nature photonics

Supplementary information

https://doi.org/10.1038/s41566-025-01799-7

Direct tensor processing with coherent light

In the format provided by the authors and unedited

Supplementary information for Direct tensor processing with coherent light

Yufeng Zhang,[1,2,3†] Xiaobing Liu,[4,5†] Chenguang Yang,[1] Jinlong Xiang,[1,3] Hao Yan,[1,6] Tianjiao Fu,[4,5] Kaizhi Wang,[1,6*] Yikai Su,[1,3*] Zhipei Sun,[2*] Xuhan Guo[1,3*]

[1.] School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

[2.] Department of Electronics and Nanoengineering, Aalto University, Espoo 02150, Finland.

[3.] State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China.

[4.] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China.

[5.] University of Chinese Academy of Sciences, Beijing 100049, China.

[6.] Yiwu Zhiyuan Research Center of Electronic Technology, Jinhua 322000, China.

Corresponding authors: Kaizhi Wang (kz_wang@sjtu.edu.cn), Yikai Su (yikaisu@sjtu.edu.cn), Zhipei Sun (zhipei.sun@aalto.fi), Xuhan Guo (guoxuhan@sjtu.edu.cn)

[†]These authors contributed equally to this work.

This Supplementary Information contains **5 Supplementary Notes**, **18 Supplementary Figures**, **4 Supplementary Tables**, and **35 References**.

Supplementary Notes

Supplementary Note 1 - Detail principle of parallelized optical matrix-matrix multiplication

Here, we provide a detailed mathematical formulation based on ideal Huygens-Fresnel diffraction, and assuming the Fourier transform achieves ideal amplitude summation based on Stanford methods[1]. As shown in Fig. S1, the matrix A is modulated onto the amplitude of planar collimated light and imaged to matrix A*, the equiphasic plane remains flat, and the wavefront only undergoes amplitude changes. For the linear phase matrix P, it is composed of multiple linear phase vectors with different gradients. The linear phase rate increases from top to bottom: $[K(N), \dots, K(2), K(1)]$, so the matrix $A^* e^{j2\pi P}$ can be written as:

$$\begin{pmatrix} a(N,M)e^{j2\pi K(N)M} & a(N,M-1)e^{j2\pi K(N)(M-1)} & \cdots & a(N,1)e^{j2\pi K(N)1} \\ \vdots & \vdots & \ddots & \vdots \\ a(2,M)e^{j2\pi K(2)M} & a(2,M-1)e^{j2\pi K(2)(M-1)} & \cdots & a(2,1)e^{j2\pi K(2)1} \\ a(1,M)e^{j2\pi K(1)M} & a(1,M-1)e^{j2\pi K(1)(M-1)} & \cdots & a(1,1)e^{j2\pi K(1)1} \end{pmatrix}$$

Where the j is the imaginary number, and according to Huygens-Fresnel theory, the optical field distribution $U(x,y)$ after diffraction of distance d can be expressed as:

$$U(x,y) = \frac{1}{j\lambda d} e^{j\frac{2\pi}{\lambda}d} e^{j\frac{\pi}{\lambda d}(x^2+y^2)} \mathcal{F}[U(x_0,y_0) e^{j\frac{\pi}{\lambda d}(x_0^2+y_0^2)}]_{f_x=\frac{x}{\lambda d}, f_y=\frac{y}{\lambda d}}.$$

Where, $U(x_0,y_0)$ is the optical field distribution before diffraction, $\mathcal{F}$ represents the Fourier transform (FT), f_x, f_y are the frequency domain values after FT. The relationship between $U(x_0,y_0)$ and $U(x,y)$ at the front and back focal planes of a lens with focal length d can be further expressed as:

$$U(x,y) = \frac{1}{j\lambda d} \mathcal{F}[U(x_0,y_0)]_{f_x=\frac{x}{\lambda d}, f_y=\frac{y}{\lambda d}}.$$

Due to the linear characteristics of the FT, the diffraction generated by the entire modulator surface can be divided into the superposition of the diffraction effects of each pixel. So, for the y-direction, the N rows of matrix A correspond to N coherent point light sources with different intensities, evenly distributed, and at equal intervals. These sources are fanned out (by FT) onto the surface of matrix B^T, where their fields are superimposed. In other words, the optical field distribution of each row after FT is the superposition of

all previous rows, but they come from different angular spectra. Assuming perfect imaging in x-direction, the amplitude (matrix value) and linear phase relationship along the x-direction can still be maintained. Therefore, the matrix formed by the combined action of x-direction imaging and y-direction FT can be approximated written as:

$$\begin{aligned} U(i,m) &= a(1,m)e^{j2\pi K(N)m}e^{j2\pi L(1,i)} + a(2,m)e^{j2\pi K(N-1)m}e^{j2\pi L(2,i)} + \cdots + a(N,m)e^{j2\pi K(1)m}e^{j2\pi L(N,i)} \\ &= \sum_{n=1}^{N} a(n,m)e^{j2\pi K(N+1-n)m}e^{j2\pi L(n,i)}, \end{aligned}$$

which is the sum of all rows of complex matrix $A^{*}e^{j2\pi P}$, and the $e^{j2\pi L(n,i)}$ represents a fixed phase caused by the different position of n^{th} row along the y-direction of matrix A and i^{th} row of matrix B^{T} (space-shifting property). And the amplitude of this superimposed optical field is modulated by matrix B^{T}:

$$U(i,m) = b^{T}(i,m)\sum_{n=1}^{N} a(n,m)e^{j2\pi K(N+1-n)m}e^{j2\pi L(n,i)},$$

If summation (FT) along the x-direction, there will be N spatial frequency points ($[K(N),\ldots,K(2),K(1)]$ to $[f_N,\ldots,f_2,f_1]$), and the amplitude of each point is the approximate result of the dot product of each row of A and this row of B^{T} (frequency-shifting property). Then if imaging along the y-direction (omitted flipping), other rows of B^{T} will be operated in the same way, the matrix-matrix multiplication (MMM) will be finished:

$$V(i,f_n) = e^{j2\pi L(n,i)}\sum_{m=1}^{M} b^{T}(i,m)a(n,m).$$

Where $V(i,f_n)$ is the complex amplitude distribution at spatial position i along the y-direction and spatial spectrum f_n along the x-direction, represents the dot product between n^{th} row of matrix A and i^{th} column of matrix B. Hence, the joint multiplexing of angular spectrum in the y-direction and combined with phase modulation in the x-direction, complement each other ingeniously.

Supplementary Note 2 - Experimental details

2.1 Experimental process

As shown in Fig. S2, the normalized values of the matrix (ranging from 0 to 1) are squared and quantized to a range of 0 to 255 and loaded onto the ASLMs (realizing intensity modulation). Simultaneously, different linear phase values, spanning from 0 to 2π, are quantized to the range 0 to 255 (8-bit mode) and

loaded onto different rows of the PSLM. To achieve optimal experimental results, we fill 1000 pixels of the ASLM by repeating a point along the row direction. The number of repetitions varies depending on the size of the matrix. So, one point in an [N, 50] matrix will be repeated 20 times (1000 / 20). To increase light intensity, each row is replicated 4 times, meaning that every 4 rows on the SLM represent one row of the matrix. According to the Huygens-Fresnel principle, the spatial extent l_{lr} of the optical field in the row direction of the resulting matrix is determined by the frequency range of the linear phase, given by: $l_{lr} = \frac{f\lambda}{\Delta_s}(f_{max} - f_{min})$, and $f_{max} - f_{min} \leq \frac{1}{2}$ (If it is acceptable for the resulting optical field to be "cut off" by the unmodulated light at the center, $f_{max} - f_{min} \leq 1$). Where, λ = 532 nm is the wavelength of the CW laser, f = 200 mm is the focal length of cylindrical lens 6, Δ_s = 8 μm is the pixel size. For the row gap of matrix A on ASLM1, which does not affect the final result, we set it to 400 / N to prevent potential inaccuracies in the 4f system from impacting other rows. And the row gap of matrix B^T on ASLM2 is also 400 / N, so the total results gap $l_{ud} = 400\Delta_s$. In our experiments, we used a variety of different l_{lr} and l_{ud} parameters for matrices with different sizes to make the visualization of the resulting optical field better, but in the ONN and large sample error analysis, these parameters are fixed. If POMMM is applied to other systems, the SLMs and detectors can be replaced with high-speed analog devices, and the lens (or phase mask) and optical path parameters can be adjusted according to the pixel size.

2.2 Installation details

There are two points that need special attention during the installation process of experimental prototypes: 1. the pixel-by-pixel alignment of three SLMs, and 2. the focal length selection of cylindrical lenses.

The validation experiment in this paper cascaded three SLMs (amplitude-phase-amplitude), and to obtain more accurate results, it is necessary to align the three SLMs pixel by pixel. Some researchers[2-4] have proposed effective methods for aligning the optical path in ONN or optical computing systems, but none of them are accurate to the pixel level. Here, we present an ingenious pixel-level alignment method that can be applied to the alignment of ASLM and PSLM, reducing the difficulty of SLM alignment in various free-space-based optical systems. As shown in the above left panel of Fig. S3, first build an optical 4f system and place the ASLM at the object plane position and the PSLM at the image plane position, that is, the

imaging results of the ASLM are located on the surface of the PSLM. We first align the pixels of ASLM and PSLM along the row dimension, set the center column and edge columns on both sides of ASLM to "1" respectively, and set the remaining positions to "0" (Fig. S3a). It means that after imaging with the 4f system, three vertical lines will be formed, with a width equal to the pixel size of ASLM. Then set the corresponding columns on the PSLM as superposed by a quadratic phase $e^{-j\frac{\pi}{\lambda d}(y_0^2)}$ and a linear phase $e^{j2\pi K y_0}$ (Fig. S3b). It is worth noting that the K (gradients of linear phases) in these three columns are distinct, denoted as $-\frac{1}{4\Delta_s}$, 0, $\frac{1}{4\Delta_s}$, respectively. And the column dimension optical field distribution $U(y)$ after diffraction of distance d can be expressed as:

$$U(y)=\frac{1}{j\lambda d}e^{j\frac{2\pi}{\lambda}d}e^{j\frac{\pi}{\lambda d}(y^2)}\mathcal{F}\left(e^{j2\pi K y_0}\right)_{f_y=\frac{y}{\lambda d}}.$$

So the optical field in the column dimension forms three evenly spaced bright dots with a separation distance of $\frac{\lambda d}{4\Delta_s}$ (frequency-shifting property). Obviously, for the row dimension, the light is equal to pass through a slit and spread out. Therefore, if ASLM and PSLM pixels are fully aligned, placing a detector at a distance of d from the PSLM will detect three bright straight lines along the row dimension, and their interval along the column dimension is $\frac{\lambda d}{4\Delta_s}$ (Fig. S3d). If not aligned, the image of ASLM will continue to diffract freely and the optical field will diverge (Fig. S3c). Similarly, use the same method to align the row dimension, as shown in the right panel of Fig. S3.

While ensuring paraxial diffraction conditions, the focal length selection of the cylindrical lens group (CL1, CL2, and CL3) between PSLM and ASLM2 depends on the pixel size and count of PSLM and ASLM2. As mentioned above, the distance from PSLM to ASLM2 is $4f$, where f is the focal length of the CL1 and 2, and the focal length of CL3 is $2f$. Due to the pixel size Δ_s, and is repeated to $4\Delta_s$, the zero-order optical field range along the column dimension is $l=\frac{f\lambda}{2\Delta_s}$. For the total length in column dimension of the ASLM2 surface is $400\Delta_s$, the l should be greater than $400\Delta_s$ to cover all the pixels:

$$\frac{f\lambda}{2\Delta_s}\geq 400\Delta_s \rightarrow f\geq\frac{800{\Delta_s}^2}{\lambda}.$$

Based on the selected SLM and laser parameters (Δ_s = 8 μm, λ = 532 nm), we choose f = 100 mm. From the perspective of FT analysis and based on our extensive testing, we recommend maximizing the focal length of each lens whenever possible, particularly when using large-pixel diffractive devices with weaker diffraction capabilities. This approach offers two key advantages: first, it reduces distance-related errors and alleviates the difficulty of alignment and assembly; second, it ensures that results in the column direction are sufficiently separated, thereby minimizing the risk of aliasing.

2.3 Result extraction details

Due to the pixel size of existing SLMs cannot be ignored (usually at the micron level), and there are intervals between pixels, i.e., discrete distributions. Therefore, the diffraction should be the result of the discrete space Fourier transform under the envelope of a sinc function ($\frac{\sin(x)}{x}$), rather than the completely standard FT[5]. As depicted in Fig. S4a, the results should be normalized and divided by the normalized matrix of the envelope (use the POMMM results of two identity matrices as the reference matrix) to obtain accurate outcomes in both simulation and experimental situations.

And in an experimental situation, since the fill factor of the SLMs is not 100%, a portion of the light remains unmodulated. Assuming the fill factor is Q ($Q \leq 1$), the light passing through the SLMs is a mixture of two states: the proportion of modulation part is Q, and the unmodulated part is 1 - Q. As illustrated in Fig. S4b, for a system comprising n cascaded SLMs, the components of its optical field resemble a binary tree, where only $\frac{1}{2^n}$ of the components represent the ideal modulation results, while the remainder constitute noise. These noise components can be categorized into three types (Fig. S4c). The first type, which includes 4 components, is never modulated by the PSLM. This category of noise remains centered in the optical field and does not interfere with the result matrix (see the bottom panel of Fig. S2) thus it can be naturally filtered out. The other two types of noise (referred to as "line noise" and "point noise" depending on whether they are modulated by ASLM2) can be pre-characterized and subtracted during post-processing. And other direct optical methods can be employed, such as using amplitude-phase modulators in place of ASLM1 and PSLM, or incorporating a spatial zero-frequency filter within the 4f system. These approaches can effectively reduce the "noise binary tree" by one layer, thereby minimizing the need for later adjustments to some

extent. Additionally, the modulation curve of the SLMs is not strictly linear, exhibiting some degree of error. Therefore, we conducted measurements of the SLM's modulation curve and applied preprocessing to the input by conducting optical path measurements 256 times (0 ~ 255) using the "laser - SLM - qCMOS" setup. For each measurement, a pure value image was loaded onto the SLM, and the average light intensity detected by the qCMOS was recorded as the result. Furthermore, the laser after collimation and beam expansion is not completely uniform, and the quality of optical components cannot be accurately evaluated, so additional calibration tests are required. Since there is a BS (Beam Splitter) in front of each SLM, the optical field on one side can be calibrated to the result of the previous SLM. Another optional method is to optimize and fit the optical field error weights in front of each level of SLM through reasonable modeling combined with random large sample testing. Regardless of the method, pre-compensation of input and output is necessary. Of course, if it is possible to use a customized optical system (such as lenses, BS, and wave plates), this process will become simple.

