OPTICAL COMPUTING

All-optical machine learning using diffractive deep neural networks

Xing Lin[1,2,3]*, Yair Rivenson[1,2,3]*, Nezih T. Yardimci[1,3], Muhammed Veli[1,2,3], Yi Luo[1,2,3], Mona Jarrahi[1,3], Aydogan Ozcan[1,2,3,4]†

Deep learning has been transforming our ability to execute advanced inference tasks using computers. Here we introduce a physical mechanism to perform machine learning by demonstrating an all-optical diffractive deep neural network (D²NN) architecture that can implement various functions following the deep learning–based design of passive diffractive layers that work collectively. We created 3D-printed D²NNs that implement classification of images of handwritten digits and fashion products, as well as the function of an imaging lens at a terahertz spectrum. Our all-optical deep learning framework can perform, at the speed of light, various complex functions that computer-based neural networks can execute; will find applications in all-optical image analysis, feature detection, and object classification; and will also enable new camera designs and optical components that perform distinctive tasks using D²NNs.

Deep learning is one of the fastest-growing machine learning methods (1). This approach uses multilayered artificial neural networks implemented in a computer to digitally learn data representation and abstraction and to perform advanced tasks in a manner comparable or even superior to the performance of human experts. Recent examples in which deep learning has made major advances in machine learning include medical image analysis (2), speech recognition (3), language translation (4), and image classification (5), among others (1, 6). Beyond some of these mainstream applications, deep learning methods are also being used to solve inverse imaging problems (7–13).

Here we introduce an all-optical deep learning framework in which the neural network is physically formed by multiple layers of diffractive surfaces that work in collaboration to optically perform an arbitrary function that the network can statistically learn. Whereas the inference and prediction mechanism of the physical network is all optical, the learning part that leads to its design is done through a computer. We term this framework a diffractive deep neural network (D²NN) and demonstrate its inference capabilities through both simulations and experiments. Our D²NN can be physically created by using several transmissive and/or reflective layers (14), where each point on a given layer either transmits or reflects the incoming wave, representing an artificial neuron that is connected to other neurons of the following layers through optical diffraction (Fig. 1A). In accordance with the Huygens-Fresnel principle, our terminology is

based on each point on a given layer acting as a secondary source of a wave, the amplitude and phase of which are determined by the product of the input wave and the complex-valued transmission or reflection coefficient at that point [see (14) for an analysis of the waves within a D²NN]. Therefore, an artificial neuron in a D²NN is connected to other neurons of the following layer through a secondary wave modulated in amplitude and phase by both the input interference pattern created by the earlier layers and the local transmission or reflection coefficient at that point. As an analogy to standard deep neural networks (Fig. 1D), one can consider the transmission or reflection coefficient of each point or neuron as a multiplicative "bias" term, which is a learnable network parameter that is iteratively adjusted during the training process of the diffractive network, using an error back-propagation method. After this numerical training phase, the D²NN design is fixed and the transmission or reflection coefficients of the neurons of all layers are determined. This D²NN design—once physically fabricated using techniques such as 3D-printing or lithography—can then perform, at the speed of light, the specific task for which it is trained, using only optical diffraction and passive optical components or layers that do not need power, thereby creating an efficient and fast way of implementing machine learning tasks.

In general, the phase and amplitude of each neuron can be learnable parameters, providing a complex-valued modulation at each layer, which improves the inference performance of the diffractive network (fig. S1) (14). For coherent transmissive networks with phase-only modulation, each layer can be approximated as a thin optical element (Fig. 1). Through deep learning, the phase values of the neurons of each layer of the diffractive network are iteratively adjusted (trained) to perform a specific function by feeding training data at the input layer and then computing the network's output through optical

diffraction. On the basis of the calculated error with respect to the target output, determined by the desired function, the network structure and its neuron phase values are optimized via an error back-propagation algorithm, which is based on the stochastic gradient descent approach used in conventional deep learning (14).

