A programmable diffractive deep neural network based on a digital-coding metasurface array

Che Liu [1,2,3,6], Qian Ma [1,2,3,6], Zhang Jie Luo [1,2,3], Qiao Ru Hong[1,2], Qiang Xiao[1,2,3], Hao Chi Zhang [1,2,3], Long Miao[3,4], Wen Ming Yu[1,2,4], Qiang Cheng [1,2,3], Lianlin Li [3,5] and Tie Jun Cui [1,2,3 ✉]

The development of artificial intelligence is typically focused on computer algorithms and integrated circuits. Recently, all-optical diffractive deep neural networks have been created that are based on passive structures and can perform complicated functions designed by computer-based neural networks. However, once a passive diffractive deep neural network architecture is fabricated, its function is fixed. Here we report a programmable diffractive deep neural network that is based on a multi-layer digital-coding metasurface array. Each meta-atom on the metasurfaces is integrated with two amplifier chips and acts an active artificial neuron, providing a dynamic modulation range of 35 dB (from −22 dB to 13 dB). We show that the system, which we term a programmable artificial intelligence machine, can handle various deep learning tasks for wave sensing, including image classification, mobile communication coding–decoding and real-time multi-beam focusing. We also develop a reinforcement learning algorithm for on-site learning and a discrete optimization algorithm for digital coding.

The development of artificial intelligence (AI) is focused on two main technical areas: computer-based machine learning methods such as deep learning[1], extreme learning[2] and reinforcement learning[3], and integrated circuits and optical chips for specific functions[4–7]. AI is typically created by using hierarchical artificial neural networks (ANNs) to imitate the structure of neurons[1] and simulate intelligent actions in human decision-making processes, and it has found widespread applications in face recognition[8,9], automatic driving[10,11], language processing[12,13] and medical diagnostics[14,15]. Beyond computer-based and circuit-based AI technologies, all wave-based ANNs have also recently been developed using three-dimensional (3D) printed optical lens arrays[16]. Such diffractive optical platforms (also known as diffractive deep neural networks; D²NN) take advantage of the wave property of photons to realize parallel calculations and simulate different interconnection structures at the speed of light[16,17]. However, wave-based D²NNs are still passive devices and thus have fixed network architectures once fabricated and cannot be re-trained for other targets and tasks, which limits their functionalities and applications.

A tunable D²NN modulated by reflection-type spatial light modulators (termed a reflection-type D²NN) was recently developed[18], but the reflection light path limits the number of layers of the neural networks, and only a two-layer neural network was created. In another reflection-type D²NN, an optoelectronic structure was used to increase the number of layers[19]. In this approach, the connection between layers was implemented with electronic circuits and thus a time delay was introduced, breaking the light-speed calculation capabilities. A transmission-type D²NN would be the preferred scheme to achieve both multi-layer neural networks and light-speed computing[16], but a D²NN hardware with real-time programmable nodes and quick-response learning ability is necessary. Furthermore, current D²NN platforms still rely on

a computer to optimize the parameters and require prior information about the environment[16–19].

In this Article we report an active and reprogrammable transmission-type D²NN structure that can directly process electromagnetic (EM) waves in free space for wave sensing, identification and wireless communications. The weight-reprogrammable nodes, which are necessary to create a programmable and re-trainable wave-based D²NN structure, are achieved using programmable metasurfaces[20] and information metasurfaces[21]. We term our system a programmable artificial intelligence machine (PAIM).

Programmable artificial intelligence machine. In the past decade, sophisticated metamaterials and metasurfaces have been developed for manipulating light and EM waves[22–25]. The digital-coding representation of meta-atoms allows an information metasurface to have reprogrammable abilities and provides a direct connection between the physical world and the digital world[20,21,26,27]. With the aid of controllable active components, programmable metasurfaces can manipulate reflected or transmitted EM waves in real time under digital instructions from field-programmable gate arrays (FPGAs). Various functions and applications have been achieved using such information metasurfaces, including wireless communications[28–31], computational imaging[32,33], space–time modulation[34] and self-adaptively smart devices[35,36].

To fabricate our PAIM we use an array of information metasurfaces in which the multi-layer metasurfaces act as the programmable physical layers of D²NN. We design the PAIM to be a real-time re-trainable system, with parameters that can be digitally set to realize artificial neurons. In the physical layer, the PAIM can hierarchically manipulate the energy distribution of transmitted EM waves using a five-layer information metasurface array[37,38], from which the amplitude of the transmitted wave through each artificial neuron
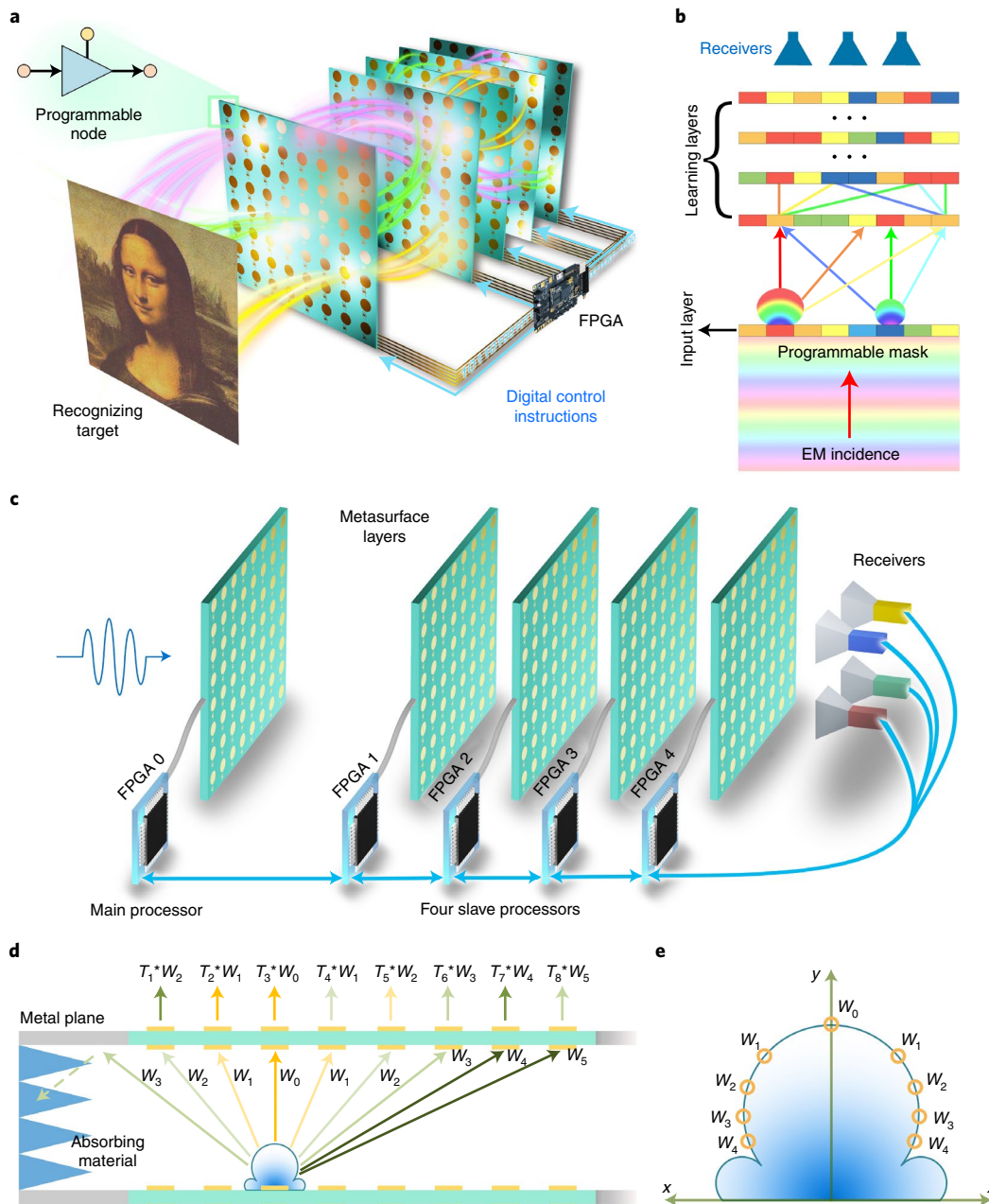
Fig. 1 | A reprogrammable D²NN platform. a, An array of programmable metasurfaces is used to construct the PAIM, in which several FGPAs are installed to control the gain factor of each artificial neuron, making PAIM a real-time and re-trainable intelligent machine. **b**, Schematic diagram of PAIM. An artificial neuron in the learning layer will receive the waves radiated from all artificial neurons in the former layer, making the PAIM structure a fully connected network. The transmission coefficient of each artificial neuron can be trained by using supervised/unsupervised learning or even reinforcement learning methods to achieve various functions. The first layer acts as the input layer by using preset transmission coefficients to encode the input information into the spatial distribution of the EM energy. **c**, The assembled FPGA modulation network for PAIM. The receiving antenna array will receive the field distribution of the EM waves, which is processed by FPGAs. FPGAs also guide the update of the bias voltages of the artificial neurons. **d**, The transmitted wave of an artificial neuron, multiplied by propagation factors W_i, illuminates all artificial neurons in the next layer. The EM wave is then multiplied by the complex-valued transmission coefficient T_i to act as the secondary source of the wave. **e**, The radiation pattern of an artificial neuron.