Another detail worth discussing is the relationship between amplitude and intensity. While the Stanford method[1] assumes that both the input and output correspond to a mapping from numerical values to optical intensity, in practice, computational paradigms based on Fourier transform summation are often grounded in the relationship between input amplitude and the amplitude of the Fourier spectrum. Our theoretical derivations and simulation results strongly support this perspective. Although this distinction does not significantly impact computational results in some situations, we still recommend establishing a direct connection between numerical values and optical amplitude. Consequently, in experiments involving devices with a linear mapping to optical intensity, such as modulators and detectors, it is necessary to apply appropriate square or root transformations to the input and output values.

Supplementary Note 3 - Theoretical error analysis of POMMM paradigm

The essence of the process of achieving summation through the focusing of light is the FT under an envelope of Hadamard product. However, using FT to perform summation is inherently an approximate approach. To ensure summation accuracy, the Fourier spectrum must be concentrated near the zero frequency, forming an approximate sinc function[1]. Consequently, each element of the matrix should occupy a sufficiently broad spatial distribution to ensure that its variation rate is "slow" relative to the linear phase

variation rate, imposing specific requirements on the modulation aperture and information density of the matrix ("repetitions" mentioned in Supplementary Note 2.1). Furthermore, since this method does not achieve perfect summation, each element of the resulting matrix is affected by unavoidable sidelobes (e.g., spectral leakage and aliasing), which not only influence other elements but are also influenced by the sidelobes of neighboring elements, especially the elements with smaller numerical values. And as the size of the matrix increases, the error will become more serious (increasing in the row direction will increase the information density of the amplitude, and increasing in the column direction will increase the density of the frequency points after FT).

Certainly, this issue is the inherent error source of Fourier transform summation and is widespread in synthetic aperture radar imaging, optical imaging, ONNs, and various applications[1,6], and the methods to reduce errors is very clear: 1. reducing the information density (increasing the aperture or the repetitions), 2. enlarging the spacing between focus in the results, which pertains to the frequency range of the linear phases and the focal length. These methods can significantly reduce errors, but they require a balance between errors and factors such as the size of the optical prototype, the scale of the device surface, pixel gap of SLMs, and computational precision. Therefore, we conduct a detailed analysis of each method.

First, we investigated the influence of information density on the average error of POMMM. An intuitive example is shown in Fig. S7a: when the repetition factor is 1, elements with small values are overwhelmed by the sidelobes of large-value elements, making them indistinguishable. In contrast, with a repetition factor of 40, these two elements can be accurately retrieved, showing high consistency with the reference values. To quantify this, we conducted simulations using three different optical models: fast Fourier transform (FFT) (Fig. S7b), Huygens-Fresnel principle (Fig. S7c), and Rayleigh-Sommerfeld diffraction theory (Fig. S7d). The pixel size was set to 3.6 μm (common in 4K SLM), and the focal length of the Fourier cylindrical lens was set to 200 mm (same as our experiments). For each model, the number of matrix rows was fixed at 20 (i.e., a fixed number of spatial frequency points), and 4 matrix sizes were tested: 20, 40, 60, and 80 columns. For each size, we tested 4 repetition factors: 10, 40, 60, and 80, resulting in 16 groups in total, each with 50 random samples. For every sample, we computed the element-wise mean absolute error, then averaged over the 50 samples within each group. The results show that increasing the repetition factor

significantly reduces the average error, with POMMM achieving highly accurate matrix-matrix multiplication (with errors approaching zero). However, this improvement is not unbounded: under a fixed focal length, increasing the aperture size eventually violates the paraxial approximation, causing discrepancies between the actual optical field results (Fig. S7d) and the ideal Fourier transform results (Fig. S7b). Therefore, as discussed in Supplementary Note 2, increasing the focal length when feasible can be beneficial, as it relaxes the constraints of the paraxial diffraction condition.

Second, we studied the impact of frequency gap on average error. All other parameters were identical to the previous experiment. We tested normalized frequency gaps of 0.2, 0.3, 0.4, and 0.5 (max value is 1). As shown in the illustrative example in Fig. S8a, two elements with a normalized frequency spacing of 0.1 yield significantly better accuracy than those with a spacing of 0.01. The results indicate that increasing the frequency spacing also markedly reduces the computation error (Fig. S8b~d). However, like repetitions, such an increase is also constrained by the validity of the paraxial approximation.

Based on the results mentioned above, increasing the repetitions (or window length) and frequency gap is highly effective in reducing computational errors, leading to more accurate computational outcomes. In summary, an effective approach to reducing POMMM computational errors is to maximize the aperture of matrix modulation (for example, by using modulator arrays with a large size) under the paraxial diffraction conditions, while simultaneously improving the spatial sampling precision of phase modulation. This can be achieved by replacing the PSLM with a fixed, finer phase mask.

Supplementary Note 4 - Details of ONNs based on POMMM

The datasets used in this study are MNIST and Fashion-MNIST. For GPU-based non-negative models and standard models, 60,000 images are selected as the training set, and 10,000 images are used as the test set. Model weights are updated using stochastic gradient descent (SGD) with the Adam optimizer. The training system is configured with two Nvidia RTX 4090 GPUs (24 GB each), an Intel Xeon Gold 5218 CPU, running Windows 11, Python 3.8, and PyTorch 1.10 with CUDA 11.8.

For the direct deployment experiments using the non-negative and real-valued simulated POMMM, the test set comprises 10,000 images. As a reference for the POMMM prototype, direct weight deployment in the non-negative CNN architecture is achieved via a single POMMM simulation (multi-channel convolution).

For ViT models, deployment requires five POMMM simulations, corresponding to the following components: multi-channel convolution (1), multi-head self-attention (3: queries, keys, and values), and multi-sample fully connected layer (1). In the unconstrained weight direct deployment setting, each linear operation is implemented using the real-valued POMMM. The real-valued POMMM is realized through two separate POMMM simulations: in the first, all negative elements are set to zero; in the second, all non-negative elements are set to zero.

Inference experiments based on the POMMM prototype require continuous refreshing of ASLMs, with each refresh triggering an exposure event on the qCMOS. A soft triggering mechanism is used to update both the ASLMs and the qCMOS, with an exposure time of 80 μs. To validate the usability of the prototype, 800 randomly selected images were used for testing. Both the images and corresponding weights were directly loaded into the system without any error correction applied to the inputs or outputs. From these, 600 results were used to fine-tune the weights in the final digital fully connected layer of the CNN and ViT architectures (non-POMMM implementations), while the remaining 200 results were used for evaluation.

Training of ONNs using GPU-based simulated POMMM is relatively straightforward: the tensor core-based MMM functions in each layer are replaced with POMMM-based implementations, as described in the Methods section of the main text. The computational graph is constructed via forward propagation, followed by backpropagation based on the generated graph. All differentiation and backpropagation operations are performed using the PyTorch framework, and training is conducted on a system equipped with two Nvidia A800 GPUs (80 GB each).

Supplementary Note 5 - Detail performance analysis and comparison with other paradigms

5.1 Analysis of computing power and consumption of prototypes

As shown in Fig. 4c in the main text, we mentioned that POMMM is a versatile optical tensor processing paradigm, highlighting its numerous advantages over existing optical computing paradigms. In addition to theoretical comparisons on the paradigm level, we have also provided computational power analyses

resulting from the integration of actual devices into the system as a reference. For one MMM with dimensions [N, M] × [M, N], the operand evaluation method is as follows:

$$C = (M + M) \times N^2 = 2MN^2.$$

Where, C represents the total number of operations. Each row performs M multiply-accumulate operations, resulting in a total of N^2 such operations in MMM. Taking the ASLM in our prototype as an example, a [100, 100] × [100, 100] POMMM is realized by an 80 μs single-shot. The total operations are $2 \times 100^3 = 2 \times 10^6$, and the power consumption is 763.026 μJ (Table S2), so the energy efficiency is 2.62 GOP/J (giga (10^9) of operations per Joule).

However, as an optical computing paradigm, POMMM only requires passive phase modulation in addition to data input and output, so it can match almost all modulators and detectors. In addition, we have proved that in specific tasks, the weights of POMMM can correspond to one value per pixel after training (Fig. 3d). Therefore, if replaced with 4K (2160 × 3840), 10-bit SLM (for example, Upolabs, HDSLM36R-PCIe4in1, 36 W), the parallel operations in single propagation will reach $2 \times 1024 \times 3840 \times 2160 = 1.70 \times 10^{10}$, and the power consumption is (0.834 + 1365.33 × 2 + 2880 + 2812.5) μJ = 8.4239 mJ. So the energy efficiency is 2.02 TOP/J (trillion (10^{12}) of operations per Joule). It is worth noting that all these computing power evaluations take into account device limitations (such as the scale of SLMs and exposure time of the detector) and are not an "idealized" evaluation based solely on the propagation time of light. These are clear examples demonstrating that the high parallel computing capability of our computing paradigm and means that even one parameter changes in the devices within the system can result in a dramatic improvement in computational power. The computational power of POMMM when combined with various SLMs, as well as its comparison with other free-space ONN systems, is shown in Table S3. Transplanting POMMM into various existing ONN systems, without changing device parameters, can significantly enhance the computational power of the systems.

Moreover, regarding computational power and energy efficiency, it is necessary to mention a representative class of optical computing paradigms based on direct paraxial diffraction (often referred to as diffractive neural networks). These paradigms demonstrate remarkable computational throughput and energy efficiency by leveraging the massive interconnectivity of millions of spatial positions and enable multi-

layer computation through direct physical cascading. However, it is important to note that diffractive neural networks do not align with GPU-compatible computational paradigms. Tasks accomplished using millions of optical "neurons" in these paradigms are often equivalent to those requiring only a few thousand-scale GPU operations, such as vector-matrix or matrix-matrix multiplications. While diffractive approaches have shown impressive performance in specific neural network tasks[7,8], there is currently no evidence that the diffraction-based computation paradigm (involving quadratic phase convolution and spatially dependent weighting) can serve as a general framework for implementing universal linear operations. Therefore, as mentioned, comparing the theoretical GPU-compatible computational power is more effective in assessing the performance of optical computing paradigms. And the high-speed analog two-dimensional modulation arrays[9] seem to be the best modulation devices suited for free-space ONN systems.

In conclusion, POMMM demonstrates impressive computational power, as evidenced by both theoretical computational parallelism analysis and comparisons with other computational paradigms under the same modulation device conditions. In some ONN works dominated by device research, replacing the conventional computing paradigm with POMMM can provide the system with greater computational power and a broader range of applications.

5.2 Comparison with wavelength and space multiplexing methods

Wavelength division multiplexing (WDM) technology is widely used in waveguide optics and photonic integrated circuits, especially with the development of optical frequency comb technology[10,11], making WDM easily implementable in waveguide optical computing. However, assigning weights to multiple wavelengths remains challenging. For a small number of wavelengths, it is easy to provide an independent modulation channel for each wavelength, but as the number of wavelengths increases, the cost of wavelength assignment rises. Using a wave shaper is a feasible solution[10], but it’s slow assignment speed significantly impacts the overall system speed, making it unacceptable in real-time weight updating computing systems. In free-space optical computing, the implementation of WDM schemes is relatively complex. Due to the difficulty in assigning different weights to the wavelengths at the same spatial position, a strategy of spatial position-wavelength joint multiplexing is often adopted. The conventional approach is to use a diffraction grating to separate different wavelengths into different spatial positions[12,13]. While this

strategy utilizes WDM, it essentially separates the parallelism from space, just using spatial position resources as an adjunct to wavelength.

In spatial multiplexing, some approaches replicate the input information multiple times and employ a larger "processor" to perform computing[4,14]. This strategy does not fundamentally enhance parallelism but rather serves as a straightforward form of resource stacking. For example, implementing an [N, N] matrix operation using M times spatial position source requires an equivalent spatial resource of [M × N, N]. When such methods are realized using SLMs or other digitally addressed devices, the matrices must be re-encoded for each configuration, introducing additional latency.

In contrast, POMMM eliminates the need for wavelength multiplexing and does not incur extra spatial resource costs. It operates using pre-set, fixed phase modulations, which significantly simplifies the system architecture and reduces spatial demands compared to traditional WDM techniques. More importantly, POMMM retains compatibility with WDM (as shown in Fig. 4a, 4b), enabling multiple POMMM units to perform parallel computations without increasing spatial complexity. This capability supports parallel tensor-matrix multiplications, thereby directly enhancing computational throughput.

5.3 Comparison with optical vector-matrix multiplication and diffraction paradigms in ONNs

Compared to the versatile paradigms such as Stanford method-based[2,15] or planar waveguide-based optical vector-matrix multiplication (OVMM)[16-18], POMMM introduces an additional computational parallelism, that is, the computing power described by time complexity from $O(N^2)$ (VMM) to $O(N^3)$ (MMM). As described in the Methods of the main text and Fig. S9, based on the GPU, the single-channel convolution is VMM, and the multi-channel convolution is MMM. In the same way, the essence of the fully connected layer is VMM and the multi fully-connected is MMM. In comparison to POMMM, traditional OVMM paradigms lack the ability to concurrently perform multi-channel (tensor) convolutions and multi-sample fully connected operations. In addition, with the introduction of various new neural networks, MMM has also been directly introduced into the network structure for versatile tensor processing, such as the multi-head self-attention. To deploy these structures to light, methods based on OVMM requires multiple

modulations and propagations. Repeated electro-optical conversion will greatly reduce efficiency and is not conducive to the establishment of a complete ONN architecture. Of course, if theoretical computational throughput and the number of optical propagation steps required to complete an ONN task are not considered, ONNs based on the vector-matrix multiplication paradigm and those based on POMMM are, in principle, equivalent in performance and equally capable of accomplishing the same ONN tasks. The discrepancies in accuracy observed across different implementations primarily stem from differences in network architecture and inherent imperfections in the optical systems themselves (Table S4).