To demonstrate the performance of the D²NN framework, we first trained it as a digit classifier to perform automated classification of handwritten digits, from 0 to 9 (Figs. 1B and 2A). For this task, phase-only transmission masks were designed by training a five-layer D²NN with 55,000 images (5000 validation images) from the MNIST (Modified National Institute of Standards and Technology) handwritten digit database (15). Input digits were encoded into the amplitude of the input field to the D²NN, and the diffractive network was trained to map input digits into 10 detector regions, one for each digit. The classification criterion was to find the detector with the maximum optical signal, and this was also used as a loss function during the network training (14).

After training, the design of the D²NN digit classifier was numerically tested using 10,000 images from the MNIST test dataset (which were not used as part of the training or validation image sets) and achieved a classification accuracy of 91.75% (Fig. 3C and fig. S1). In addition to the classification performance of the diffractive network, we also analyzed the energy distribution observed at the network output plane for the same 10,000 test digits (Fig. 3C), the results of which clearly demonstrate that the diffractive network learned to focus the input energy of each handwritten digit into the correct (i.e., the target) detector region, in accord with its training. With the use of complex-valued modulation and increasing numbers of layers, neurons, and connections in the diffractive network, our classification accuracy can be further improved (figs. S1 and S2). For example, fig. S2 demonstrates a Lego-like physical transfer learning behavior for D²NN framework, where the inference performance of an already existing D²NN can be further improved by adding new diffractive layers—or, in some cases, by peeling off (i.e., discarding) some of the existing layers—where the new layers to be added are trained for improved inference (coming from the entire diffractive network: old and new layers). By using a patch of two layers added to an existing and fixed D²NN design (N = 5 layers), we improved our MNIST classification accuracy to 93.39% (fig. S2) (14); the state-of-the-art convolutional neural network performance has been reported as 99.60 to 99.77% (16–18). More discussion on reconfiguring D²NN designs is provided in the supplementary materials (14).

Following these numerical results, we 3D-printed our five-layer D²NN design (Fig. 2A), with each layer having an area of 8 cm by 8 cm, followed by 10 detector regions defined at the output plane of the diffractive network (Figs. 1B and 3A). We then used continuous-wave illumination at 0.4 THz to test the network's inference performance (Figs. 2, C and D). Phase values of

[1]Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095, USA. [2]Department of Bioengineering, University of California, Los Angeles, CA 90095, USA. [3]California NanoSystems Institute (CNSI), University of California, Los Angeles, CA 90095, USA. [4]Department of Surgery, David Geffen School of Medicine, University of California, Los Angeles, CA 90095, USA.
*These authors contributed equally to this work.
†Corresponding author. Email: ozcan@ucla.edu

each layer's neurons were physically encoded using the relative thickness of each 3D-printed neuron. Numerical testing of this five-layer D^{2}NN design achieved a classification accuracy of 91.75% over ~10,000 test images (Fig. 3C). To quantify the match between these numerical testing results and our experiments, we 3D-printed 50 handwritten digits (five different inputs per digit), selected among the same 91.75% of the test images for which numerical testing was successful. For each input object that is uniformly illuminated with the terahertz source, we imaged the output plane of the D^{2}NN to map the intensity distribution for each detector region that is assigned to a digit. The results (Fig. 3B) demonstrate the success of the 3D-printed diffractive neural network and its inference capability: The average intensity distribution at the output plane of the network for each input digit clearly reveals that the 3D-printed D^{2}NN was able to focus the input energy of the beam and achieve a maximum signal at the corresponding detector region assigned for that digit. Despite 3D-printing errors, possible alignment issues, and other experimental error sources in our setup (14), the match between the experimental and numerical testing of our five-layer D^{2}NN design was found to be 88% (Fig. 3B). This relatively small reduction in the perform-

ance of the experimental network compared to our numerical testing is especially pronounced for the digit 0 because it is challenging to 3D-print the large void region at the center of the digit. Similar printing challenges were also observed for other digits that have void regions; e.g., 6, 8, and 9 (Fig. 3B).