(meta-atom) could be enhanced or attenuated by controlling the amplifier chips through FPGAs (Fig. 1a–c). The phase modulation of the transmitted wave is coupled with the amplitude modulation, and both of them together constitute the complex-valued gain factors of the programmable artificial neuron. The complex-valued gain factor is a one-to-one response corresponding to bias voltage, which is modulated by a customized FPGA circuit (Figs. 1c and 4b, and Supplementary Note 1).

When the incident beam passes through a programmable artificial neuron in the first-layer metasurface, the amplitude and phase of the transmitted wave are determined by the product of the incident electric field and the complex-valued transmission coefficient of the artificial neuron, and the transmitted wave will act as a secondary source and illuminate all programmable artificial neurons in the second-layer metasurface (Fig. 1b,d), based on the Huygens–Fresnel principle[16]. The transmitted waves (from all directions)
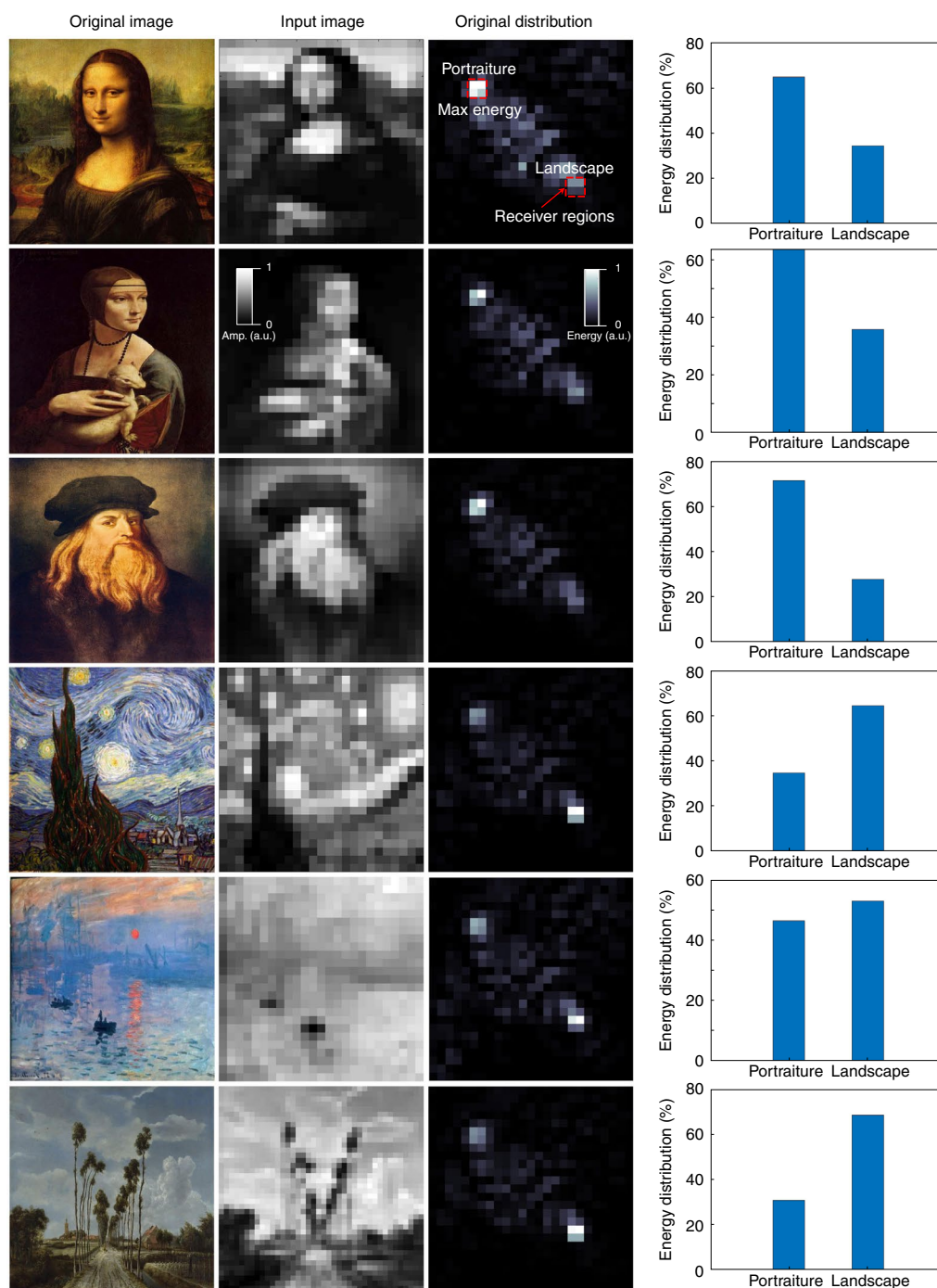
Fig. 2 | The simulation results for oil-painting recognition. Before inputting to the PAIM, the original images (first column) are greyed and reshaped to 25 × 25 pixels (second column). According to the output energy distribution (third column), the receiver region with the maximum receiving energy corresponds to the kind of classification result (fourth column). The results indicate that the PAIM has the potential to imitate high-level human intelligence, such as art appreciation. Credits: (top to bottom) World History Archive/Alamy Stock Photo; Sowa Sergiusz/Alamy Stock Photo; Lebrecht Music & Arts/Alamy Stock Photo; Martin Shields/Alamy Stock Photo; Artefact/Alamy Stock Photo; classicpaintings/Alamy Stock Photo.

illuminating an artificial neuron in the second layer are then added up and the whole acts as the wave incident on the artificial neuron in the second-layer metasurface (Fig. 1b). This process is continued until the last metasurface layer. The reprogrammable interconnection architecture in the PAIM is the factor fundamental to achieving the dynamically artificial neurons.