Furthermore, compared to two-dimensional specialized paradigms based on paraxial diffraction[19-22], POMMM demonstrates remarkable versatility. POMMM can implement all the linear functionalities based on the tensor processing within a single optical system, providing greater flexibility in the design and construction of ONN systems. In fact, diffractive neural networks possess several unique advantages. First, they are highly scalable and naturally suited for large-scale parallelism. Second, the free-space propagation of light enables straightforward cascading and expansion, while directly leveraging complex-amplitude information to perform computational and signal processing tasks. These characteristics allow current diffractive neural networks to outperform GPU-compatible ONN paradigms in terms of both task complexity and inference accuracy (Table S4). However, as mentioned before, their fundamental limitation lies in the significant mismatch between their computational model and that of GPUs. Although diffractive neural networks can indeed realize some neural network tasks by "full connection" across layers, according to the Rayleigh-Sommerfeld diffraction theory, the weight relationships between each neuron correspond to their spatial positions[1,20]. In other words, the weights in the "full connection" process cannot be individual modified or arbitrarily trained, only the weights after this process can be adjusted, corresponding to the biases in neural networks. Since their proposal[20], these networks have been highly controversial and have not been proven robust across various tasks in digital systems. This type of ONN has poor integration with the mainstream direction of neural network development and may struggle to adapt to more complex neural network tasks in the future. In contrast, our proposed paradigm offers superior theoretical computational capacity and broader generality compared to the diffractive approach. Therefore, as system scale continues

353 to grow, our paradigm is expected to emerge as a more favorable optical computing solution for future
354 developments.
355

Supplementary Figures

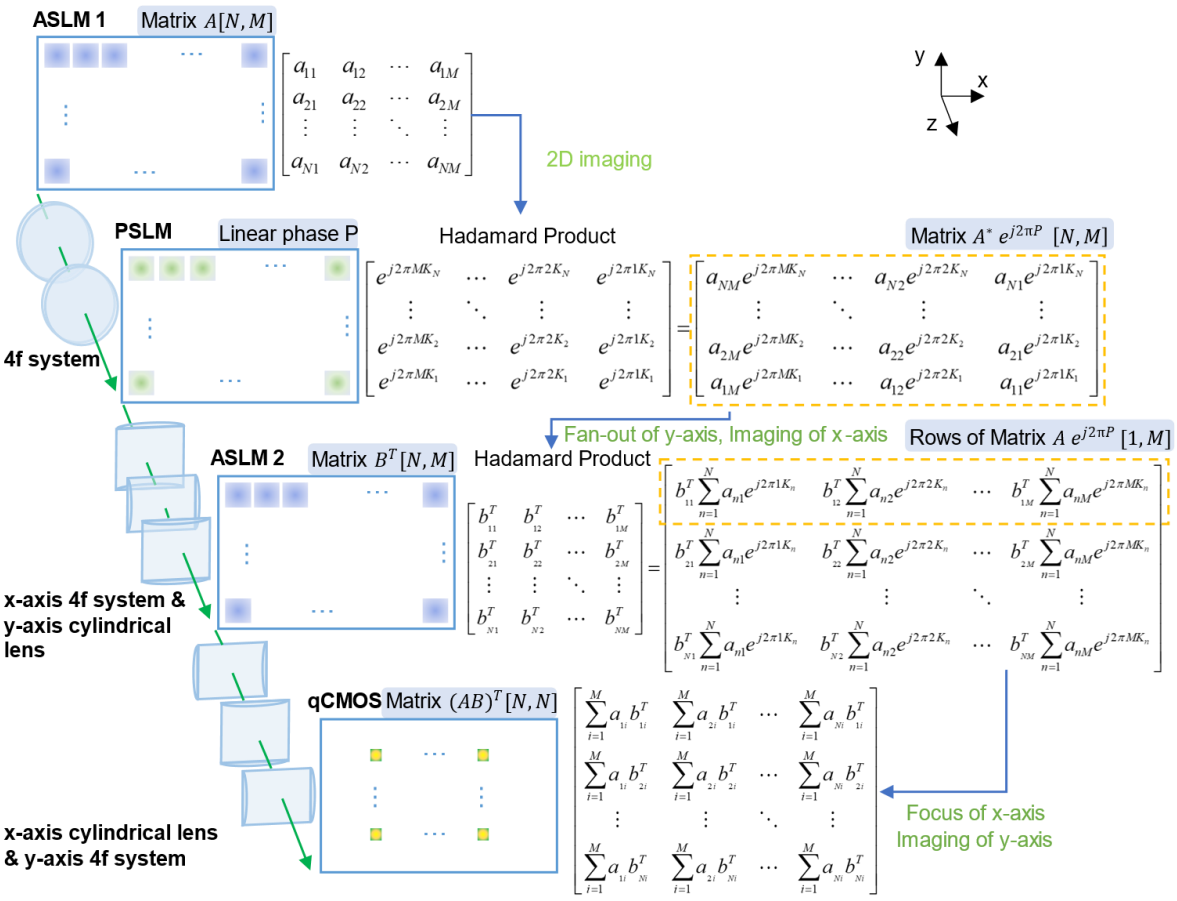


Fig. S1 Detail principle based on Fig. 1. The optical setup (left panel) and the corresponding matrix operation (right panel). The x-direction represents the row direction, the y-direction represents the column direction, and the z-direction corresponds to the direction of light propagation. The fixed phase difference caused by the space-shift property and the flipping caused by the imaging of the last group of cylindrical lenses along the y-direction are omitted in the figure.

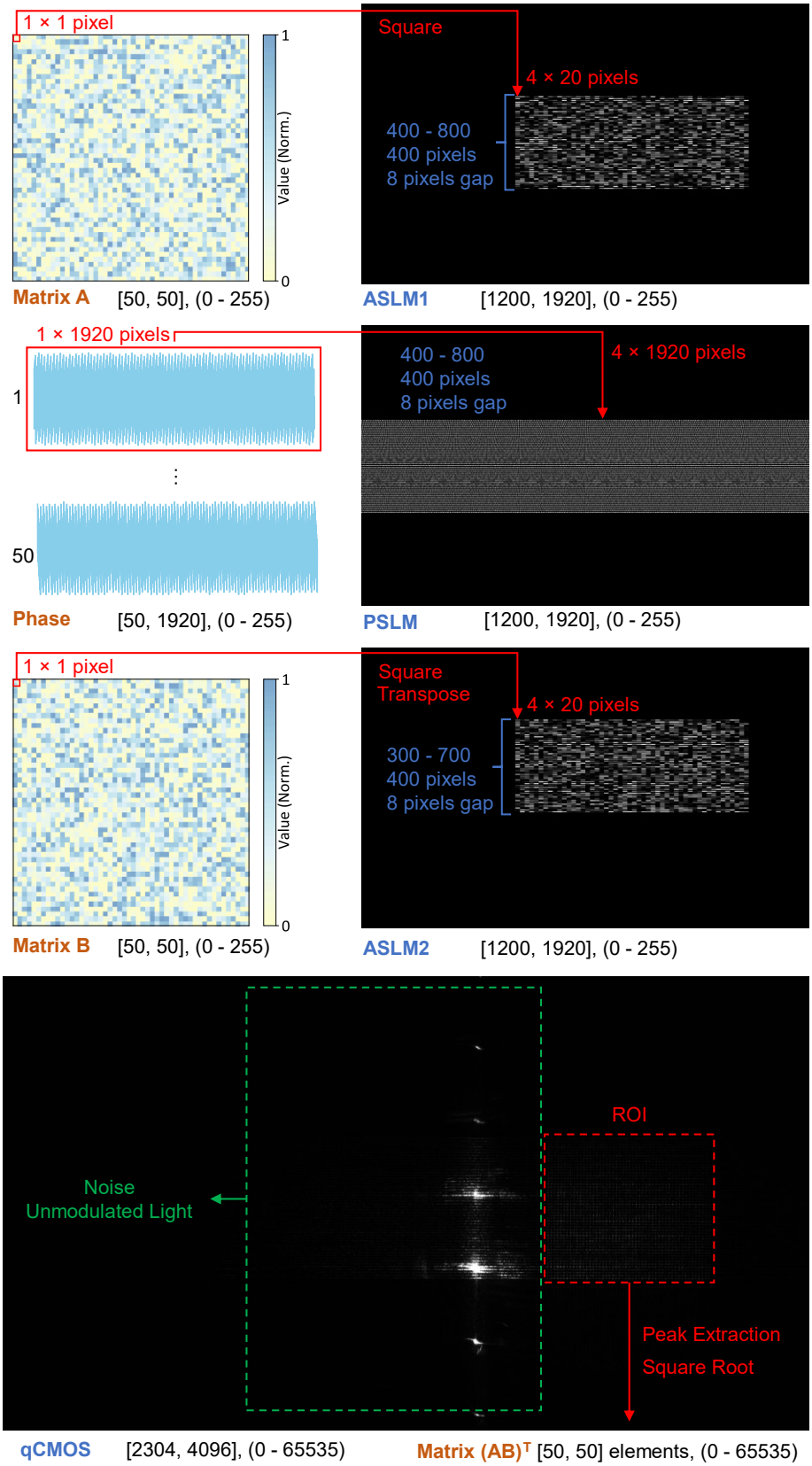
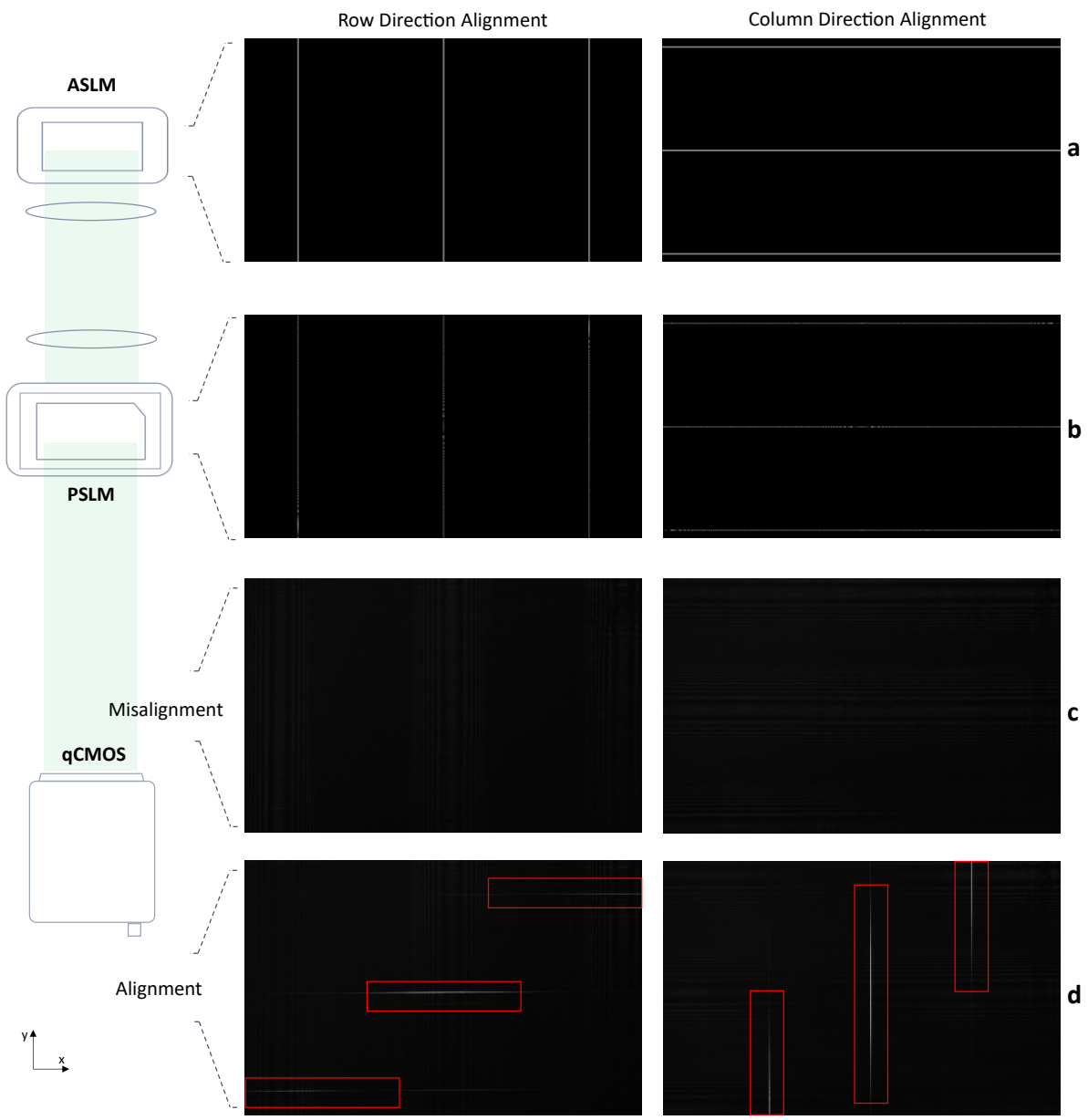

1 × 1 pixel
1
Value (Norm.)
0
Square
4 × 20 pixels
400 - 800
400 pixels
8 pixels gap
Matrix A [50, 50], (0 - 255)
ASLM1 [1200, 1920], (0 - 255)
1 × 1920 pixels
1
50
4 × 1920 pixels
400 - 800
400 pixels
8 pixels gap
Phase [50, 1920], (0 - 255)
PSLM [1200, 1920], (0 - 255)
1 × 1 pixel
Square
Transpose
4 × 20 pixels
300 - 700
400 pixels
8 pixels gap
Matrix B [50, 50], (0 - 255)
ASLM2 [1200, 1920], (0 - 255)
ROI
Noise
Unmodulated Light
Peak Extraction
Square Root
qCMOS [2304, 4096], (0 - 65535)
Matrix (AB)T [50, 50] elements, (0 - 65535)

Fig. S2 A matrices loading and result extraction example of POMMM process. The results captured by qCMOS can be divided into two parts: on the right side is the result matrix (ROI, region of interest), on the left side are the unmodulated optical field. The red box represents the POMMM result, the amplitude of each bright dot represents each element of result matrix. It is worth noting that the row direction undergoes two imaging processes from ASLM1 to ASLM2, so the installation of ASLM1 and ASLM2 in the horizontal direction must be consistent. Since the schematic diagram and formula omit the column direction imaging from ASLM2 to qCMOS, in order to directly obtain $(AB)^T$, it is recommended to install ASLM2 upside down along the column direction, or load data upside down.



373

374 **Fig. S3 Pixel level alignment method. a** The pattern loaded on ASLM. **b** The pattern loaded on PSLM. **c** Misaligned

375 optical field distribution. **d** Aligned optical field distribution.