Next, we tested the classification performance of D^{2}NN framework with a more complicated image dataset—i.e., the Fashion-MNIST dataset (19), which includes 10 classes, each representing a fashion product (t-shirts, trousers, pullovers, dresses, coats, sandals, shirts, sneakers, bags, and ankle boots; see fig. S3 for sample images). In general, for a coherently illuminated D^{2}NN, we can use the amplitude and/or phase channels of the input plane to represent data to be classified or processed. In our digit classification results reported earlier, input objects were encoded by using the amplitude channel, and to demonstrate the utility of the phase channel of the network input, we encoded each input image corresponding to a fashion product as a phase-only object modulation (14). Our D^{2}NN inference results (as a function of the number of layers, neurons, and connections) for classification of fashion products are summarized in figs. S4 and S5. To provide an example of our performance, a phase-only and a complex-

valued modulation D^{2}NN with $N = 5$ diffractive layers (sharing the same physical network dimensions as the digit classification D^{2}NN shown in Fig. 2A) reached an accuracy of 81.13 and 86.33%, respectively (fig. S4). By increasing the number of diffractive layers to $N = 10$ and the total number of neurons to 0.4 million, our classification accuracy increased to 86.60% (fig. S5). For convolutional neural net–based standard deep learning, the state-of-the-art performance for Fashion-MNIST classification accuracy has been reported as 96.7%, using ~8.9 million learnable parameters and ~2.5 million neurons (20).

To experimentally demonstrate the performance of fashion product classification using a physical D^{2}NN, we 3D-printed our phase-only five-layer design and 50 fashion products used as test objects (five per class) on the basis of the same procedures employed for the digit classification diffractive network (Figs. 2A and 3), except that each input object information was encoded in the phase channel. Our results are summarized in Fig. 4, revealing a 90% match between the experimental and numerical testing of our five-layer D^{2}NN design, with five errors out of 50 fashion products. Compared with digit classification (six errors out of 50 digits; Fig. 3), this experiment yielded a slightly better match
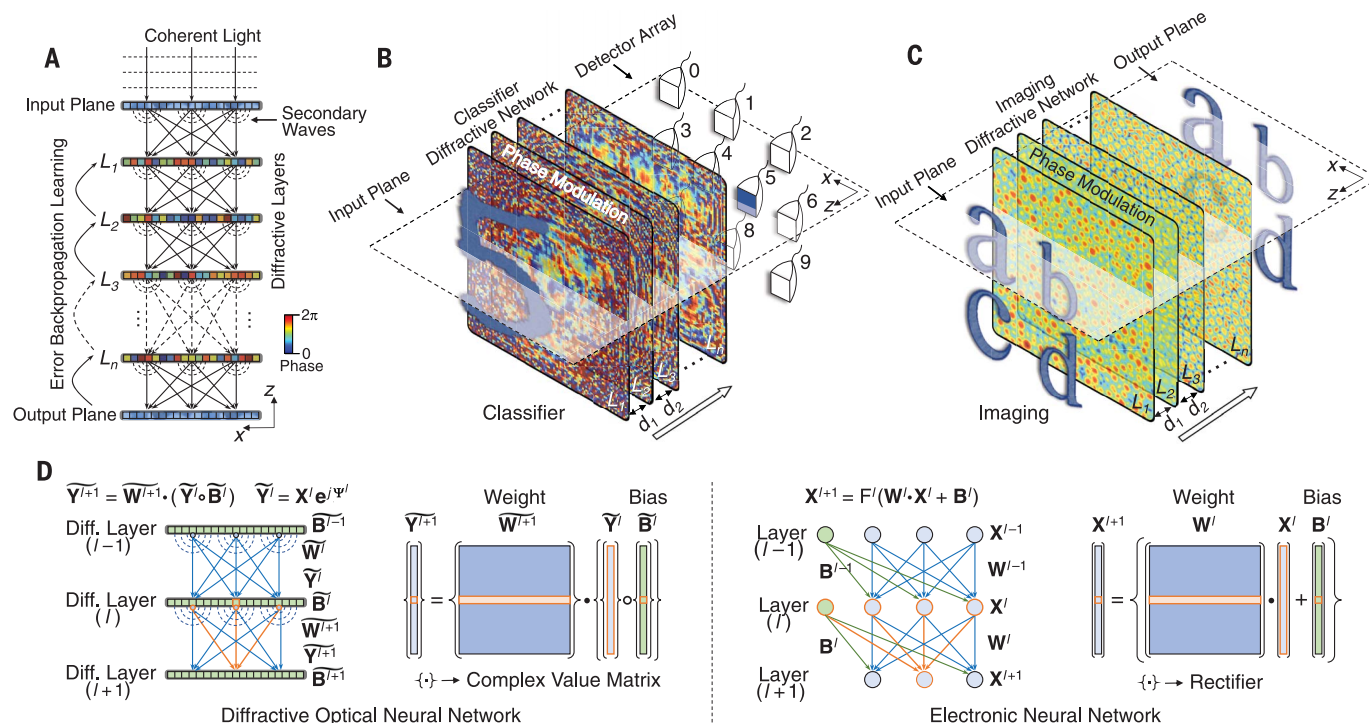
Fig. 1. Diffractive deep neural networks (D^{2}NNs). (A) A D^{2}NN comprises multiple transmissive (or reflective) layers, where each point on a given layer acts as a neuron, with a complex-valued transmission (or reflection) coefficient. The transmission or reflection coefficients of each layer can be trained by using deep learning to perform a function between the input and output planes of the network. After this learning phase, the D^{2}NN design is fixed; once fabricated or 3D-printed, it performs the learned function at the speed of light. L, layer. (**B** and **C**) We trained and experimentally implemented different types of D^{2}NNs: (B) classifier (for handwritten digits and fashion products) and (C) imager.

