 8 and 10 patterns was increased to 11 and 15, respectively. The training data set size was extended to 150,000 and 200,000 images, respectively, with the mean resolution of images increased to 256×256. Consequently, the training time also increased to 314.6 and 426.8 min, respectively. The number of parameters in the model correspondingly increased to 2.75 million and 3.22 million. It can be found that even under the condition of an increased number of patterns, our method

Table 2. Average Error of the Six Model Weights.

	$\overline{\Delta p_1}$	$\overline{\Delta p_2}$	$\overline{\Delta p_3}$	$\overline{\Delta p_4}$	$\overline{\Delta p_5}$	$\overline{\Delta p_6}$
Average weights error	0.47%	0.48%	0.42%	0.48%	0.53%	0.55%

Table 3. Average Error of the Relative Phase of the Six Modes.

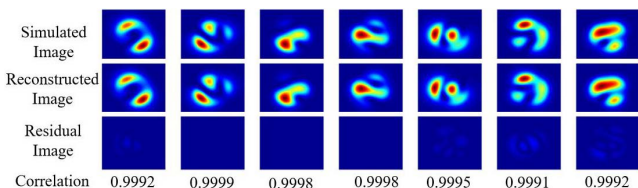|	$\overline{\Delta \theta_1}$	$\overline{\Delta \theta_2}$	$\overline{\Delta \theta_3}$	$\overline{\Delta \theta_4}$	$\overline{\Delta \theta_5}$
Average weights error	0.47%	0.48%	0.42%	0.48%	0.53%

Fig. 7. Simulated near-field light-field map, reconstructed near-field light-field map, residual images, and their correlation.

Table 4. Time Spent in Different Phases of Testing.

	T_1	T_2	T_3	T_4
Predicting model weight and phase	267.5 min	2.41 s	3.86 s	36.24 s

Table 5. Parameter Size of Different Neural Network Models.

	MobileNetV3_Light	MoblieNetV2	Xception	Resnet50	VGG-16
Parameters	2.5×10^6	3.4×10^6	22.85×10^6	25.56×10^6	138.36×10^6
Mode size	6.5 MB	14.2 MB	88 MB	98 MB	528 MB

Fig. 8. Relation between the mode number and correlation.

Fig. 9. Simulated and reconstructed images under the influence of different intensities of noise and their correlation.
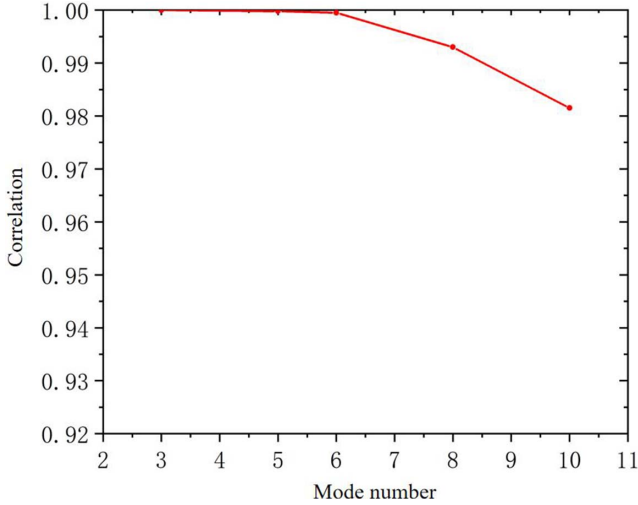
still shows great advantages compared to models such as VGG-16.

Figure 8 depicts the relationship between the number of supported modes and the correlation. It can be found that the correlation decreases as the number of supported modes increases, reaching 0.98 when there are 10 modes. The decomposition scheme based on MobileNetV3_Light does not show an advantage in supporting more modes. The reason may be that the expansion of modes will lead to an increase in the number of similar near-field light-field maps with different mode coefficients, which will introduce ambiguities. A promising way to reduce this error is to introduce far-field light-field images. Far-field light-field images corresponding to similar near-field light-field images with different mode coefficients exhibit significant differences. Therefore, by combining near-field and far-field light-field images, MobileNetV3_Light is expected to accurately predict the mode coefficients with almost no blurring. Figure 8 reveals that our proposed scheme is feasible when the number of modes is less than or equal to six.