376

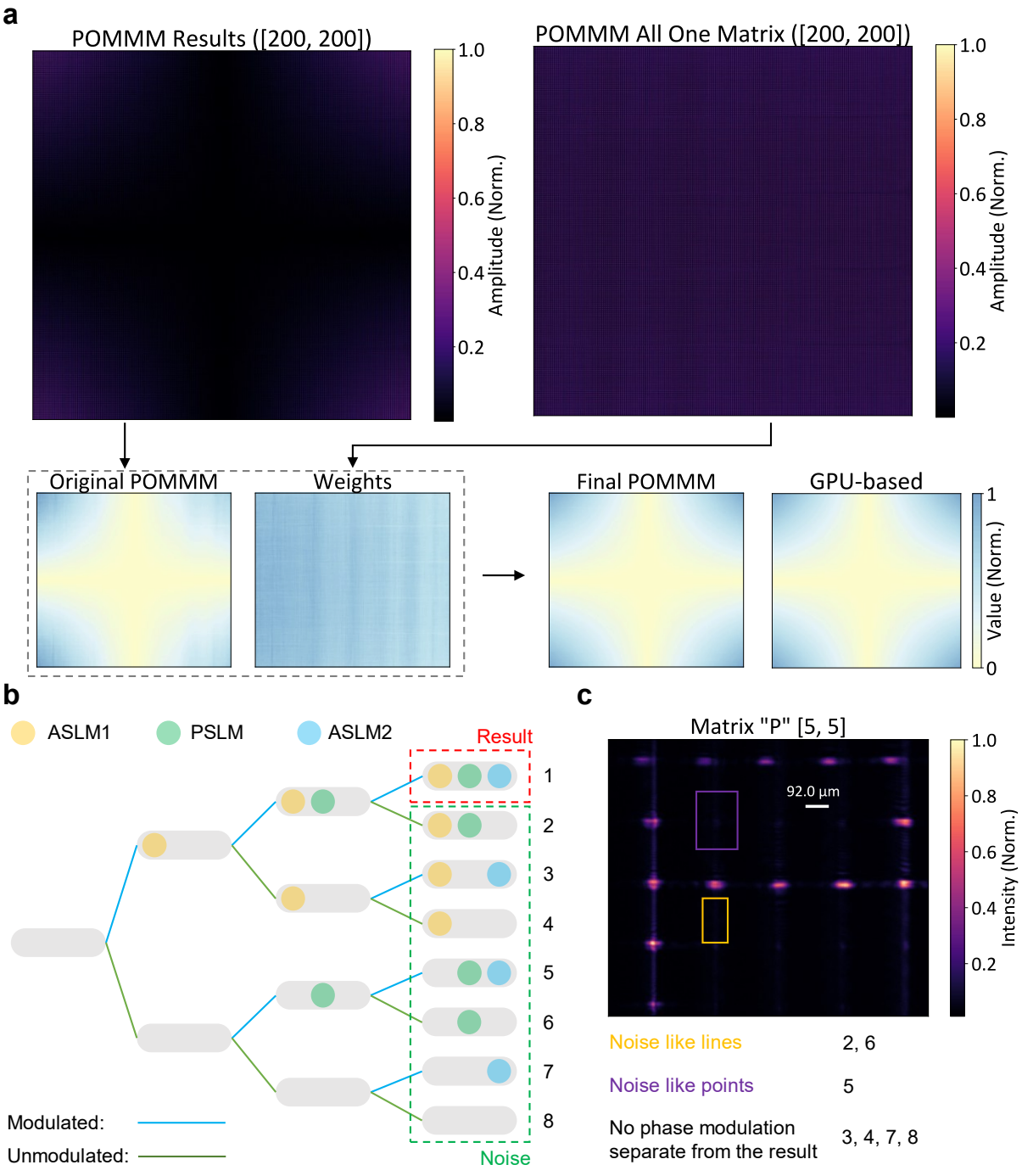
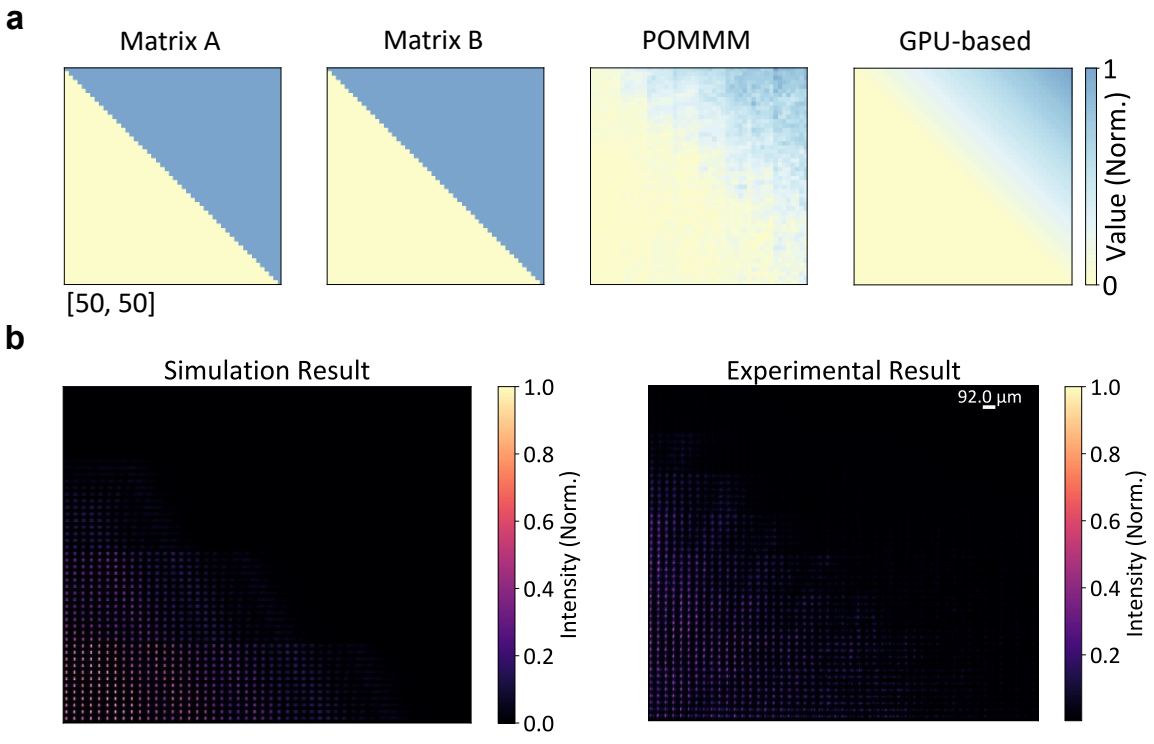


Fig. S4 A result extraction example and the noise analysis. a Process diagram for adjusting result matrix in simulated POMMM ([200, 200] × [200, 200]). The upper panel is the original optical field of simulation POMMM result (due to the limitation of contrast, amplitude distribution is used here for clearer display), the left part is the result to be adjusted (center reverse diffusion symmetric matrix), and the right part is the unit matrix result. The bottom

382 panel describes how to obtain the correct POMMM results. **b** 8 categories of components in experimental results. The
383 component modulated by both ASLM1, PSLM, and ASLM2 constituting the result, while the remainder constitutes
384 noise. **c** An example of noise distribution based on a low SNR [5, 5] experimental POMMM result.

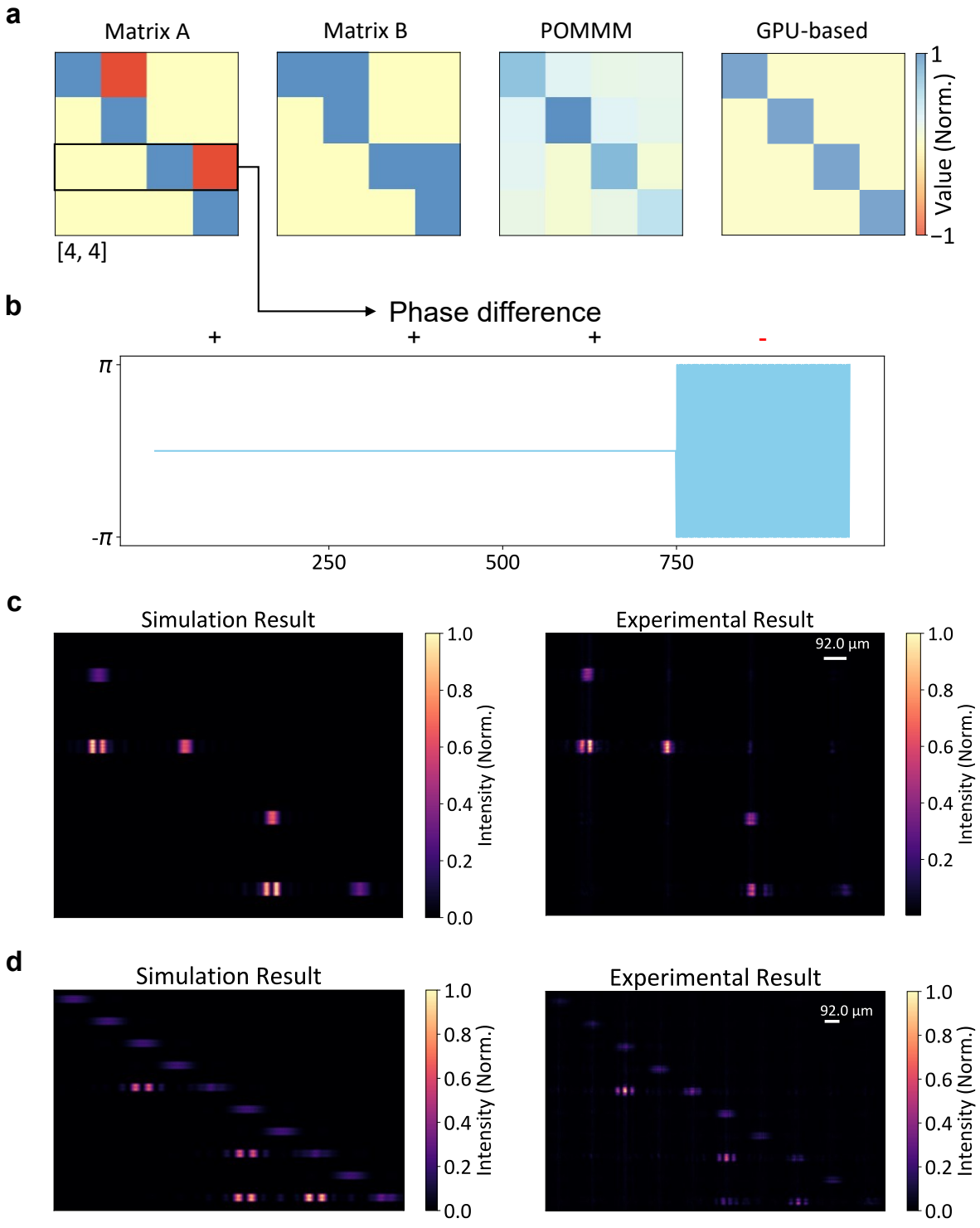
385



386

387 **Fig. S5 Experimental POMMM results compared with GPU-based results. a** Comparison between POMMM
388 results and GPU-based results (two [50, 50] upper triangular matrices result in an [50, 50] upper triangular gradient
389 matrix). **b** Comparison between the original experimental optical field and the simulated optical field of Fig. S5a.

390



391

392 **Fig. S6 Experimental real-valued POMMM results compared with GPU-based results. a** Comparison between

393 real-valued POMMM results and GPU-based results (two [4, 4] complex conjugate produce a [4, 4] diagonal matrix).

b Phase difference between negative and positive elements in POMMM. The negative elements are encoded by an extra phase shifting of π, realizing the destructive interference. **c** Comparison between original experimental optical field and simulated optical field of Fig. S6a ([4, 4] × [4, 4]). **d** Comparison between original experimental optical field and simulated optical field of Fig. 2a in the main text (two [10, 10] complex conjugate produce a [10, 10] diagonal matrix). There are clear dark gaps in the elements with destructive interference in **c** and **d**.

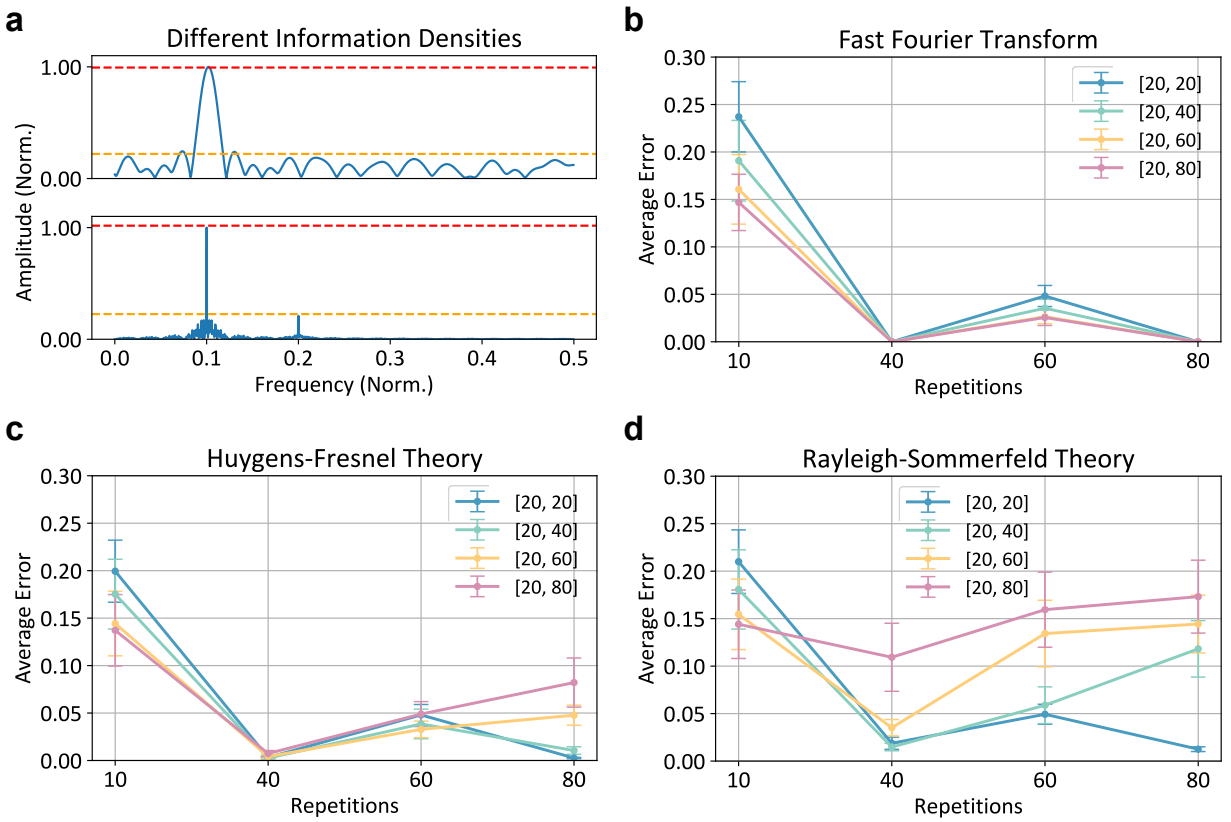
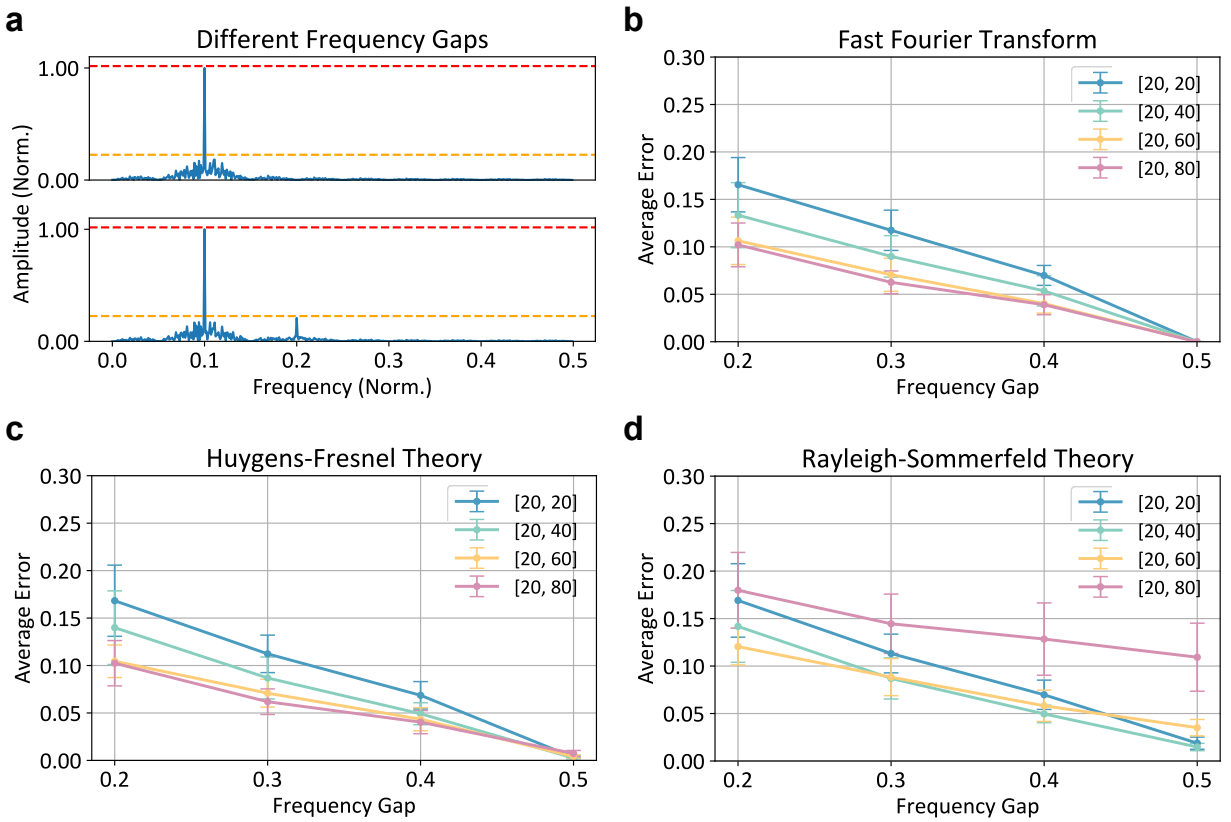