d, distance. (**D**) Comparison between a D^{2}NN and a conventional neural network (14). Based on coherent waves, the D^{2}NN operates on complex-valued inputs, with multiplicative bias terms. Weights in a D^{2}NN are based on free-space diffraction and determine the interference of the secondary waves that are phase- and/or amplitude-modulated by the previous layers. "o" denotes a Hadamard product operation. "Electronic neural network" refers to the conventional neural network virtually implemented in a computer. **Y**, optical field at a given layer; **Ψ**, phase of the optical field; **X**, amplitude of the optical field; F, nonlinear rectifier function [see (14) for a discussion of optical nonlinearity in D^{2}NN].
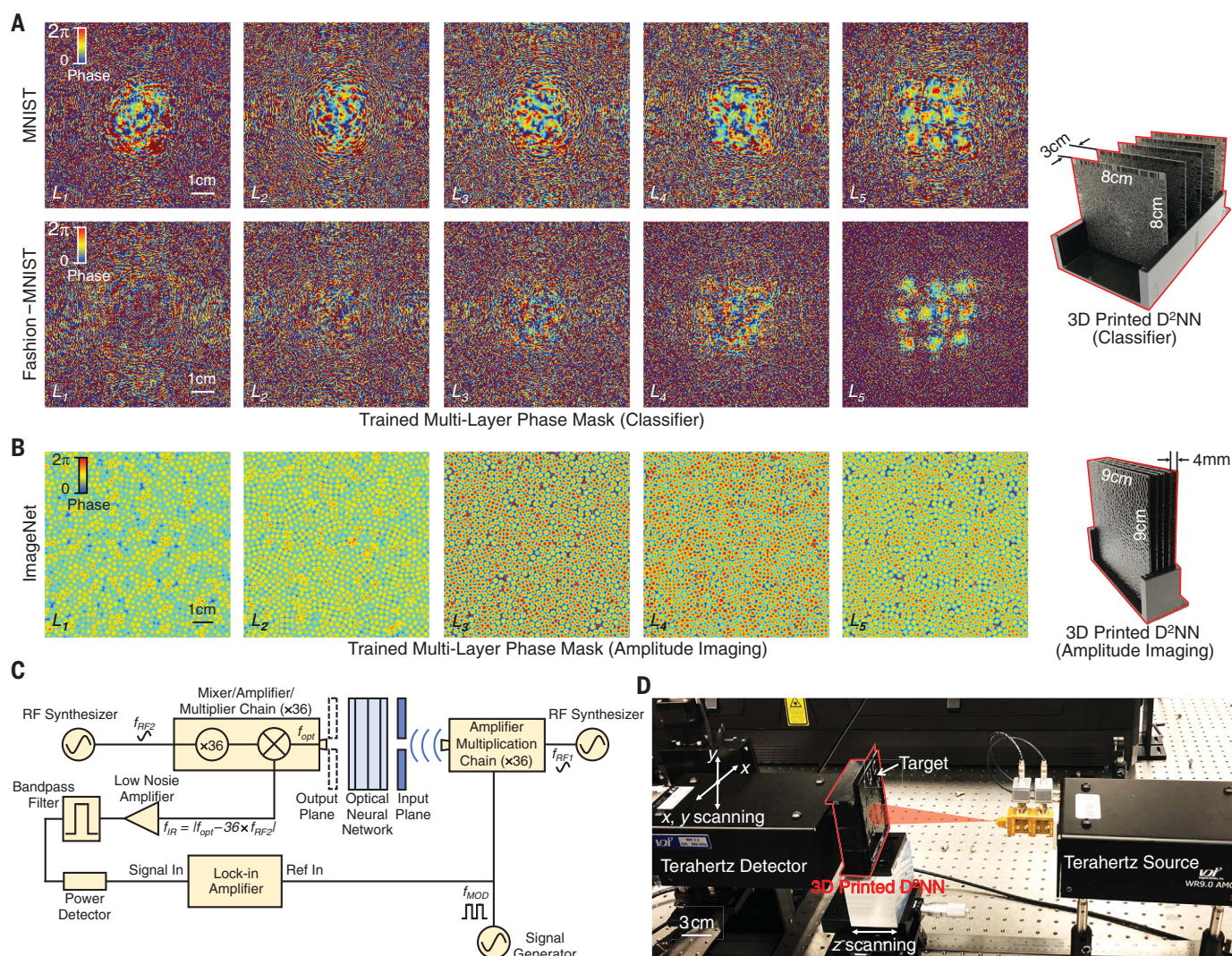
Fig. 2. Experimental testing of 3D-printed D²NNs. (**A** and **B**) After the training phase, the final designs of five different layers (L_1, L_2, ..., L_5) of the handwritten digit classifier, fashion product classifier, and the imager D²NNs are shown. To the right of the network layers, an illustration of the corresponding 3D-printed D²NN is shown. (**C** and **D**) Schematic (C) and photo (D) of the experimental terahertz setup. An amplifier-multiplier chain was used to generate continuous-wave radiation at 0.4 THz, and a mixer-amplifier-multiplier chain was used for the detection at the output plane of the network. RF, radio frequency; f, frequency.