Finally, we investigate the robustness of MobileNetV3_Light by adding Gaussian noise to the near-field light-field map. For the noise generation, the simulated near-field light-field map is used as the root, which is achieved by multiplying each pixel of the simulated near-field light-field map by a noise function $p(\sigma)$, as shown in Eq. (9)[25],
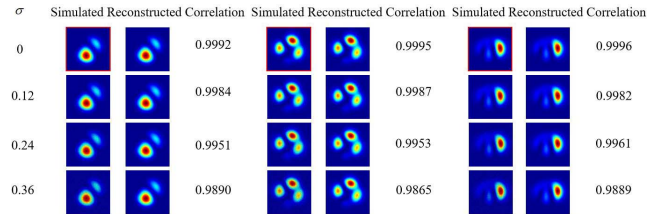
$$p(\sigma) = 1 + N(0,1) \cdot \sigma, \qquad (9)$$

where $N(0,1)$ represents the standard normal distribution and σ is the noise intensity. Here different sizes of noise are achieved by varying the σ. Taking the six modes as an example, we test 1000 samples at different noise intensities and input them into the trained MobileNetV3_Light. Figure 9 lists the simulated near-field light-field maps and the corresponding reconstructed near-field light-field maps for three different values of σ and their correlations. It can be found that even if σ reaches 0.24, the correlation still exceeds 0.995, which indicates the high noise immunity of our method.

5. Summary

We propose a complete MD technique based on lightweight neural networks that offer high accuracy, high performance, and low experimental equipment requirements. The proposed algorithm uses depth-separable convolution instead of conventional convolution without any pretraining, which both reduces the network model size and improves the speed of decomposition while maintaining high accuracy in MD conditions. The results show that for the FMF supporting six LP modes (LP01, LP11e, LP11o, LP21e, LP21o, LP02), our trained neural network achieves an average mode weight error of less than 0.56% and the average relative phase error of less than 0.85%. The MD speed reaches about 6 ms per frame, and the model size of the network is only about 6.5 MB, making it feasible for real-time MD on portable mobile devices. Additionally, our proposed method demonstrates robustness, even in the presence of high noise intensity, up to 0.36.