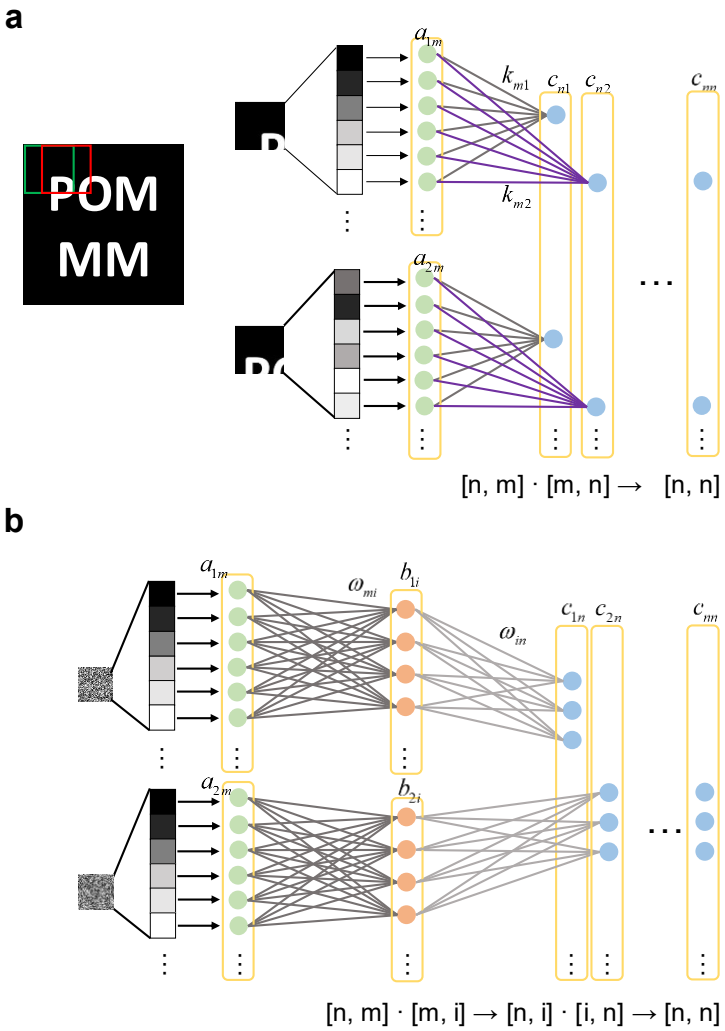
Fig. S7 Relationship between theoretical computing error (Mean Absolute Error) and information density. a Schematic diagram of different information densities. The repetition is 1 in the upper panel and is 40 in the bottom panel. The dashed lines are the reference value. **b** Simulation results based on Fast Fourier Transform. **c** Simulation results based on Huygens-Fresnel theory. **d** Simulation results based on Rayleigh-Sommerfeld theory. Different colors represent different matrix sizes. The horizontal axis indicates different repetitions. Each point shows the average MAE over 50 random samples of the corresponding repetition and size, and the error bars represent ± standard deviation.



408

409 **Fig. S8 Relationship between theoretical computing error (Mean Absolute Error) and frequency gap. a**
410 Schematic diagram of different frequency gap. The normalized frequency gap is 0.01 in the upper panel and is 0.1 in
411 the bottom panel. The dashed lines are the reference value. **b** Simulation results based on Fast Fourier Transform. **c**
412 Simulation results based on Huygens-Fresnel theory. **d** Simulation results based on Rayleigh-Sommerfeld theory.
413 Different colors represent different matrix sizes. The horizontal axis indicates different frequency gaps. Each point
414 shows the average MAE over 50 random samples of the corresponding frequency gap and size, and the error bars
415 represent ± standard deviation.

416



417

418 **Fig. S9 The mathematical process of multi-channel convolution and multi-sample fully connected layers. a** The

419 principle of multi-channel convolution (tensor convolution) realized based on GPU tensor processing. **b** The principle

420 of two cascaded multi-sample fully connected (tensor full connection) layer realized based on GPU tensor processing.

421

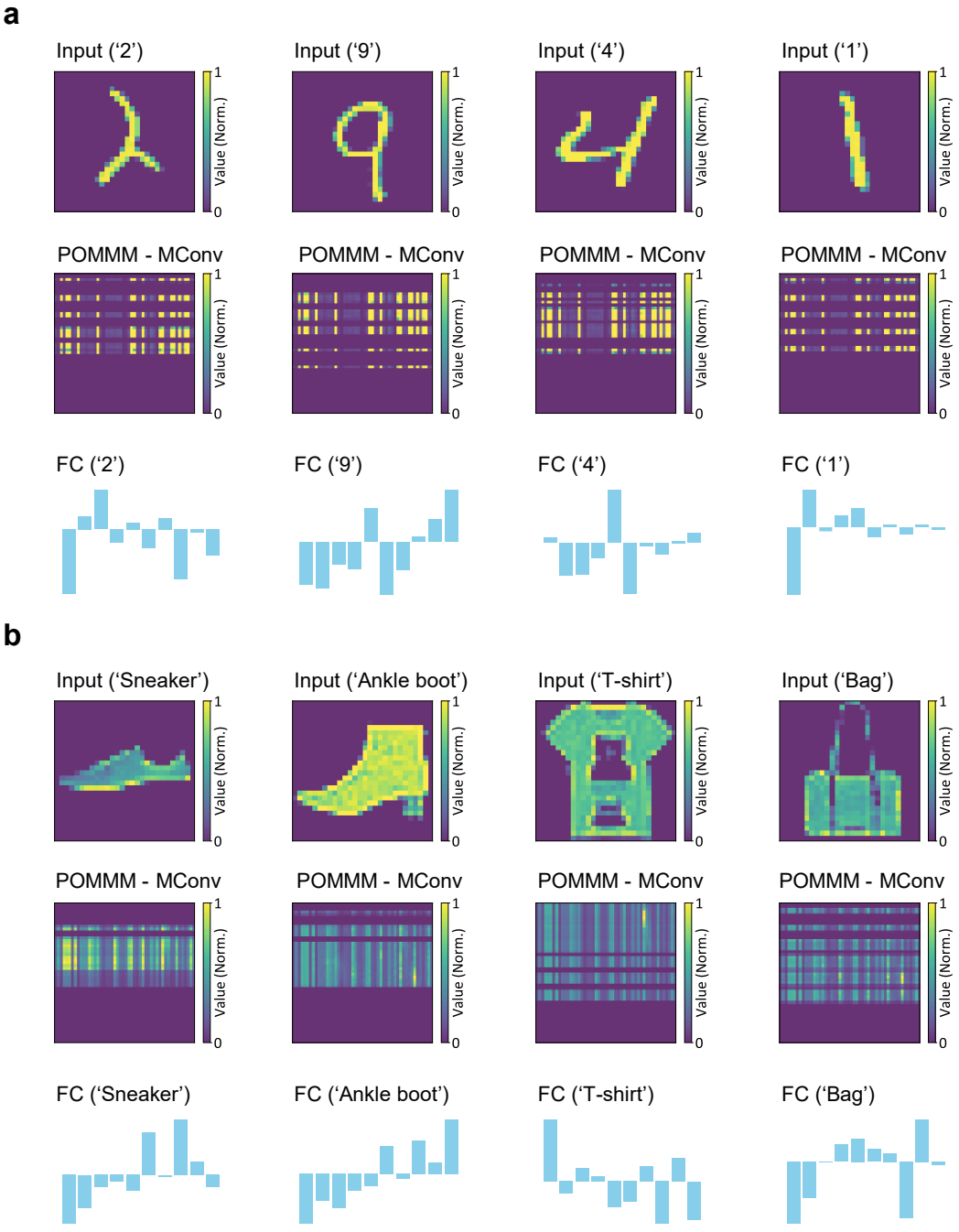


Fig. S10 Experimental results of direct optical deployment of CNN. The input, weight, and output of each layer are directly modulated and extracted without any preprocessing and weighting, ensuring true direct deployment. The result matrices are all [50, 50], only [36, 50] are valued (see Methods). **a** MNIST dataset. MConv, multi-channel convolution; FC, full connection. **b** Fashion MNIST dataset.

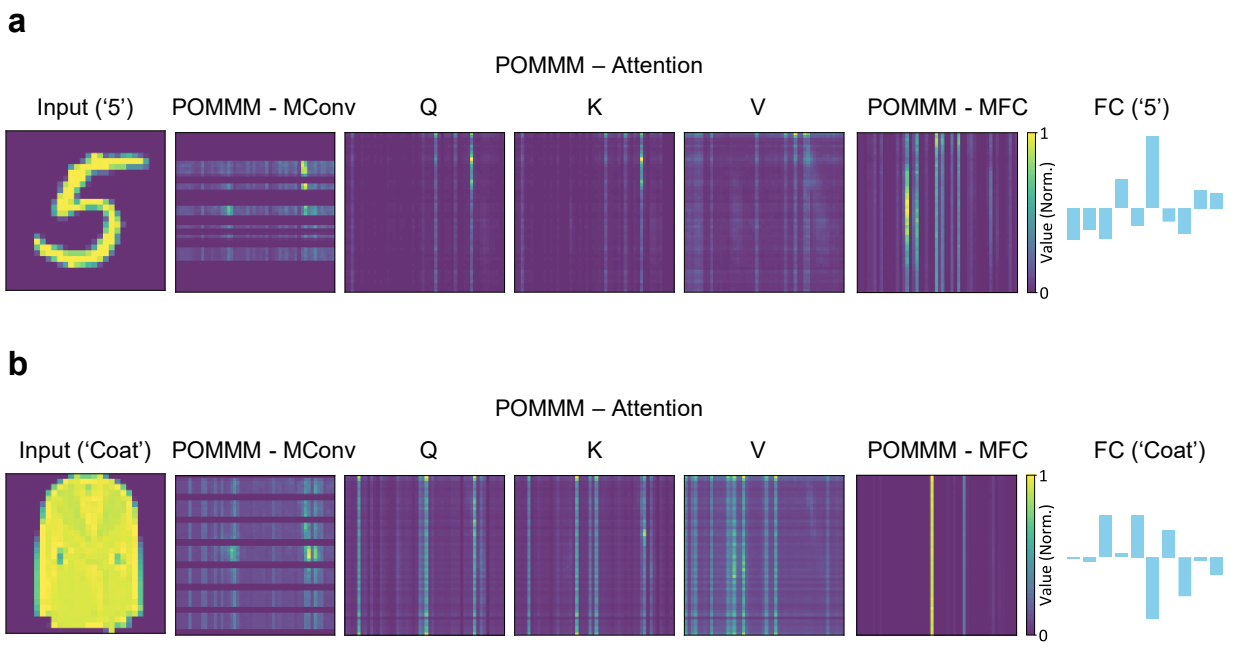


Fig. S11 Experimental results of direct optical deployment of ViT network. The input, weight, and output of each layer are directly modulated and extracted without any preprocessing and weighting, ensuring true direct deployment. The result matrices are all [50, 50], only [49, 50] are valued in MConv (see Methods). **a** MNIST dataset. Q, query; K, key, V, value; MFC, multi-sample full connection. **b** Fashion MNIST dataset.