between the experimental and numerical testing results (despite the more challenging nature of Fashion-MNIST dataset), perhaps because we used the phase channel, which does not suffer from the challenges associated with 3D-printing of void regions [such as in digits 0, 6, 8, and 9 (Fig. 3)], to encode input image information for fashion products.
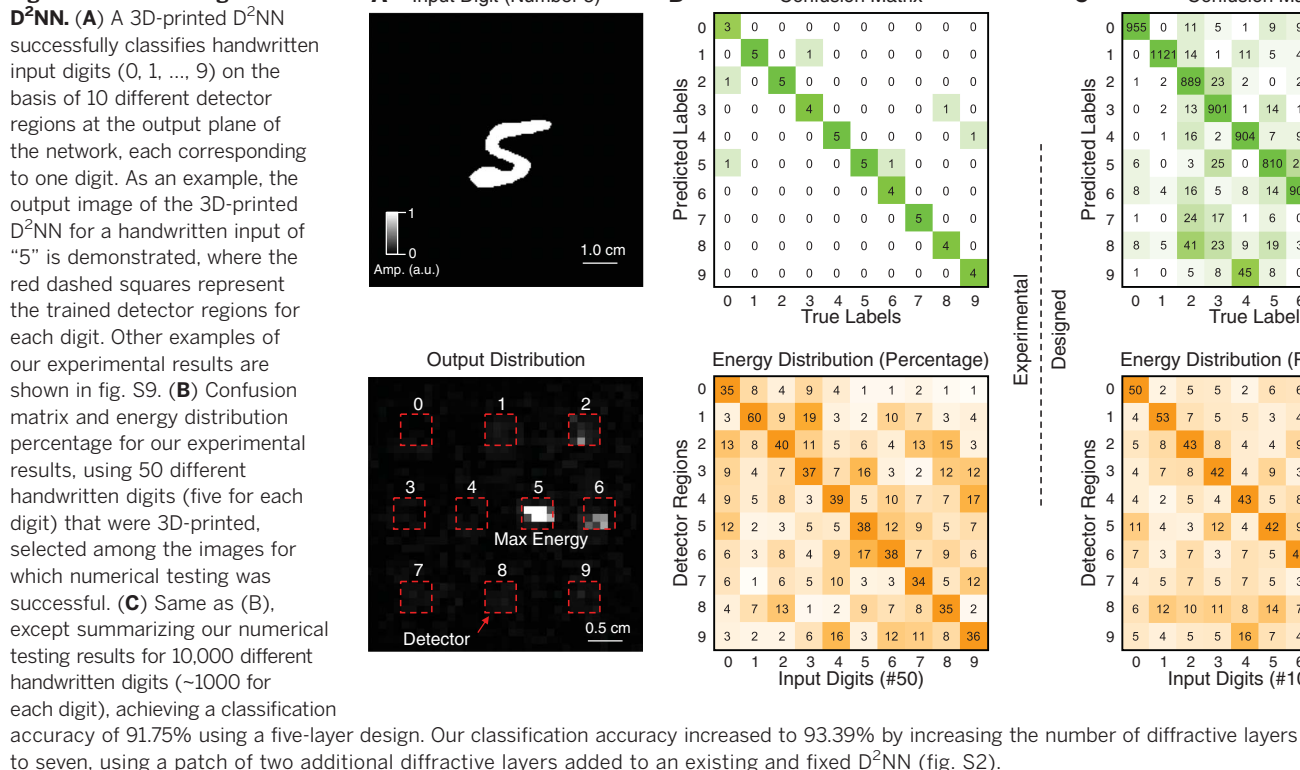
Next, we tested the performance of a phase-only D²NN, composed of five 3D-printed transmission layers to implement amplitude imaging (Fig. 2B). This network was trained using the ImageNet database (21) to create a unit-magnification image of the input optical field amplitude at its output plane (~9 cm by 9 cm)—that is, the output image has the same physical size as the input object (14). As illustrated in fig. S6, A and C, the trained network initially connects every amplitude point at the input plane to various neurons and features of the fol-