Acknowledgements

References

1. X. Chen, T. Yao, and L. Huang, "Functional fibers and functional fiber-based components for high-power lasers," Adv. Fiber Mater. **5**, 59 (2023).
2. D. J. Richardson, J. M. Fini, and L. E. Nelson, "Space-division multiplexing in optical fibres," Nat. Photonics **7**, 354 (2013).
3. J. Du, W. Shen, J. Liu, et al., "Mode division multiplexing: from photonic integration to optical fiber transmission [Invited]," Chin. Opt. Lett. **19**, 091301 (2021).
4. L. G. Wright, D. N. Christodoulides, and F. W. Wise, "Spatiotemporal mode-locking in multimode fiber lasers," Science **358**, 94 (2017).
5. S. Chen, W. Hu, Y. Xu, et al., "Mode-locked pulse generation from an all-FMF ring laser cavity," Chin. Opt. Lett. **17**, 121405 (2019).
6. S. Bai, Y. Lu, and Z. Zhang, "Mode field switching in narrow linewidth mode-locked fiber laser," Chin. Opt. Lett. **20**, 020602 (2022).
7. K. Krupa, A. Tonello, B. M. Shalaby, et al., "Spatial beam self-cleaning in multimode fibres," Nat. Photonics **11**, 237 (2017).
8. J. Carpenter, B. J. Eggleton, and J. Schröder, "110 × 110 optical mode transfer matrix inversion," Opt. Express **22**, 96 (2014).
9. L. Huang, J. Leng, P. Zhou, et al., "Adaptive mode control of a few-mode fiber by real-time mode decomposition," Opt. Express **23**, 28082 (2015).
10. D. Flamm, K.-C. Hou, P. Gelszinnis, et al., "Modal characterization of fiber-to-fiber coupling processes," Opt. Lett. **38**, 2128 (2013).
11. C. Schulze, A. Lorenz, D. Flamm, et al., "Mode resolved bend loss in few-mode optical fibers," Opt. Express **21**, 3170 (2013).
12. D. Flamm, C. Schulze, R. Brüning, et al., "Fast M^2 measurement for fiber beams based on modal analysis," Appl. Opt. **51**, 987 (2012).
13. J. Demas and S. Ramachandran, "Sub-second mode measurement of fibers using C^2 imaging," Opt. Express **22**, 23043 (2014).
14. J. W. Nicholson, A. D. Yablon, S. Ramachandran, et al., "Spatially and spectrally resolved imaging of modal content in large-mode-area fibers," Opt. Express **16**, 7233 (2008).
15. N. Andermahr, T. Theeg, and C. Fallnich, "Novel approach for polarization-sensitive measurements of transverse modes in few-mode optical fibers," Appl. Phys. B **91**, 353 (2008).
16. Y. Z. Ma, Y. Sych, G. Onishchukov, et al., "Fiber-modes and fiber-anisotropy characterization using low-coherence interferometry," Appl. Phys. B **96**, 345 (2009).
17. T. Kaiser, D. Flamm, S. Schröter, et al., "Complete modal decomposition for optical fibers using CGH-based correlation filters," Opt. Express **17**, 9347 (2009).
18. R. Brüning, P. Gelszinnis, C. Schulze, et al., "Comparative analysis of numerical methods for the mode analysis of laser beams," Appl. Opt. **52**, 7769 (2013).
19. L. Huang, S. Guo, J. Leng, et al., "Real-time mode decomposition for few-mode fiber based on numerical method," Opt. Express **23**, 4620 (2015).
20. F. Stutzki, H.-J. Otto, F. Jansen, et al., "High-speed modal decomposition of mode instabilities in high-power fiber lasers," Opt. Lett. **36**, 4572 (2011).
21. H. Lü, P. Zhou, X. Wang, et al., "Fast and accurate modal decomposition of multimode fiber based on stochastic parallel gradient descent algorithm," Appl. Opt. **52**, 2905 (2013).
22. L. Li, J. Leng, P. Zhou, et al., "Multimode fiber modal decomposition based on hybrid genetic global optimization algorithm," Opt. Express **25**, 19680 (2017).
23. W. Yan, X. Xu, and J. Wang, "Modal decomposition for few mode fibers using the fractional Fourier system," Opt. Express **27**, 13871 (2019).
24. E. S. Manuylovich, V. V. Dvoyrin, and S. K. Turitsyn, "Fast mode decomposition in few-mode fibers," Nat. Commun. **11**, 5507 (2020).
25. Y. An, L. Huang, J. Li, et al., "Learning to decompose the modes in few-mode fibers with deep convolutional neural network," Opt. Express **27**, 10127 (2019).
26. X. Fan, F. Ren, Y. Xie, et al., "Mitigating ambiguity by deep-learning-based modal decomposition method," Opt. Commun. **471**, 125845 (2020).
27. Z.-H. Zhu, Y.-Y. Xiao, and R.-M. Yao, "CNN-based few-mode fiber modal decomposition method using digital holography," Appl. Opt. **60**, 7400 (2021).
28. S. Rothe, Q. Zhang, N. Koukourakis, et al., "Intensity-only mode decomposition on multimode fibers using a densely connected convolutional network," J. Lightwave Technol. **39**, 1672 (2021).
29. H. Gao, Z. Chen, Y.-X. Zhang, et al., "Rapid mode decomposition of few-mode fiber by artificial neural network," J. Lightwave Technol. **39**, 6294 (2021).
30. Z. Tian, L. Pei, J. Wang, et al., "High-precision mode decomposition for few-mode fibers based on multi-task deep learning," J. Lightwave Technol. **40**, 7711 (2022).
31. F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2017), p. 1800.
32. A. Howard, M. Sandler, B. Chen, et al., "Searching for MobileNetV3," in *IEEE/CVF International Conference on Computer Vision (ICCV)* (IEEE, 2019), p. 1314.
33. A. W. Snyder and J. D. Love, *Optical Waveguide Theory* (Chapman and Hall, 1983).
34. A. G. Howard, M. Zhu, B. Chen, et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," arXiv:1704.04861 (2017).
35. P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," arXiv:1710.05941 (2017).
36. B. Yan, J. Zhang, M. Wang, et al., "Degenerated mode decomposition with convolutional neural network for few-mode fibers," Opt. Laser Technol. **154**, 108287 (2022).
37. Y. An, L. Huang, J. Li, et al., "Deep learning-based real-time mode decomposition for multimode fibers," IEEE J. Sel. Top. Quantum Electron. **26**, 4400806 (2020).