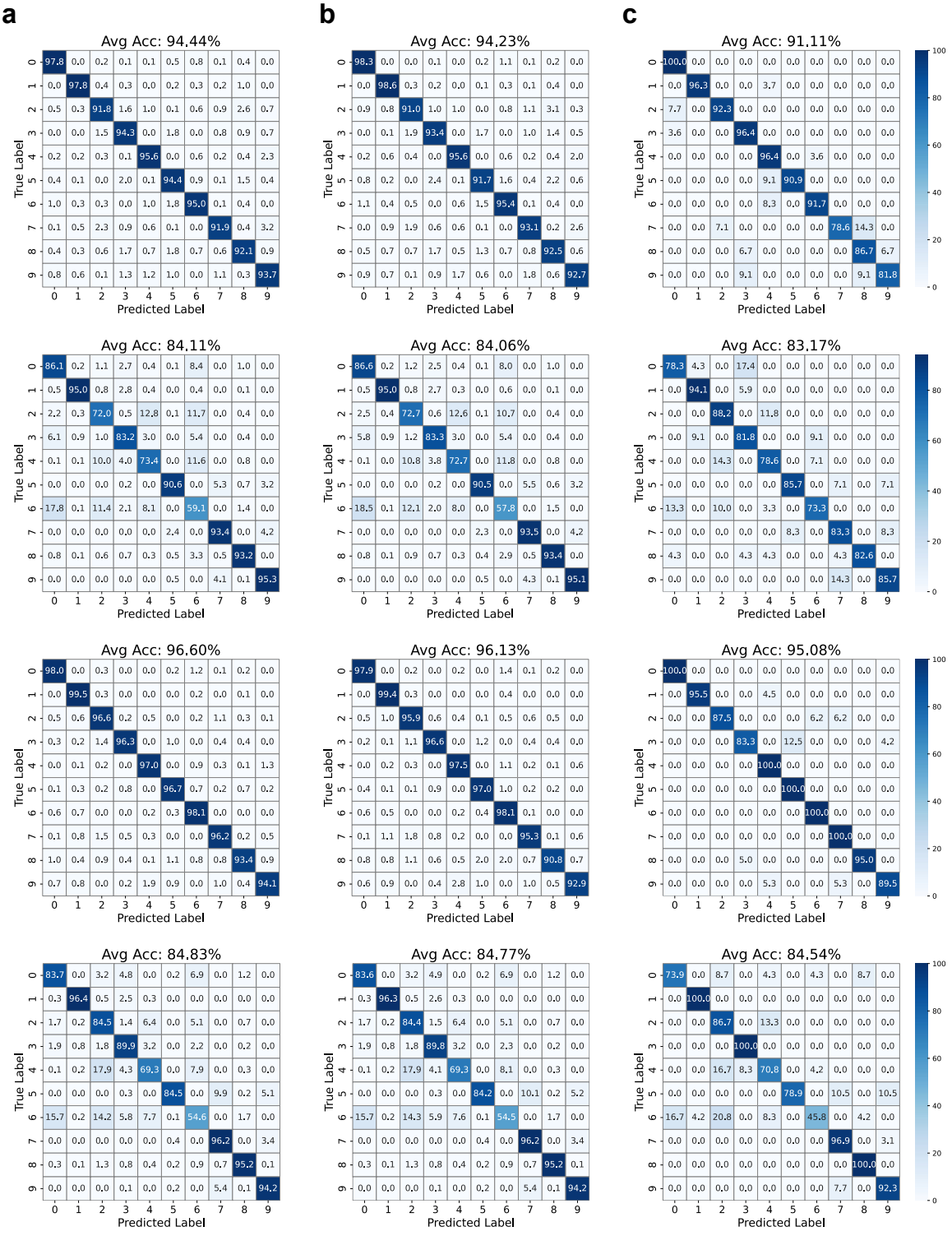
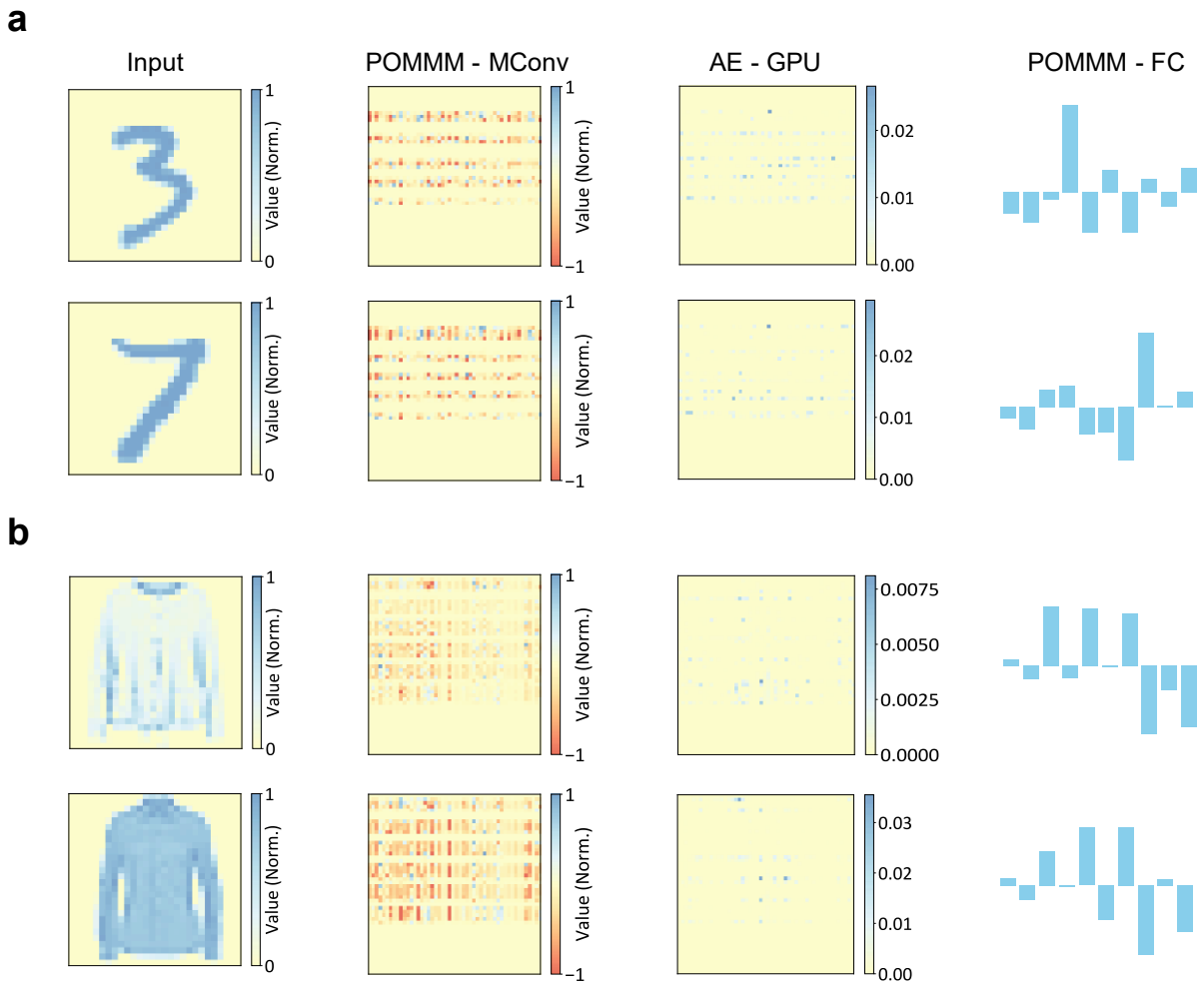


Fig. S12 Confusion matrixes of direct optical deployment tests. From top to bottom: CNN-MNIST, CNN-FMNIST, ViT-MNIST, ViT-FMNIST. Avg Acc, average accuracy, mean accuracy of each category. **a** Trained and inference

436 test based on a GPU-based non-negative MMM (control group). **b** Trained by GPU-based non-negative MMM,
437 inference test on simulation POMMM. **c** Trained by GPU-based non-negative MMM, inference test on POMMM
438 prototype.

439



440

441 **Fig. S13 Theoretical simulation results of full-step direct optical deployment of CNN. a** MNIST dataset. AE-GPU

442 represent the absolute error maps compared with GPU-based results. **b** Fashion MNIST dataset.

443

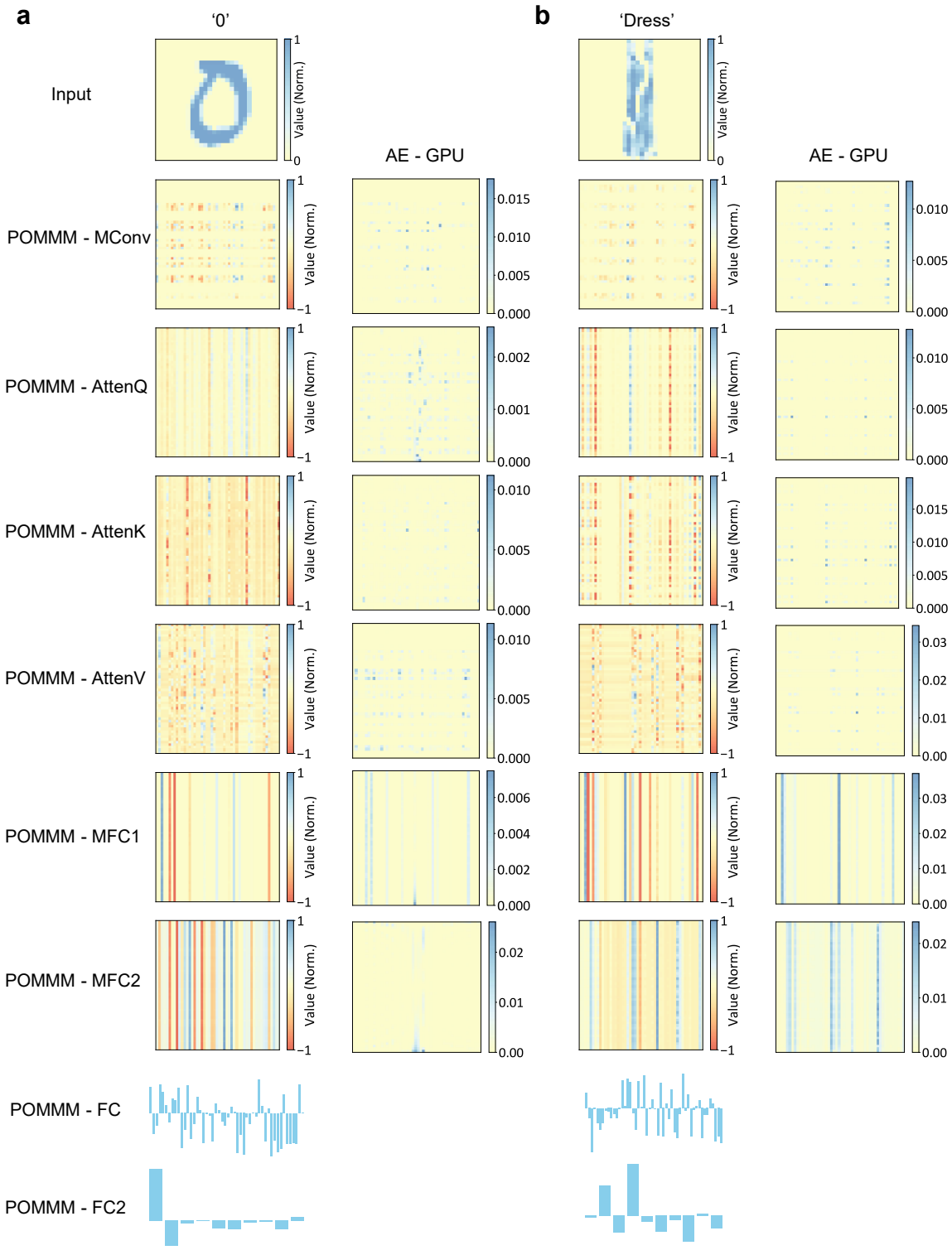


Fig. S14 Theoretical simulation results of full-step direct optical deployment of ViT networks. a MNIST dataset. **b** Fashion MNIST dataset.

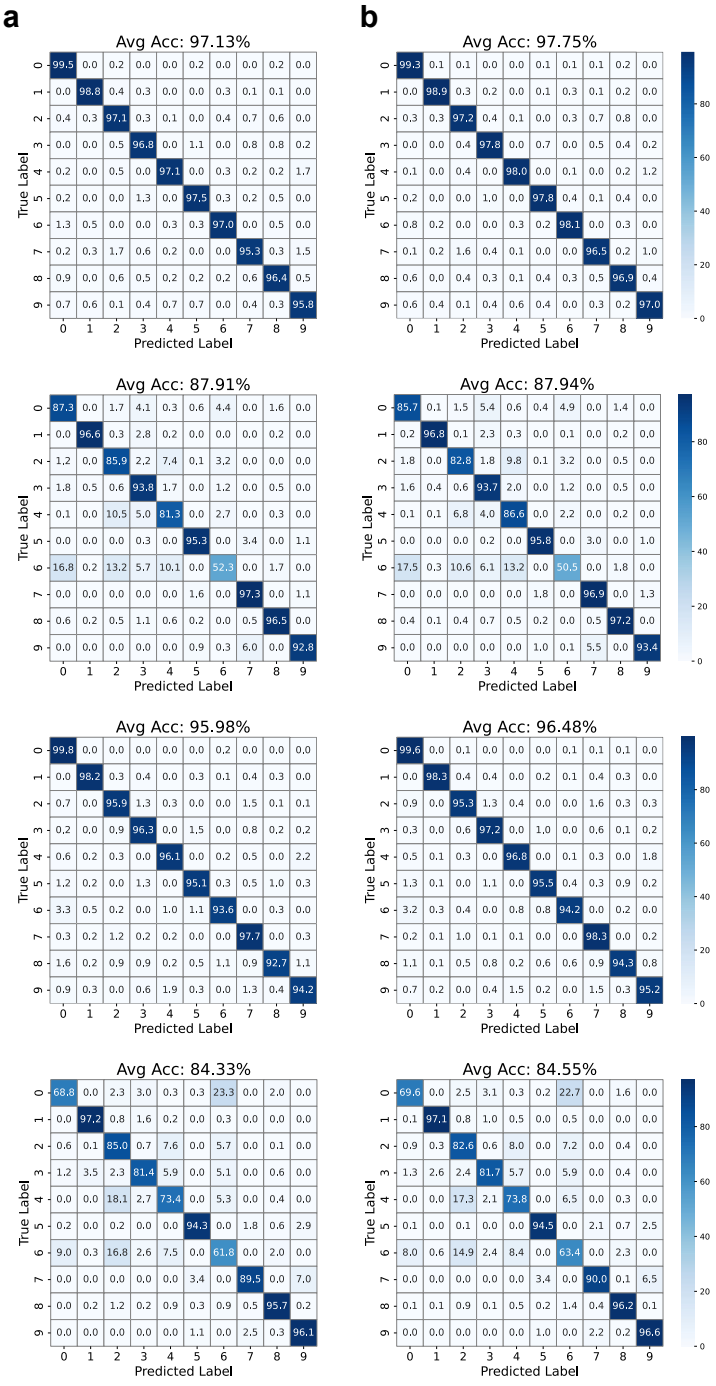


Fig. S15 Confusion matrixes of non-constrained direct optical deployment tests based on simulation POMMM. From top to bottom: CNN-MNIST, CNN-FMNIST, ViT-MNIST, ViT-FMNIST. **a** Trained and inference test based on a GPU-based standard MMM (control group). **b** Trained by GPU-based standard MMM, inference test on simulation POMMM.