lowing layers, which then focus the light back to a point at the output (i.e., image) plane, which is, as expected, quite different than the case of free-space diffraction (i.e., without the presence of the diffractive network), illustrated in fig. S6, B and D.

After training and blind testing, which served to numerically prove the imaging capability of the network (figs. S6 and S7), we then 3D-printed this designed D²NN. Using the same experimental setup shown in Fig. 2, C and D, we imaged the output plane of the 3D-printed D²NN for various input objects that were uniformly illuminated by continuous-wave radiation at 0.4 THz. Figure S8 summarizes our experimental results achieved with this 3D-printed D²NN, which successfully projected unit-magnification images of the input patterns at the output plane of the network, learning the function of an imager, or a physical auto-encoder. To evaluate the point spread

function of this D²NN, we imaged pinholes with different diameters (1, 2, and 3 mm), which resulted in output images, each with a full width at half maximum of 1.5, 1.4, and 2.5 mm, respectively (fig. S8B). Our results also revealed that the printed network can resolve a linewidth of 1.8 mm at 0.4 THz (corresponding to a wavelength of 0.75 mm in air), which is slightly worse in resolution compared with the numerical testing of our D²NN design, where the network could resolve a linewidth of ~1.2 mm (fig. S7C). This experimental degradation in the performance of the diffractive network can be due to factors such as 3D-printing errors, potential misalignments, and absorption-related losses in the 3D-printed network (14).