According to the radiation pattern of the artificial neuron (Fig. 1e), the power transmitted by the artificial neuron presents a certain gain distribution on the next-layer metasurface (Fig. 1d).

Accordingly, the forward propagation model (see Forward propagation model section) of the PAIM can be regarded as a fully connected network (Supplementary Fig. 1). However, compared with the traditional fully connected network constructed from real numbers[1], the PAIM parameters have complex values and the trainable parts are complex-valued transmission coefficients of the artificial neurons. Hence, we have fewer trainable parameters. The traditional error backpropagation method could be used to train the PAIM parameters (see Error backpropagation section). Also, owing to its
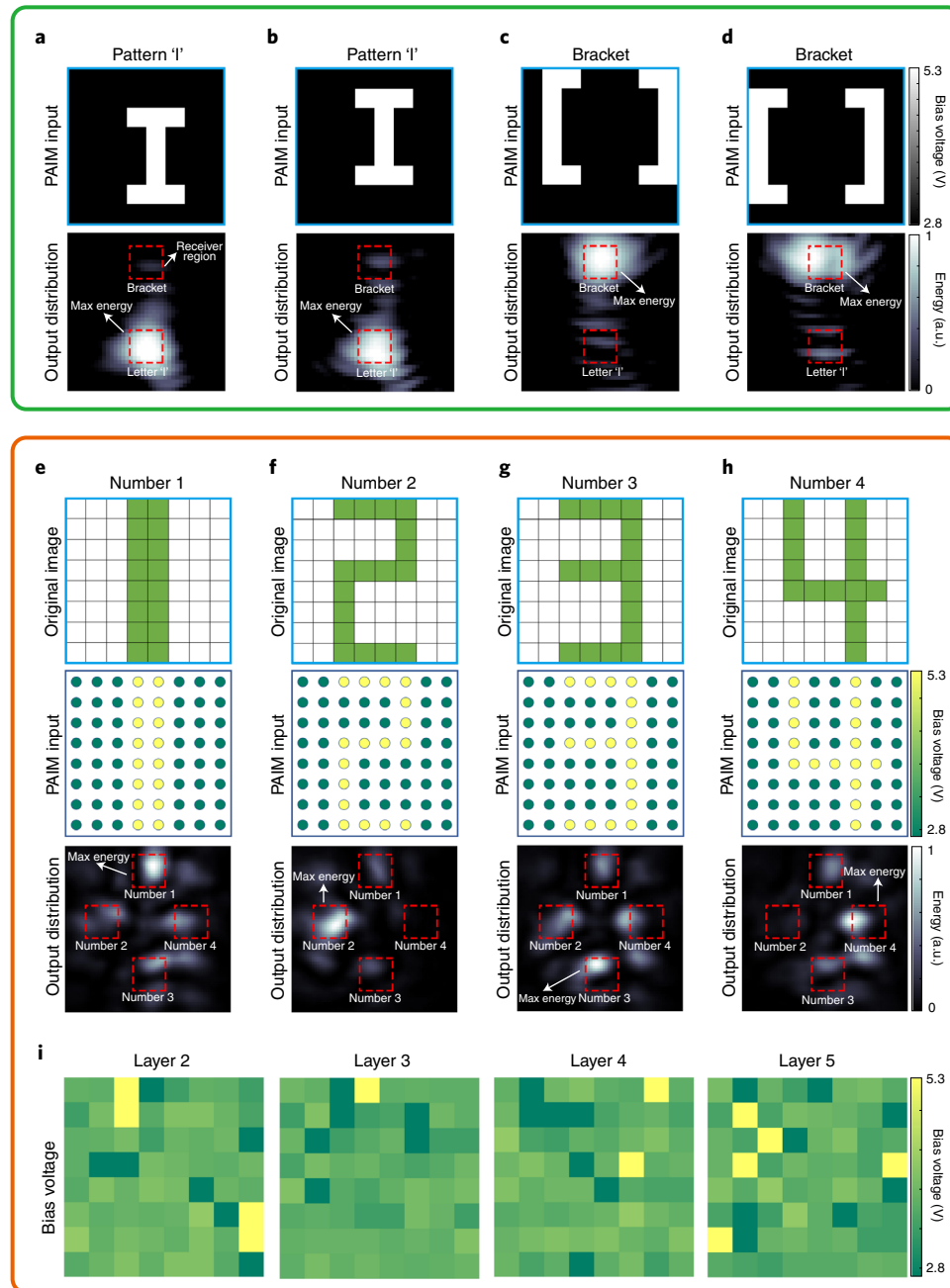
Fig. 3 | Experimental results of image classifications using PAIM. a–d, Two kinds of pattern (letter 'I' and brackets '[ ]') are represented by the distributions of bias voltages of 8 × 8 artificial neurons in the first PAIM layer. The input image consists of 8 × 8 blocky squares of colour, and each square represents the bias voltage of an artificial neuron. The artificial neurons in the remaining four layers are assigned the bias voltages designed during the training process and can recognize the two patterns by ranking the receiving energies from the two receiving antennas. The patterns of letter 'I' in **a** and **b** and of '[ ]' in **c** and **d** have different locations. **e–h,** The original images of four digits (1, 2, 3 and 4 in **e**, **f**, **g** and **h**, respectively) are discretized into 8 × 8 blocky squares of colour, representing different levels of bias voltages of 8 × 8 artificial neurons in the first PAIM layer. The images of the output distributions indicate the expected testing results. **i,** The bias-voltage configurations of layers 2 to 5, obtained by a training process run on a computer.

fast parameter-switching ability and direct feedback from receivers (Fig. 1c), the PAIM enables self-learning capability by using the data gained from direct interaction with environment, and does not need any prior knowledge. Thus, the PAIM possesses reinforcement learning capacity[39].

We note that the artificial neurons work in their linear range in this study, so the PAIM is a linear system in the complex domain. Because we measure the amplitude of the output electric fields as the final output results of the PAIM, and this requires calculation of