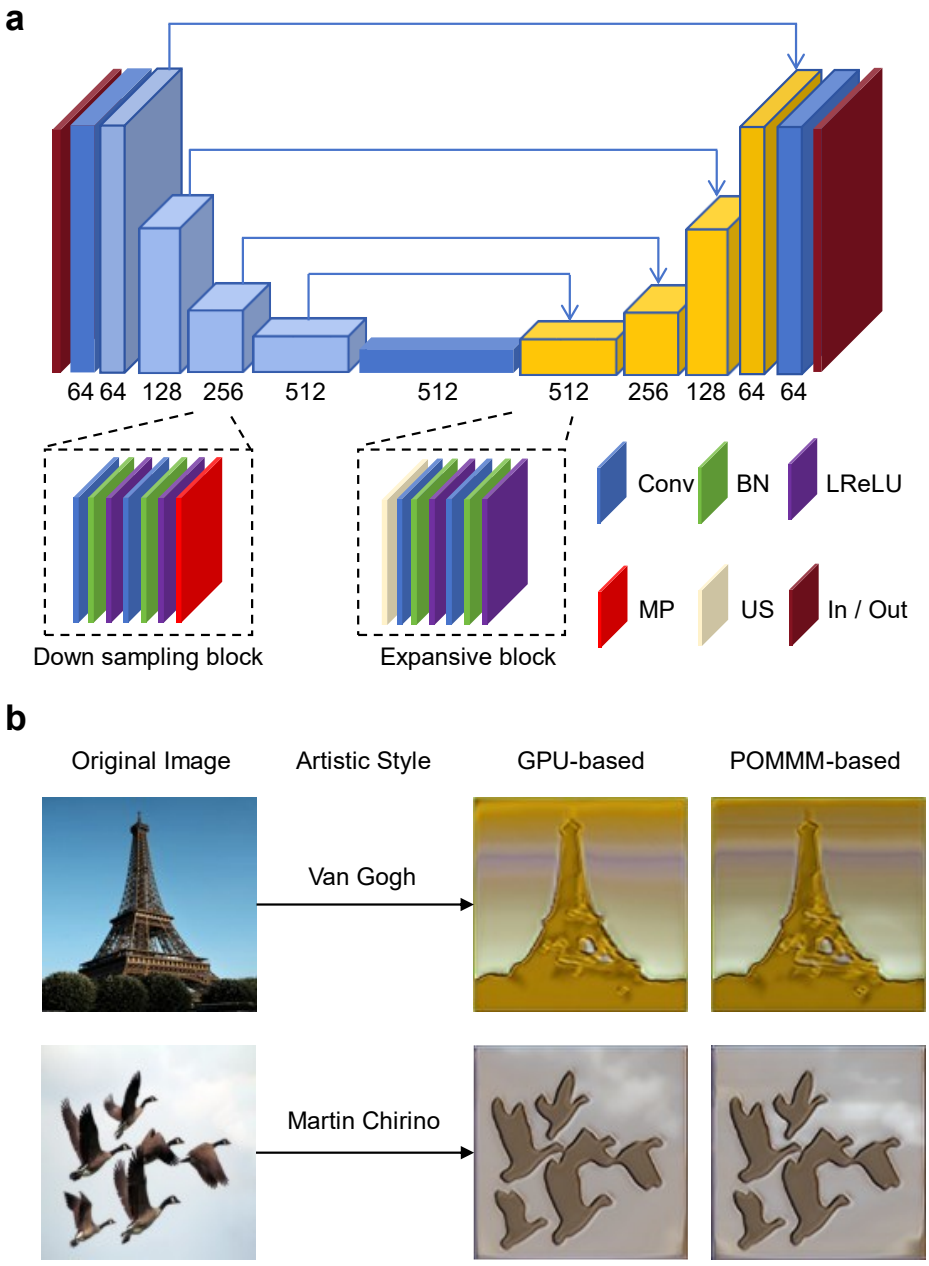


Fig. S16 Simulation results of direct optical deployment of U-Net. a The structure of U-Net for style transfer. It adopts the standard U-Net encoder-decoder architecture and comprises a symmetrical four-level feature processing pipeline. The encoder consists of four cascaded down-sampling blocks, each containing two 3 × 3 convolutional layers (Conv), with the channel number increasing from 64 to 512. Each convolutional layer is equipped with batch normalization (BN) and Leaky ReLU (LReLU) activation ($\alpha = 0.1$), followed by a 2 × 2 max pooling layer (MP) for feature subsampling. The decoder adopts a mirror-symmetrical structure that performs 2 × up-sampling (US) through

transposed convolutions, followed by dual 3 × 3 convolutional layers for feature refinement. The features are then concatenated along the channel dimension with the corresponding layers' features from the encoder. A single 3 × 3 convolutional bottleneck layer is set in the middle of the network to maintain the feature resolution. Finally, a 3 × 3 convolutional layer maps the 64 channel features to an RGB three-channel space, achieving end-to-end reconstruction with aligned input and output dimensions. The entire architecture realizes multi-scale feature aggregation through skip connections and contains a total of 19 convolutional layers, forming a complete "contraction-feature maintenance-expansion" processing link. **b** The test images include one provided by the authors (upper panel) and one from the public domain (bottom panel, CC0, the link: https://picryl.com/media/geese-leaving-unsplash-efff55). The artistic styles are derived from Vincent van Gogh (upper panel) and Martín Chirino (bottom panel). The model was trained on the DIV2K dataset[23], using GPU-based non-negative MMM. Then the trained model is deployed directly on both the GPU-based computing and POMMM-based computing (weights of the first 17 layers, the last 2 layers are realized by GPU-based MMM) to perform style transfer.

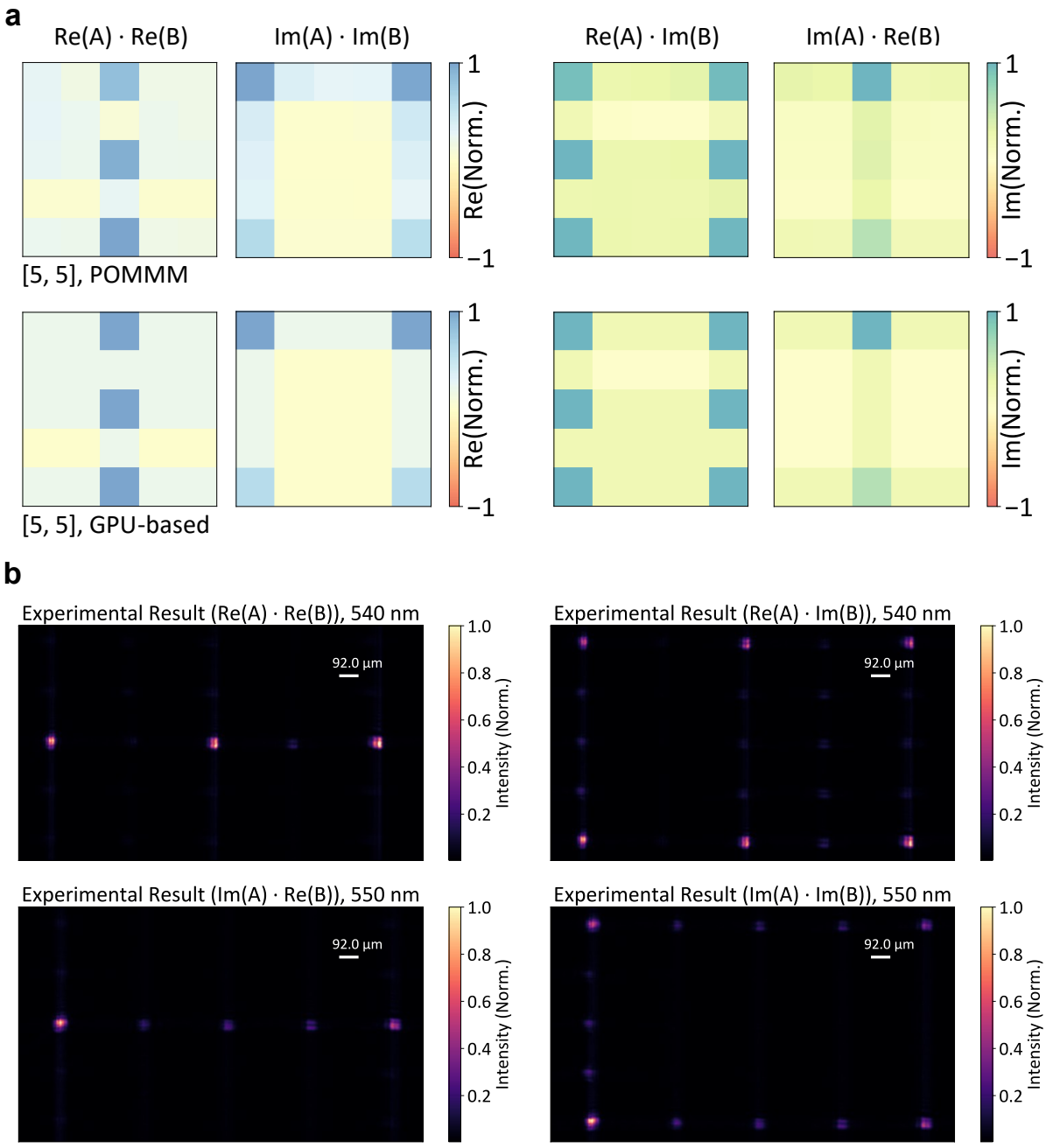
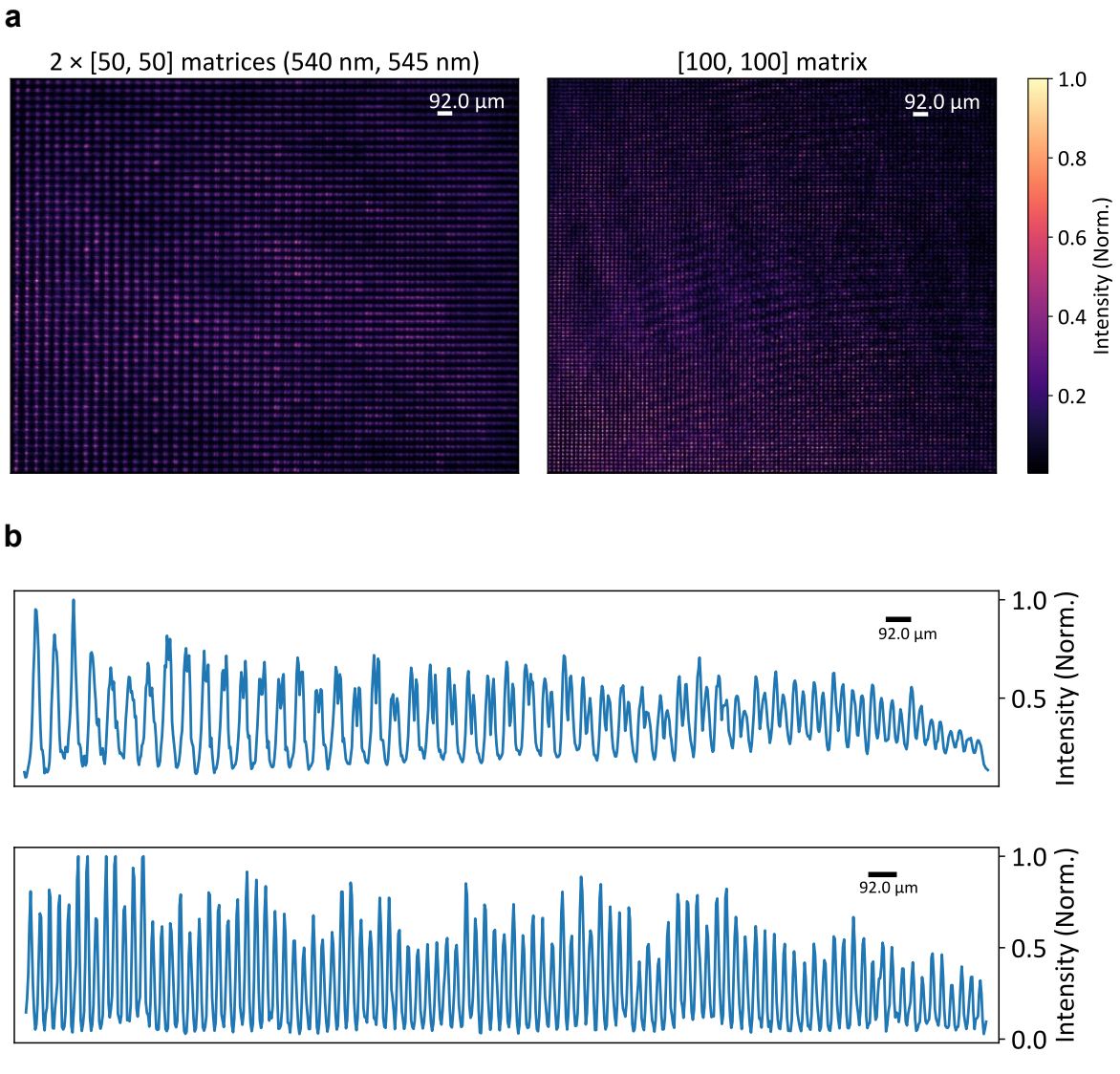


Fig. S17 Experimental results and corresponding original optical field of complex-valued POMMM by different wavelengths. a Comparison between complex-valued POMMM results and GPU-based results (complex matrix A and B are the input Fig. 4b in the main text). **b** Corresponding original optical field. The results of different wavelengths have slight spatial movement and are naturally separated, like the function of a spectrometer with different gratings simultaneously.



478

479 **Fig. S18 The exploration of scale limits of experimental POMMM prototype. a** Original experimental optical field
480 of two-wavelength ([2, 50, 50] × [50, 50]) and single-wavelength ([100, 100] × [100, 100]) POMMM. **b** Cross-section
481 of one row of results (corresponding to the maximum value of several rows of the original optical field). Upper panel,
482 two-wavelength. Bottom panel, single wavelength.