Optical implementation of machine learning in artificial neural networks is promising because of the parallel computing capability and power efficiency of optical systems (22–24). Compared

Fig. 3. Handwritten digit classifier D²NN. (**A**) A 3D-printed D²NN successfully classifies handwritten input digits (0, 1, ..., 9) on the basis of 10 different detector regions at the output plane of the network, each corresponding to one digit. As an example, the output image of the 3D-printed D²NN for a handwritten input of "5" is demonstrated, where the red dashed squares represent the trained detector regions for each digit. Other examples of our experimental results are shown in fig. S9. (**B**) Confusion matrix and energy distribution percentage for our experimental results, using 50 different handwritten digits (five for each digit) that were 3D-printed, selected among the images for which numerical testing was successful. (**C**) Same as (B), except summarizing our numerical testing results for 10,000 different handwritten digits (~1000 for each digit), achieving a classification

accuracy of 91.75% using a five-layer design. Our classification accuracy increased to 93.39% by increasing the number of diffractive layers to seven, using a patch of two additional diffractive layers added to an existing and fixed D²NN (fig. S2).

Fig. 4. Fashion product classifier D²NN. (**A**) As an example, the output image of the 3D-printed D²NN for a sandal input (Fashion-MNIST class 5) is demonstrated. The red dashed squares represent the trained detector regions for each fashion product. Other examples of our experimental results are shown in fig. S10. (**B**) Confusion matrix and energy distribution percentage for our experimental results, using 50 different fashion products (five per class) that were 3D-printed, selected among the images for which numerical testing was successful. (**C**) Same as (B), except summarizing our numerical testing results for 10,000 different fashion products (~1000 per class), achieving a classification accuracy of 81.13% using a five-layer design. By increasing the number of diffractive layers to 10, our classification accuracy increased to 86.60% (fig. S5).
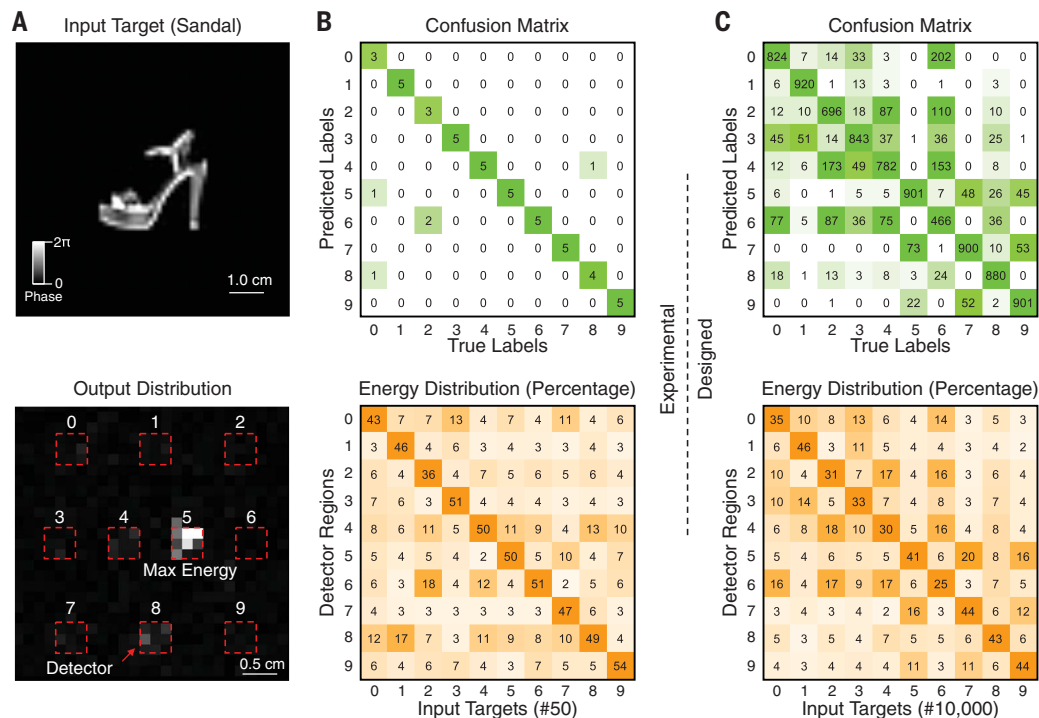
with previous optoelectronics-based learning approaches (22, 25–27), the D²NN framework provides a distinctive all-optical machine learning engine that efficiently operates at the speed of light using passive components and optical diffraction. An important advantage of D²NNs is that they can be easily scaled up using various high-throughput and large-area 3D-fabrication methods (such as soft lithography and additive manufacturing), as well as wide-field optical components and detection systems, to cost-effectively reach tens to hundreds of millions of neurons and hundreds of billions of connections in a scalable and power-efficient manner. For example, integration of D²NNs with lensfree on-chip imaging systems (28, 29) could provide extreme parallelism within a cost-effective and portable platform. Such large-scale D²NNs may