the modulus of complex numbers (complex electric fields), with the modulus calculation process being a nonlinear activation function, it is interesting that the PAIM can be equivalent to a linear neural network connected to a nonlinear activation function and can handle some nonlinear classification problems such as exclusive-OR classification problems (Supplementary Note 2). Nevertheless, technically speaking, the PAIM is still a linear system from the viewpoint of photonic neuromorphic computing. We also note that nonlinear modulation of the PAIM in the complex domain could be explored
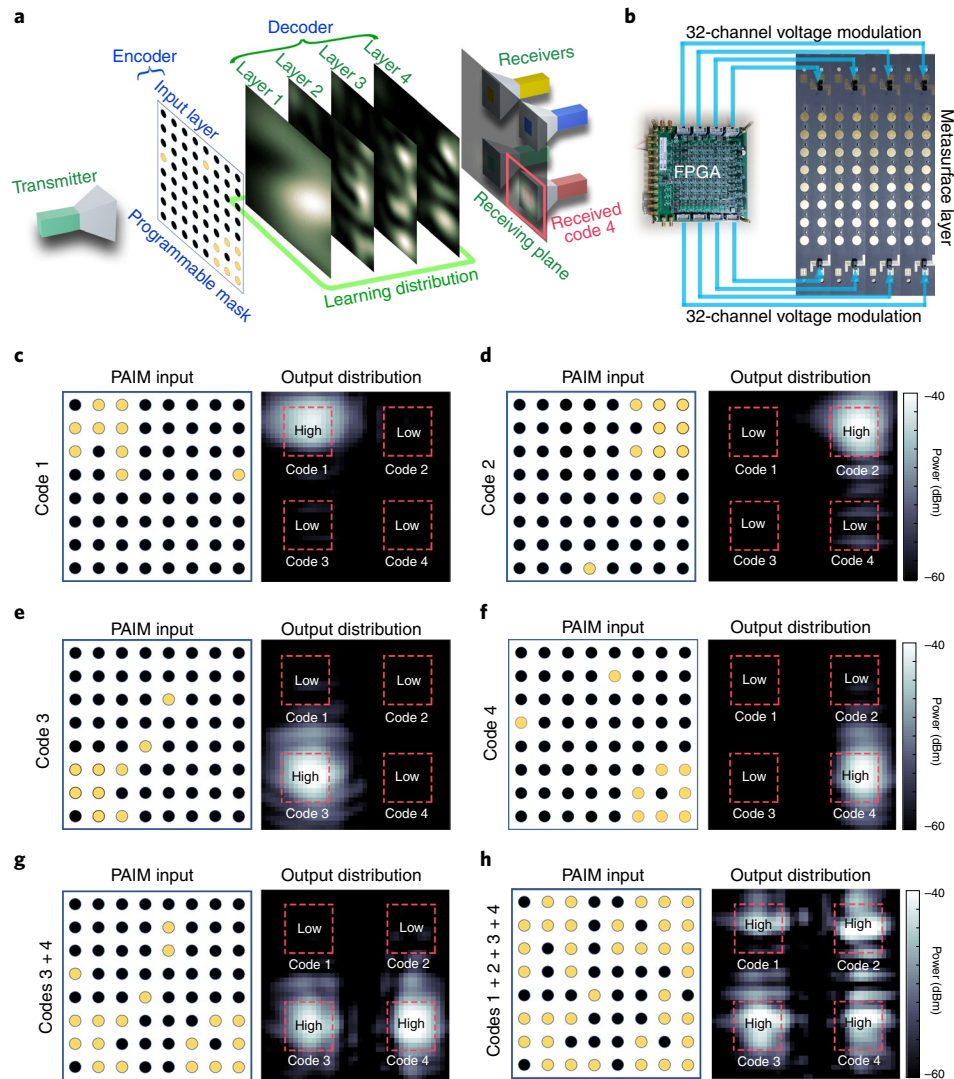
Fig. 4 | Experimental results for the encoder and decoder in the CDMA task using the PAIM. a, The energy distributions of all layers when radiating the coding information of the fourth user. The input layer acts as an encoder to transform the user code to the energy distribution in free space. The yellow and black dots in the input layer represent the binary digits '1' and '0', respectively. The PAIM receives the spatial energy distribution and decodes it via four metasurface layers to judge which user code has been transmitted. The distance between the input layer and the receiving plane (output layer) is 1.6 m. **b**, Photograph of one of the fabricated metasurface layers, which is controlled by 64 channel voltages with the corresponding FPGA. **c–f**, The energy distributions of user codes 1 to 4 (**c–f**, respectively) on the output plane when transmitting one user code, showing that only the correct receiver corresponding to the transmitted user code can collect high energy. **g**,**h**, The output energy distributions when simultaneously transmitting two (3 and 4, **g**) or four (1, 2, 3 and 4, **h**) user codes. Only the receivers corresponding to the transmitted user codes collect high energies, indicating that all user codes could be decoded correctly and simultaneously.

in the future by making the artificial neurons work in the nonlinear range[40], which has the potential to further increase the fitting quality of our PAIM system.

When processing the given data, we make the first-layer programmable metasurface a digital-to-analogue converter to modulate the given data into an amplitude distribution of the EM wave when illuminated by plane waves (Fig. 1a,b). The transmitted EM waves then carry the information of the given data and will be processed by the remaining metasurface layers. Therefore, without using independent and complicated input modules, the PAIM is more flexible and compact than all-optical D²NN platforms[16,17]. In fact, the PAIM can directly receive and process the EM waves radiated by radars, communication base stations and wireless routers, making it more environmentally compatible. (The relationship between the PAIM and current digital hardware is discussed in Supplementary Note 3.)

Image classification tasks based on the PAIM. To verify the powerful capabilities of the PAIM, we first used it to deal with two image classification tasks: oil-painting (Fig. 2) and handwritten-digit (Supplementary Fig. 2) classifications. In the first classification task, using two kinds of oil painting (portraiture and landscape painting) we simulated the PAIM with six-layer metasurfaces, each consisting of 25 × 25 programmable artificial neurons. The input image was greyed and reshaped to 25 × 25 pixels (corresponding to the size of the metasurface) and then input to the PAIM by configuring the first-layer metasurface, in which the transmission coefficient of each artificial neuron was set as the corresponding pixel value of the image. Thus, the EM wave would carry the information of the input image when going through the first layer. The remaining five layers constituted the recognition network. At the end of the PAIM we assigned two receivers to receive the two kinds of oil painting.

The receiving energy at each receiver represents the level of possibility that the input image has been classified. The receiver with the maximum receiving energy corresponds to the kind of classification result (Fig. 2). After training with 500 oil-painting images and testing with 100 images, the mean accuracy rate of recognizing the two oil painting styles was 97%. The number of trainable parameters was only $5 \times 25 \times 25$, much fewer than that required in traditional fully connected ANN networks. A similar architecture was used for the handwritten-digit classification, with a layer size of 40×40 and with the handwritten digits classified into ten different kinds; this reached a 90.76% classification accuracy after training (Supplementary Fig. 2).

To demonstrate the versatility of the PAIM for use in the real world, we designed and fabricated a PAIM sample with five-layer programmable metasurfaces controlled by five FPGA modules, with each layer consisting of 8×8 artificial neurons (Figs. 1a and 4b). Each artificial neuron, integrated with two amplifiers (Supplementary Fig. 3), can individually modulate more than 500 different grades of transmitted gain of the EM wave (Supplementary Fig. 4) under the control of bias voltage by means of FPGAs. We use the programmable transmitted gain to represent the network weight in the ANN, and hence the artificial neuron can be regarded as a dynamic neuron in the fully connected network. The support structure of the PAIM sample is presented in Supplementary Fig. 5; the first layer (that is, the input layer) is illuminated by microwaves at 5.4 GHz radiated by a horn antenna. The measurement was performed in a standard microwave chamber (Supplementary Fig. 6c,d).

To test the real experimental performance of the PAIM in image classification, we designed two example cases, to classify simple patterns and to recognize binary images of four digits (1 to 4). As mentioned above, the first-layer metasurface of the PAIM acts as a digital-to-analogue converter to convert the input image into the corresponding spatial distribution of EM waves. More specifically, different pixel values in the input image correspond to different transmission coefficients of the artificial neurons in the first layer, radiating the EM waves with different spatial distributions onto the second layer. The remaining four layers act as the recognizer, and several receiving antennas are placed at the end of the PAIM. The training process is run in a computer to obtain the appropriate transmission coefficients for the artificial neurons in the recognizer. A gradient backpropagation algorithm could be used to train the PAIM. However, because the adjustable parameters of the digital coding metasurfaces are discrete, we specially designed a discrete optimization algorithm (see Discrete optimization algorithm section) to make the training results more practical. Meanwhile, we calibrate the wave propagation coefficients between two adjacent metasurface layers using a gradient descent method to make our training results more precise. (More details on the calibration approach are provided in Supplementary Note 11.)