483

Supplementary Tables

Table S1 Comparison with state-of-the-art GPU-compatible optical computing paradigms

Reference	Function	Paradigm	Parallelism (Single-shot)	Modulators (Computing resource)	Experimental demonstrated computing size
Ref. [24]	Dot Product	MZI combination	N	N	9
Ref. [10]		Time - wavelength multiplexing *	N	1	90
Ref. [9,25]		Spatial summation by lens	N^2	N^2	[5, 5], [8, 4]
Ref. [16,26]	VMM	MZI array	N^2	N^2	[4, 4], [64, 64]
Ref. [15]		Stanford method	N^2	N^2	[56, 56]
Ref. [17]		MRR array *	N^2	N^2	[4, 4]
Ref. [27]	MMM	PCM array *	N^3	N^2	[8, 8, 16]
Ref. [12]		Stanford method + wavelength multiplexing *	N^3	N^2	[10, 10, 10]
Ref. [13]		OFT + wavelength multiplexing *	N^3	N^2	[8, 8, 7]
This work		**OFT + Phase coding + spatial spectrum multiplexing**	**N^3**	**N^2**	**[100, 100, 100]**
	TMM	**OFT + Phase coding + spatial spectrum multiplexing + wavelength multiplexing ***	**N^4**	**N^2**	**[2, 50, 50, 50]**

In this table, we compare the computing paradigm, without discussing the performance of the devices used. The symbol * denotes WDM. VMM, vector-matrix multiplication; MMM, matrix-matrix multiplication; TMM, tensor-matrix multiplication; MZI, Mach–Zehnder interferometer; MRR, micro-ring resonator; PCM, phase change material; OFT, optical Fourier transform.

Table S2 Power consumption analysis of our prototype for demonstration

Device	Power (Device)	Power consumption (Single [100, 100] POMMM)
Laser	145 W (Supplied power), 0.022 W (Output power)	0.306 μJ
ASLM1	18 W (Supplied power)	250 μJ
PSLM	18 W (Supplied power)	250 μJ
ASLM2	18 W (Supplied power)	250 μJ
qCMOS	150 W (Supplied power)	12.72 μJ
Total	349 W (Supplied power)	763.026 μJ

At this laser output power, the light intensity integration time required for a single POMMM is 80 microseconds. The power consumption is calculated according to the optical field range used, the number of modulation and detection units, and the total power. The detailed power consumption evaluation method is as follows. For [100, 100] × [100, 100] POMMM, we use total 4 × 100 × 1000 pixels in SLMs, and the total pixels are 1200 × 1920 (do not consider the alternative pixels). And The exposure time for each POMMM is 80 microseconds, using 100 × 100 pixels on the qCMOS (total pixels are 2304 × 4096). So the power consumption of each SLM for single POMMM is: $(4 \times 100 \times 1000) / (1200 \times 1920) \times (80 \times 10^{-6}) \times 18\ \text{J} \approx 250\ \mu\text{J}$, and for qCMOS is: $(100 \times 100) / (2304 \times 4096) \times (80 \times 10^{-6}) \times 150\ \text{J} \approx 12.72\ \mu\text{J}$. For the laser, assuming that the expanded laser beam fully irradiates the SLM surface, the power consumption is: $(4 \times 100 \times 1000) / (1200 \times 1920) \times (80 \times 10^{-6}) \times 0.022\ \text{J} \approx 0.306\ \mu\text{J}$.

Table S3 Comparison with typical free-space optical computing paradigm based on same devices

Device					Paradigm	Actual Scale	Energy Efficiency (GOP/J)
Type	**Scale**	**Refresh rate (Hz)**	**Digit (bits)**	**Brand**			
GPU*	-	-	8	Nvidia A100 [28]	Digital	-	1.56×10^3
LC-SLM (Our prototype used)	1080 × 1920	60	8	Upolabs	**POMMM (N^3)**	[100,100] × [100,100]	2.62
LC-SLM-4K	2160 × 3840	180	10	Upolabs/Holoeye	Dot product $(N)^{\#}$	[1,3840] · [3840,1]	1.92×10^{-3}
					OVMM $(N^2)^{\#}$	[1,3840] × [3840,2160]	1.97
					POMMM $(N^3)^{\#}$	[1024,3840] × [3840,2160]	2.02×10^3
DMD	1600 × 2560	12987	1	Vialux	Dot product $(N)^{\#}$	[1,2560] · [2560,1]	6.93×10^{-4}
					OVMM $(N^2)^{\#}$	[1,2560] × [2560,1600]	1.11
					POMMM $(N^3)^{\#}$	[1024,2560] × [2560,1600]	1.13×10^3
VCSEL array*	20 × 20 (16 5 × 5 arrays)	25×10^9	Analog	Ref. [9]	2D-Dot product (N^2)*	[20, 20] · [20, 20]	4×10^5
					POMMM (N^3)	[20, 20] × [20, 20]	8×10^6

LC, liquid crystal; VCSEL, vertical-cavity surface-emitting laser; the symbol * estimates based on parameters provided in the referenced paper. The symbol # denotes full use of pixels.

504 **Table S4 Comparison with state-of-the-art optical neural networks**

Reference	Architecture	Parallelism	Paradigm	Data type	Tasks	Accuracy
Ref. [29]	RNN	N (DP)	Imaging and OFT of FS	N, R	Prediction of chaotic MG sequence	-
Ref. [24]	CNN	N (DP)	MZI combination	N, R	AUTOMAP (Reconstruction)	-
Ref. [10]		N (DP)	Time - wavelength multiplexing *	N, R	MNIST	Sim: 89.6%, Exp: 87.6% (500 Samples)
Ref. [9]		N^2 (2D-DP)	2D-Imaging and OFT of FS	N, R	MNIST	Sim: 95.1%, Exp: 93.1% (1000 Samples)
Ref. [21]		N^2 (2D-DP)	2D-Imaging and OFT of FS	N, R	MNIST	Exp: 99% (130 Samples)
Ref. [26]		N^2 (2D-DP)	2D-Imaging and OFT of FS	N, R	The phases of a prototypical Ising model (NL)	-
Ref. [16]		N^2 (VMM)	MZI array	N, R	Speech command	Exp: 76.7% (180 Samples)
Ref. [30]		N^3 (MConv)	Multi-MRR array and delay lines *	N, R	Video action recognition	Exp: 97.9% (96 Samples)
Ref. [4]		N^3 (MConv)	OFT and Dammann grating of FS	N, R	MNIST	Exp: 97.3% (1000 Samples)
Ref. [28]		N^3 (MMM)	PCM array *	N, R	MNIST	Sim: 96.1%, Exp: 95.3% (10000 Samples)
Ref. [13]		N^3 (MMM)	OFT of FS + wavelength multiplexing *	N, R	MNIST	Sim: 97.5%, Exp: 96.4% (1000 Samples)
Ref. [2]	FCNN	N^2 (VMM)	Stanford method	N, C	MNIST	Exp: 93.2% (5000 Samples)
Ref. [31]	FNN	$N^2\log(N)$ (2D-FT)	OFT of FS	N, R	MNIST	Exp: 98% (5000 Samples)
					CIFAR-10	Exp: 54% (5000 Samples)

Ref. [32]				N, C	CIFAR-10	Sim: 44.4%, Exp: 51.0% (100 Samples)
Ref. [19]	DNN	$N^2\log(N)$ (2D-FT + Dot product + 2D-IFT)	Diffraction of FS	N, C	MNIST	Sim: 99.0%, Exp: 96.6% (10000 Samples)
					FMNIST	Sim: 90.2%, Exp: 84.6% (10000 Samples)
					KTH	Sim: 96.3%, Exp: 96.4% (2160 Samples)
Ref. [20]					MNIST	Exp: 93.39% (10000 Samples)
					FMNIST	Exp: 81.13% (10000 Samples)
Ref. [22]					CIFAR-10	Exp: 89.53% (10000 Samples)
					ImageNet-32	Exp: 44.7% (1500 Samples)
Ref. [33]		Nlog(N) (1D-FT + Dot product + 1D-IFT)	Diffraction of PW	N, R	MNIST	Exp: 86.0% (100 Samples)
Ref. [34]				N, R	Four-vowel classification	Exp: 93.8% (64 Samples)
Ref. [35]				N, R	Iris flower, (NL)	Exp: 96.7% (30 Samples)
					Speech command (NL)	Exp: 91.7% (120 Samples)
This work	CNN	N^3, N^4 (MMM, TMM)	POMMM	C	MNIST	Sim: 94.23% (N), 97.75% (10000 Samples) Exp: 91.11% (200 Samples)
					FMNIST	Sim: 84.06% (N), 87.94% (10000 Samples) Exp: 83.17% (200 Samples)

	ViT				MNIST	Sim: 96.13% (N), 96.48% (10000 Samples) Exp: 95.08% (200 Samples)
					FMNIST	Sim: 84.77% (N), 84.55% (10000 Samples) Exp: 84.54% (200 Samples)
	UNet				DIV2K (Sim, N)	-

In this table, we compare the functions and performance of different optical computing paradigms in neural network tasks. RNN, recurrent neural network; CNN, convolutional neural network; FCNN, fully connected neural network; FNN, Fourier neural network; DNN, diffractive neural network; ViT, vision transformer; DP, dot product; MConv, multi-channel convolution; FT, Fourier transform; IFT, inverse Fourier transform; OFT, optical Fourier transform; FS, free space; PW, planner waveguide; N, non-negative; R, real; C, complex; NL, non-linear; Sim, simulation; Exp, experiment.

References

1 Goodman, J. W. *Introduction to Fourier optics*. (Roberts and Company publishers, 2005).

2 Spall, J., Guo, X. & Lvovsky, A. I. Hybrid training of optical neural networks. *Optica* **9**, 803-811 (2022).

3 Wright, L. G. *et al.* Deep physical neural networks trained with backpropagation. *Nature* **601**, 549-555 (2022).

4 Ma, G., Yu, J., Zhu, R. & Zhou, C. Optical multi-imaging–casting accelerator for fully parallel universal convolution computing. *Photonics Research* **11**, 299-312 (2023).

5 Liu, X., Kilcullen, P., Wang, Y., Helfield, B. & Liang, J. Diffraction-gated real-time ultrahigh-speed mapping photography. *Optica* **10**, 1223-1230 (2023).

6 Richards, M. A. *Fundamentals of radar signal processing*. Vol. 1 (Mcgraw-hill New York, 2005).

7 Chen, Y. *et al.* All-analog photoelectronic chip for high-speed vision tasks. *Nature* **623**, 48-57 (2023).

8 Xu, Z. *et al.* Large-scale photonic chiplet Taichi empowers 160-TOPS/W artificial general intelligence. *Science* **384**, 202-209 (2024).

9 Chen, Z. *et al.* Deep learning with coherent VCSEL neural networks. *Nature Photonics* **17**, 723-730 (2023).

10 Xu, X. *et al.* 11 TOPS photonic convolutional accelerator for optical neural networks. *Nature* **589**, 44-51 (2021).

11 Shu, H. *et al.* Microcomb-driven silicon photonic systems. *Nature* **605**, 457-463 (2022).

12 Latifpour, M. H., Park, B. J., Yamamoto, Y. & Suh, M.-G. Hyperspectral in-memory computing with optical frequency combs and programmable optical memories. *Optica* **11**, 932-939 (2024).

13 Luan, C., Davis III, R., Chen, Z., Englund, D. & Hamerly, R. Single-Shot Matrix-Matrix Multiplication Optical Tensor Processor for Deep Learning. *arXiv preprint arXiv:2503.24356* (2025).

14 Ma, G. *et al.* Dammann gratings-based truly parallel optical matrix multiplication accelerator. *Optics Letters* **48**, 2301-2304 (2023).

15 Spall, J., Guo, X., Barrett, T. D. & Lvovsky, A. Fully reconfigurable coherent optical vector–matrix multiplication. *Optics Letters* **45**, 5752-5755 (2020).

16 Shen, Y. *et al.* Deep learning with coherent nanophotonic circuits. *Nature Photonics* **11**, 441-446 (2017).

17 Yang, L., Ji, R., Zhang, L., Ding, J. & Xu, Q. On-chip CMOS-compatible optical signal processor. *Optics Express* **20**, 13560-13565 (2012).

18 Clements, W. R., Humphreys, P. C., Metcalf, B. J., Kolthammer, W. S. & Walmsley, I. A. Optimal design for universal multiport interferometers. *Optica* **3**, 1460-1465 (2016).

19 Zhou, T. *et al.* Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit. *Nature Photonics* **15**, 367-373 (2021).

20 Lin, X. *et al.* All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004-1008 (2018).
21 Wang, T. *et al.* An optical neural network using less than 1 photon per multiplication. *Nature Communications* **13**, 123 (2022).
22 Yuan, X., Wang, Y., Xu, Z., Zhou, T. & Fang, L. Training large-scale optoelectronic neural networks with dual-neuron optical-artificial learning. *Nature Communications* **14**, 7110 (2023).
23 Agustsson, E. & Timofte, R. in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops.* 126-135.
24 Xu, S. *et al.* Optical coherent dot-product chip for sophisticated deep learning regression. *Light: Science & Applications* **10**, 221 (2021).
25 Zuo, Y. *et al.* All-optical neural network with nonlinear activation functions. *Optica* **6**, 1132-1137 (2019).
26 Hua, S. *et al.* An integrated large-scale photonic accelerator with ultralow latency. *Nature* **640**, 361-367 (2025).
27 Feldmann, J. *et al.* Parallel convolutional processing using an integrated photonic tensor core. *Nature* **589**, 52-58 (2021).
28 NVIDIA, P. & Linux, R. H. E. NVIDIA DGX A100. *Data-Center/nvidia-dgx-a100-datasheet. pdf* (2022).
29 Bueno, J. *et al.* Reinforcement learning in a large-scale photonic recurrent neural network. *Optica* **5**, 756-760 (2018).
30 Xu, S., Wang, J., Yi, S. & Zou, W. High-order tensor flow processing using integrated photonic circuits. *Nature Communications* **13**, 7970 (2022).
31 Miscuglio, M. *et al.* Massively parallel amplitude-only Fourier neural network. *Optica* **7**, 1812-1819 (2020).
32 Chang, J., Sitzmann, V., Dun, X., Heidrich, W. & Wetzstein, G. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports* **8**, 1-10 (2018).
33 Fu, T. *et al.* Photonic machine learning with on-chip diffractive optics. *Nature Communications* **14**, 70 (2023).
34 Wu, T., Menarini, M., Gao, Z. & Feng, L. Lithography-free reconfigurable integrated photonic processor. *Nature Photonics* **17**, 710-716 (2023).
35 Wu, T., Li, Y., Ge, L. & Feng, L. Field-programmable photonic nonlinearity. *Nature Photonics*, 1-8 (2025).