For our first experimental case, image classification, we used two simple patterns, the letter 'I' and brackets '[ ]'. The positions of the patterns could be altered to make the input images more varied. The pixels belonging to the pattern areas and background were allocated different bias voltages of the artificial neurons in the first PAIM layer. The experimental results show that the PAIM can classify the two patterns with an accuracy of 100% (Fig. 3a–d and Supplementary Fig. 7). In the second case, digit recognition, we chose binary images of four digits (1 to 4). We discretized the original image into an 8×8 binary pixel matrix and used different bias voltages of the artificial neurons in the first PAIM layer to represent different pixel values. The bias-voltage configurations of layers 2–5 (corresponding to the recognition part of the PAIM) for this case are shown in Fig. 3I. A recognition accuracy of 100% was also achieved for these experimental tests (Fig. 3e–h). Besides the aforementioned two image classification cases, an extra case (game
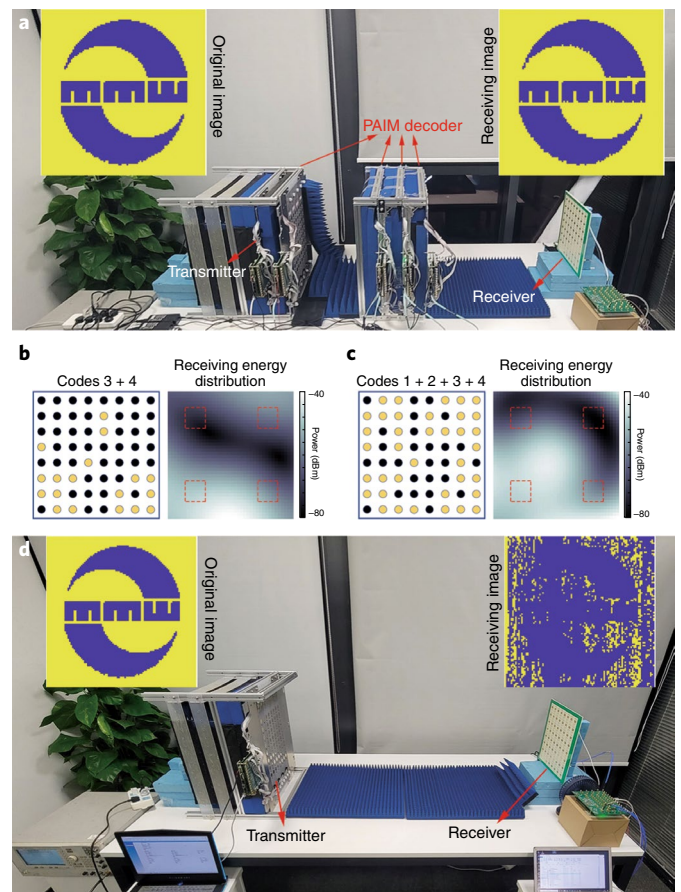

Fig. 5 | Space–time telecommunication system with and without the decoding part of PAIM. a, The space–time telecommunication system with the decoding part of the PAIM. The first layer of the PAIM acts as a transmitter, radiating space–time EM waves at 5.5 GHz. The remaining four layers of the PAIM are trained as a denoising and decoding processor located in the transmission channel. The EM environment is fouled by a commercial 5G Wi-Fi router. However, with the help of the PAIM decoder, the transmission error rate declines to 0.52% along with no extra time delay for signal preprocessing. **b**, The energy distribution on the receiving plane when code 3 and code 4 are transmitted into free space without the decoding part of the PAIM. **c**, The energy distribution on the receiving plane when all four codes are transmitted into free space without the decoding parts of the PAIM. **d**, The space–time telecommunication system without the decoding part of the PAIM. The first layer of the PAIM acts as a transmitter, radiating space–time EM waves (according to four user codes) at 5.5 GHz, which reach the receiver through free-space propagation. The EM environment is fouled by a commercial 5G Wi-Fi router, causing a high error rate of transmission (49.02%) when transmitting a binary image of our laboratory's emblem.

props recognition), designed to demonstrate the ability of the PAIM to recognize colourful images, is provided in Supplementary Note 8 (Supplementary Fig. 8).

Code division multiple access based on PAIM. Besides image classification, we used the same PAIM for a mobile communication codec, which can perform coding and decoding tasks in the code division multiple access (CDMA) scheme and transmit four kinds of orthogonal user code simultaneously or separately in one channel. Each user code is a string of binary numbers with length of 64. As shown in Fig. 4a, the first-layer PAIM metasurface is set as an encoder, on which each artificial neuron sequentially corresponds
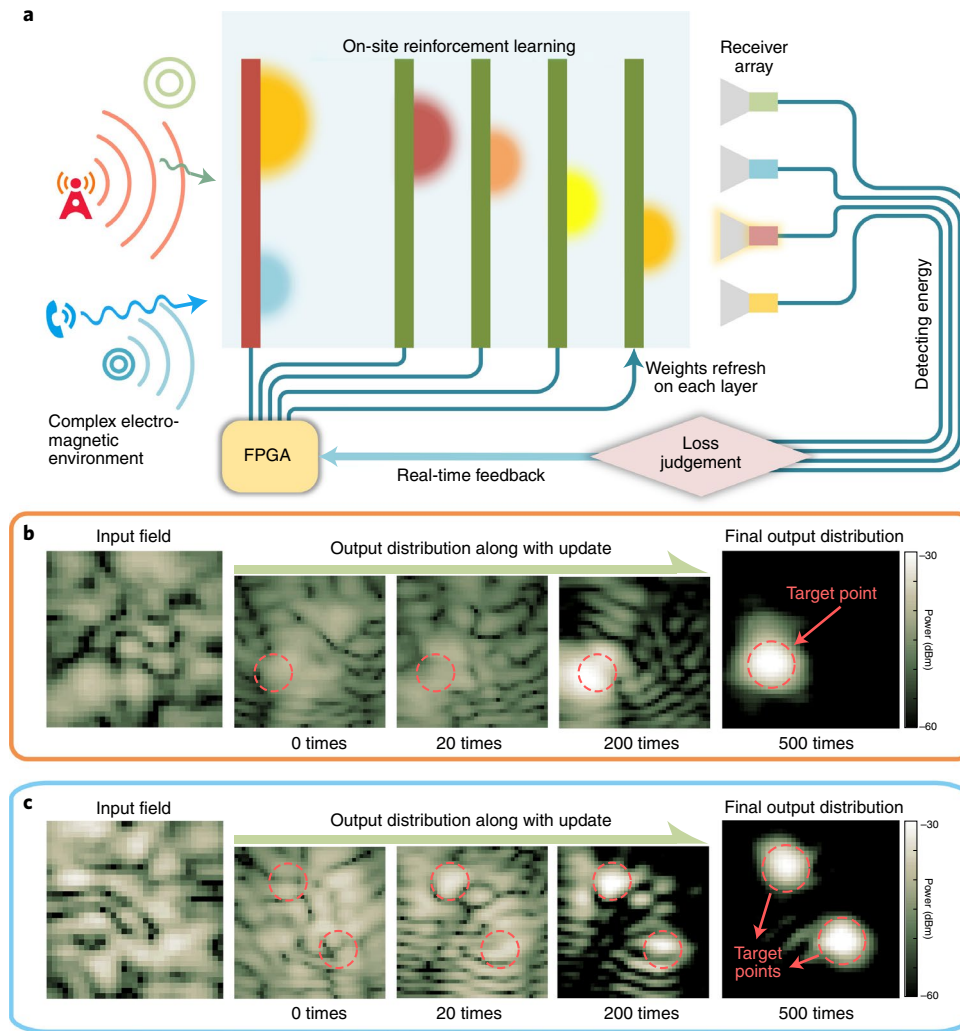
Fig. 6 | Experimental results of dynamic multi-beam focusing by the on-site reinforcement learning process using the PAIM. a, The on-site reinforcement learning process of PAIM, in which the transmission coefficients of each PAIM layer are continually controlled by an FPGA according to real-time feedback signals. **b,c**, The evolution of the output energy distributions along with the updated time in the reinforcement learning process. Here, the distributions of input fields in **b** and **c** are randomly generated but remained unchanged during the reinforcement learning process. We observe that the energies of the output fields are gradually focused on the target points when the updating procedure goes on.

to one bit in the binary number string. When a high or low bias voltage is sent to the artificial neuron, it will correspond to the '1' or '0' bit in the 64-bit user code, respectively. We located four receiving antennas at the end of the PAIM, and each antenna represents a user code. When one of the antennas receives high-energy EM waves, this means that the corresponding user code related to this antenna is transmitted (Fig. 4a).

We use $\{C_1, C_2, C_3, C_4\}$ to represent the four user codes and $\{E_1, E_2, E_3, E_4\}$ to represent the receiving energies at the corresponding antennas. The remaining four-layer PAIM metasurfaces are trained as a decoder. When C_1 is transmitted by the first layer, the values of $\{E_1, E_2, E_3, E_4\}$ will be $f(C_1) = \{\text{high}, \text{low}, \text{low}, \text{low}\}$, in which the function f represents the linear forward propagation function of the PAIM, and the term 'low' indicates that the receiving energy is much less than that of 'high'. Similarly, when C_3 is transmitted, the receiving energy values would be $f(C_3) = \{\text{low}, \text{low}, \text{high}, \text{low}\}$. In a more complicated situation, when two user codes C_1 and C_3 are transmitted simultaneously, the receiving energy values will be $f(C_1 + C_3) = f(C_1) + f(C_3) = \{\text{high}, \text{low}, \text{high}, \text{low}\}$. The same is true when three or four user codes are transmitted simultaneously. Owing to the independence of wave propagation, the transmission

of one user code has little influence on the transmission of others, and hence each user code can be transmitted independently in one channel.

This property allows simplification of the training process: we do not need to train all combinations of the four user codes. Instead, when the output EM wave distribution of each user code conforms to the designed distribution, the combinations will satisfy the expectation automatically. The total loss function for training is the sum of mean square errors (m.s.e.s) between the designed output energy distribution and the one generated by inputting each of the four user codes. Random Gaussian noise is added to the training input to make our system more robust. The experimental results show that our PAIM can decode the user codes correctly even when they are overlapped (Fig. 4c–h and Supplementary Fig. 9).

To demonstrate the strong capability of the PAIM for space–time telecommunications, we used it to transmit a picture of our laboratory's emblem as 100×100 pixels, which could be represented as a binary sequence of 10,000 bits. The receiver was a patch antenna array that could detect the energy at each receiving domain corresponding to the relevant user code (receiving user domain). The modulation scheme for communication is amplitude modulation.

In detail, when the receiving user domain has high energy at the current clock, the transmitted binary signal is '1', and vice versa. Because we have four user codes corresponding to four user channels, we can transmit four images simultaneously. Equivalently, in this case, we chose to use different user channels to transmit different parts of the binary image, making the transmission speed four times faster than when using only one channel. Working at 5.5 GHz, our experimental communication system was placed in a noisy environment fouled by a commercial 5G Wi-Fi router. The distance between the transmitter and receiver was 1.6 m. We first performed a contrast experiment in which we let the space–time amplitude modulation signal (modulated by a single layer of PAIM) propagate in free space (Fig. 5b,c). The received decoding image is shown in Fig. 5d. The error rate of transmission is 49.02%, making the received image indecipherable.

We then trained the remaining four layers of the PAIM as a denoising and decoding processor and located it in the transmission channel, as shown in Fig. 5a. In this case, the received image becomes legible, with an error rate of 0.52%, verifying that the PAIM can directly denoise and decode the information in wave space during the EM propagation process, with nearly no extra time delay. The speed of transmission is 1,000 bit s⁻¹. However, we remark that the transmission speed is only determined by the code-switching rate of the transmitting terminal and the signal-sampling rate of the receiving terminal, which could be remarkably accelerated by using mature communication devices. After training, the configuration parameters of the denoising and decoding parts of the PAIM are constants, meaning that its denoising and decoding functions still work even when the transmission speed is extremely high.

Unlike the traditional CDMA scheme, the PAIM performs the encoding, decoding and denoising functions directly in the wave domain (Supplementary Fig. 10), so the PAIM has the advantage of reducing the time delay in wireless communications. Furthermore, its strong capability to process distributed space EM waves makes the PAIM a good candidate for realizing space division multiplexing and thus increase channel capacity. The decoding function of the PAIM is operated as an independent system and is able to deal with the signals from distributed communication base stations (Supplementary Fig. 11). As spectrum resources have become exhausted, the space division multiplexing technique has received increasing attention and is becoming one of the key technologies in the fifth and sixth generations of wireless communications[41].

Reinforcement learning function of PAIM. Finally, we turned our PAIM into a dynamic multi-beam focusing lens that could focus the EM energy onto multiple points with arbitrary positions. Unlike the aforementioned cases, in which the training process is executed on a computer in advance, here we directly perform on-site training of PAIM using a reinforcement learning method in real time, which completely overcomes the limitation of requiring prior knowledge as in previous optical D²NN platforms[16–19]. Reinforcement learning based on integrated photonic platforms has been demonstrated using a reflected spatial light modulator to manipulate training weights, and has successfully predicted the chaotic Mackey–Glass sequence[42]. Similarly, benefitting from the real-time programmable ability of the PAIM, we can train the parameters through continuous interaction with the unknown and complicated EM environments. Figure 6a presents a schematic diagram of the reinforcement learning process, in which the bias voltages of artificial neurons are randomly changed and controlled by an FPGA. The same FPGA also receives feedback signals from the receivers and calculates the trend of the error function to determine whether the change in bias voltages is reserved or eliminated. In this case, the m.s.e. is again used as the error function, and an extra function is added to restrain the redundant EM waves (see Reinforcement-learning process section). This could be regarded as a real-time optimization

procedure, and the optimal objective is minimizing the distance between the desired pattern and the EM wave distribution generated by the PAIM. The output EM wave distributions along with the updated times of parameters are presented in Fig. 6b,c. After automatic training, the PAIM can transform the output EM waves into target point(s) with more than 90% concentrated energy, and the focusing point(s) can shift with a frequency of 100 Hz to realize multi-beam scanning. In this application, by benefitting from the real-time updates of parameters in the PAIM, no extra training dataset is needed. The experimental results indicate that the PAIM could be applied in different kinds of EM environment.

Conclusions

Our PAIM is an on-site programmable D²NN platform that operates via real-time control of EM waves and can perform computations based on the parallelism of EM wave propagations at the speed of light. It can deal with traditional deep learning tasks such as image recognition and feature detection, and can also provide an on-site way to manipulate spatial EM waves for use in multi-channel encoding and decoding in the CDMA scheme and dynamic multi-beam focusing. Our PAIM could thus be of use in applications such as wireless communications, signal enhancement, medical imaging, remote control and the Internet of Things. The nonlinear version of the PAIM could also be developed further by introducing stable nonlinear amplifiers in the artificial neurons, potentially leading to a variety of new applications.

Methods

Forward propagation model. Figure 1b demonstrates the 2D structure of the PAIM model, and a more detailed version is provided in Supplementary Fig. 1. We use E^i, $i = 0, 1, 2, …, M$ to represent the complex electric field illuminating on the ith PAIM layer, which is an N-dimensional vector (N is the number of meta-atoms in a layer), and each element indicates the field received by the corresponding meta-atom. E^{M+1} is the complex field on the output plane (output layer). Thus, the length of E^{M+1} depends on the number of receiving antennas or the sampling numbers of the moving probe. T^i, $i = 0, 1, 2, …, M$ represents the complex transmission coefficients of the ith layer, which is also an N-dimensional vector, and each element in this vector corresponds to the complex transmission coefficient of the meta-atom in the ith layer. Then the forward propagation formula can be written as

$$E^{i+1} = W^i \left(E^i \odot T^i \right), \quad i = 0, 1, 2, …, M \tag{1}$$

in which $\odot$ is the Hadamard product and W^i represents the space attenuation coefficients from the ith layer to the $(i + 1)$th layer. In fact, W^i, $i = 0, 1, 2, …, M$ is an $N \times N$-dimensional matrix, and its element in the pth row and qth column represents the space attenuation coefficient from the qth meta-atom in the ith layer to the pth meta-atom in the $(i + 1)$th layer. W^M connects the last PAIM layer and the output layer, and hence it is a $K \times N$ matrix, where K is the number of receiving antennas or the sampling number of a moving probe. A schematic diagram of EM-wave propagations between adjacent layers and the radiation pattern of one meta-atom are demonstrated in Fig. 1d,e. The symbols W_n, $n = 1, 2, 3, 4, …$ in Fig. 1e are different from the matrix W^i in equation (1). In fact, W_n, $n = 1, 2, 3, 4, …$ constitute one of the columns in W^i. Using the hierarchical equation (1), we obtain the final output field E^{M+1}, once the input field E^0 is given.

Error backpropagation. Equation (1) presents the forward propagation model of PAIM, which is a linear model. We use a^i and Φ^i to represent the amplitude and phase parts of T^i, $i = 0, 1, 2, …, M$, respectively. The gradients of a^{M-L} and Φ^{M-L}, $L = 0, 1, 2, …, M$ can be easily obtained by matrix multiply operations using the chain rule for the derivative:

$$\frac{\partial E^{M+1}}{\partial \Phi^{M-L}} = \frac{\partial E^{M+1}}{\partial T^{M-L}} \frac{\partial T^{M-L}}{\partial \Phi^{M-L}} = A^{M-L} \mathrm{diag} \left(j T^{M-L} \right)$$

$$\frac{\partial E^{M+1}}{\partial a^{M-L}} = \frac{\partial E^{M+1}}{\partial T^{M-L}} \frac{\partial T^{M-L}}{\partial a^{M-L}} = A^{M-L} \mathrm{diag} \left(\exp \left(j \Phi^{M-L} \right) \right)$$

$$A^{M-L} = W^M \mathrm{diag} \left(T^M \right) W^{M-1} \mathrm{diag} \left(T^{M-1} \right) W^{M-2} \mathrm{diag} \left(T^{M-2} \right) … W^{M-L} \mathrm{diag}(E^{M-L}) \tag{2}$$

The matrix partial derivatives in equation (2) apply for the numerator format, and diag means matrix diagonalization. For different tasks, we define different error functions, symbolized by $\mathrm{Error}(E^{M+1})$. For the classification tasks we use

the cross-entropy error function, and for other tasks we use the m.s.e. function. Applying the chain rule, the final gradient formula can be written as

$$\frac{\partial \text{Error}(E^{M+1})}{\partial a^{M-L}} = \frac{\partial \text{Error}(E^{M+1})}{\partial E^{M+1}} \frac{\partial E^{M+1}}{\partial a^{M-L}} \tag{3}$$

$$\frac{\partial \text{Error}(E^{M+1})}{\partial \Phi^{M-L}} = \frac{\partial \text{Error}(E^{M+1})}{\partial E^{M+1}} \frac{\partial E^{M+1}}{\partial \Phi^{M-L}} \tag{4}$$

When all gradients are calculated, the method of gradient descent is used to optimize a^i or Φ^i independently or simultaneously. Various kinds of deep learning optimizer can be used, such as Adam[43].

Discrete optimization algorithm. In practice, the complex transmission coefficient of the meta-atom cannot be controlled continuously, and can only be set by several discrete values. Thus, it is necessary to develop a discrete optimization algorithm to optimize the complex transmission coefficients of the meta-atom. The aim of the discrete optimization algorithm is the same as the aforementioned gradient descent optimization algorithm: minimizing the error function. We use a_p^i, $i = 1, 2, \ldots, M; p = 1, 2, \ldots, N$ to present the complex transmission coefficient of the pth meta-atom in the ith layer, the complex transmission coefficients of which are represented as T^i. The procedure of the optimization algorithm is given as follows (optimizing a_p^i as an example):

- Step 1: Initialize all parameters of T^i, $i = 0, 1, 2, \ldots, M$ by discretizing the uniform random distribution.
- Step 2: Calculate the output electric fields E^{M+1} by using the forward propagation equation (1) and the current error $\text{Error0} = \text{Error}(E^{M+1})$ of the self-defined error function.
- Step 3: Initialize the update step value of a_p^i randomly, expressed by Δd_p^i. Here, Δd_p^i is usually the difference of adjacent discrete values. Because of the linearity of our network, if the value a_p^i increases by Δd_p^i, the output field will increase by $\Delta d_p^i \frac{\partial E^{M+1}}{\partial a_p^i}$. Thus, the output field becomes $E^{M+1} + \Delta d_p^i \frac{\partial E^{M+1}}{\partial a_p^i}$. Calculate the auxiliary error by

$$\text{Error}_p^i = \text{Error}\left(E^{M+1} + \Delta d_p^i \frac{\partial E^{M+1}}{\partial a_p^i}\right) \tag{5}$$

- Step 4: Compare the values of Error0 and Error_p^i. If $\text{Error0} \leq \text{Error}_p^i$, the value of a_p^i remains unchanged; otherwise, the value of a_p^i is updated and replaced by $a_p^i + \Delta d_p^i$.

For all a_p^i, $i = 1, 2, \ldots, M$; $p = 1, 2, \ldots, N$, we execute steps 2 to 4 in loops until the current error is less than the preset value, and then save the corresponding T^i, $i = 0, 1, 2, \ldots, M$ as the final optimization outcome. Several methods could be used to accelerate the optimization process, such as organizing the computational process in matrix form or using parallel algorithms.

Reinforcement-learning process. To fully demonstrate the real-time re-trainable advantage of the PAIM, we specifically designed a reinforcement learning task to realize dynamic multi-beam focusing. The reinforcement learning process does not need pre-prepared training data, but updates the configuration parameters according to feedback by interacting with the environment. It can thus guarantee the training result to be adaptive to the actual application scenario. For the multi-beam focusing task, the error is calculated by

$$\text{Error} = \sum_k y_k \left(s_k - \theta_k\right)^2 + aS^2 \tag{6}$$

where k represents the number of points on which we want to concentrate the EM energy, s_k and θ_k indicate the measured energy and desired energy at the kth point, respectively, S represents the total leaking energy (which is the sum of energies radiated outside the target point) and y_k and a are constant scale factors.

In one iteration of the training process, we randomly choose 20% trainable meta-atoms and change their bias voltages by small random values, one by one. Instantly after the bias voltage of a meta-atom is changed, the change in output energy distribution is measured by the receiving antennas, and, at the same time, the change in error is calculated by equation (6), as shown in Fig. 1c. If the error decreases, which means that the bias voltage change of the meta-atom can make the current output EM distribution closer to the desired one, the current bias voltage will be reserved, otherwise the bias voltage will restore to the value before it was just changed. In our multi-beam focusing task, the PAIM could focus the EM energies to the desired positions after ~500 iterations.

One advantage of reinforcement learning is the result-oriented strategy, with which we do not need to worry about the accuracy of simulations or other factors that could make the designed parameters deviate from the measurement results. Hence, it enables the PAIM to deal with very complicated and unknown environments, broadening its application range.

References
1. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
2. Huang, G., Huang, G. B., Song, S. J. & You, K. Y. Trends in extreme learning machines: a review. *Neural Netw.* **61**, 32–48 (2015).
3. Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
4. Ren, H. R., Li, X. P., Zhang, Q. M. & Gu, M. On-chip noninterference angular momentum multiplexing of broadband light. *Science* **352**, 805–809 (2016).
5. Shen, Y. et al. Deep learning with coherent nanophotonic circuits. *Nat. Photon.* **11**, 441–446 (2017).
6. Hughes, T. W., Minkov, M., Shi, Y. & Fan, S. Training of photonic neural networks through in situ backpropagation and gradient measurement. *Optica* **5**, 864–871 (2018).
7. Feldmann, J., Youngblood, N., Wright, C. D., Bhaskaran, H. & Pernice, W. H. P. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature* **569**, 208–214 (2019).
8. Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S. & Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 210–227 (2009).
9. Ding, C., Choi, J., Tao, D. & Davis, L. S. Multi-directional multi-level dual-cross patterns for robust face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**, 518–531 (2016).
10. Vazifeh, M. M., Santi, P., Resta, G., Strogatz, S. H. & Ratti, C. Addressing the minimum fleet problem in on-demand urban mobility. *Nature* **557**, 534–538 (2018).
11. Li, W. et al. AADS: augmented autonomous driving simulation using data-driven algorithms. *Sci. Robot.* **4**, eaaw0863 (2019).
12. Palangi, H. et al. Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval. *IEEE ACM Trans. Audio Speech Lang. Process.* **24**, 694–707 (2016).
13. Young, T., Hazarika, D., Poria, S. & Cambria, E. Recent trends in deep learning based natural language processing. *IEEE Comput. Intell. Mag.* **13**, 55–75 (2018).
14. He, J. et al. The practical implementation of artificial intelligence technologies in medicine. *Nat. Med.* **25**, 30–36 (2019).
15. Haque, A., Milstein, A. & Fei-Fei, L. Illuminating the dark spaces of healthcare with ambient intelligence. *Nature* **585**, 193–202 (2020).
16. Lin, X. et al. All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004–1008 (2018).
17. Khoram, E. et al. Nanophotonic media for artificial neural inference. *Photon. Res.* **7**, 823–827 (2019).
18. Zuo, Y. et al. All-optical neural network with nonlinear activation functions. *Optica* **6**, 1132–1137 (2019).
19. Zhou, T. et al. Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit. *Nat. Photon* **15**, 367–373 (2021).
20. Cui, T. J., Qi, M. Q., Wan, X., Zhao, J. & Cheng, Q. Coding metamaterials, digital metamaterials and programmable metamaterials. *Light Sci. Appl.* **3**, e218 (2014).
21. Cui, T. J., Liu, S. & Zhang, L. Information metamaterials and metasurfaces. *J. Mater. Chem. C* **5**, 3644–3668 (2017).
22. Pendry, J. B., Luo, Y. & Zhao, R. K. Transforming the optical landscape. *Science* **348**, 521–524 (2015).
23. Qiu, C. W. & Yang, Y. J. Vortex generation reaches a new plateau. *Science* **357**, 645–645 (2017).
24. Hu, G. W. et al. Coherent steering of nonlinear chiral valley photons with a synthetic Au–WS₂ metasurface. *Nat. Photon.* **13**, 467–472 (2019).
25. Li, L. et al. Metalens-array-based high-dimensional and multiphoton quantum source. *Science* **368**, 1487–1490 (2020).
26. Cui, T. J., Li, L., Liu, S., Ma, Q. & Cheng, Q. J. I. Information metamaterial systems. *iScience* **23**, 101403 (2020).
27. Ma, Q. & Cui, T. J. Information metamaterials: bridging the physical world and digital world. *PhotoniX* **1**, 1–32 (2020).
28. Zhao, J. et al. Programmable time-domain digital-coding metasurface for non-linear harmonic manipulation and new wireless communication systems. *Natl Sci. Rev.* **6**, 231–238 (2019).

29. Cui, T. J., Liu, S., Bai, G. D. & Ma, Q. Direct transmission of digital message via programmable coding metasurface. *Research* **2019**, 2584509 (2019).

30. Zhao, H. et al. Metasurface-assisted massive backscatter wireless communication with commodity Wi-Fi signals. *Nat. Commun.* **11**, 3926–3926 (2020).

31. Zhang, L. et al. A wireless communication scheme based on space- and frequency-division multiplexing using digital metasurfaces. *Nat. Electron.* **4**, 218–227 (2021).

32. Li, L. L. et al. Electromagnetic reprogrammable coding-metasurface holograms. *Nat. Commun.* **8**, 197 (2017).

33. Li, L. et al. Intelligent metasurface imager and recognizer. *Light Sci. Appl* **8**, 97 (2019).

34. Zhang, L. et al. Space-time-coding digital metasurfaces. *Nat. Commun.* **9**, 4334 (2018).

35. Ma, Q. et al. Smart metasurface with self-adaptively reprogrammable functions. *Light Sci. Appl.* **8**, 98 (2019).

36. Zhang, X. G. et al. An optically driven digital metasurface for programming electromagnetic functions. *Nat. Electron.* **3**, 165–171 (2020).

37. Ma, Q. et al. Controllable and programmable nonreciprocity based on detachable digital coding metasurface. *Adv. Opt. Mater.* **7**, 1901285 (2019).

38. Chen, L. et al. Space-energy digital-coding metasurface based on an active amplifier. *Phys. Rev. Appl.* **11**, 054051 (2019).

39. Schmidhuber, J. Deep learning in neural networks: an overview. *Neural Netw.* **61**, 85–117 (2015).

40. Ryou, A. et al. Free-space optical neural network based on thermal atomic nonlinearity. *Photon. Res.* **9**, B128–B134 (2021).

41. Zhu, J., Zhao, M., Zhang, S. & Zhou, W. Exploring the road to 6G: ABC—foundation for intelligent mobile networks. *China Commun.* **17**, 51–67 (2020).

42. Bueno, J. et al. Reinforcement learning in a large-scale photonic recurrent neural network. *Optica* **5**, 756–760 (2018).

43. Kingma, D. P. & Ba, J. L. Adam: a method for stochastic optimization. in 3rd International Conference on Learning Representations (2015).

Acknowledgements

Author contributions

T.J.C., C.L., Q.M. and L.L. conceived the research. C.L. and Q.M. designed the PAIM devices and relevant algorithms. C.L., Q.M. and Z.J.L. contributed to the experiments. T.J.C., C.L. and Q.M. prepared the manuscript. T.J.C. initiated and supervised the research. All authors contributed to the data analysis and writing of the manuscript, which was reviewed by all authors.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41928-022-00719-9.

Correspondence and requests for materials should be addressed to Tie Jun Cui.

Peer review information *Nature Electronics* thanks Arka Majumdar and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.