be transformative for various applications, including image analysis, feature detection, and object classification, and may also enable new microscope or camera designs that can perform specific imaging tasks using D^2NNs. To achieve these new technologies, nonlinear optical materials (14) and a monolithic D^2NN design that combines all layers of the network as part of a 3D-fabrication method would be desirable. Among other techniques, laser lithography based on two-photon polymerization (30) can provide solutions for creating such D^2NNs.

REFERENCES AND NOTES

1. Y. LeCun, Y. Bengio, G. Hinton, *Nature* **521**, 436–444 (2015).
2. G. Litjens *et al.*, *Med. Image Anal.* **42**, 60–88 (2017).
3. A. Graves, A. Mohamed, G. Hinton, in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (IEEE, 2013), pp. 6645–6649.
4. K. Cho *et al.*, arXiv:1406.1078 [cs.CL] (3 June 2014).
5. A. Krizhevsky, I. Sutskever, G. E. Hinton, in *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger, Eds. (Curran Associates, 2012), pp. 1097–1105.
6. D. Silver *et al.*, *Nature* **529**, 484–489 (2016).
7. U. S. Kamilov *et al.*, *Optica* **2**, 517 (2015).
8. Y. Rivenson *et al.*, *Optica* **4**, 1437 (2017).
9. K. H. Jin, M. T. McCann, E. Froustey, M. Unser, *IEEE Trans. Image Process.* **26**, 4509–4522 (2017).
10. Y. Rivenson, Y. Zhang, H. Gunaydin, D. Teng, A. Ozcan, *Light Sci. Appl.* **7**, 17141 (2018).
11. A. Sinha, J. Lee, S. Li, G. Barbastathis, *Optica* **4**, 1117 (2017).
12. K. Hammernik *et al.*, *Magn. Reson. Med.* **79**, 3055–3071 (2018).
13. Y. Rivenson *et al.*, *ACS Photonics* **5**, 2354–2364 (2018).
14. Materials and methods are available as supplementary materials.
15. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, *Proc. IEEE* **86**, 2278–2324 (1998).
16. D. Ciregan, U. Meier, J. Schmidhuber, in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2012), pp. 3642–3649.
17. C.-Y. Lee, P. W. Gallagher, Z. Tu, in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics* (AISTATS, 2016), pp. 464–472.
18. M. A. Ranzato, C. Poultney, S. Chopra, Y. LeCun, in *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, J. C. Platt, D. Koller, Y. Singer, S. T. Roweis, Eds. (Curran Associates, 2007), pp. 1137–1144.
19. H. Xiao, K. Rasul, R. Vollgraf, arXiv:1708.07747 [cs.LG] (25 August 2017).
20. S. Zeng, B. Zhang, J. Gou, arXiv:1802.07589 [cs.CV] (21 February 2018).
21. ImageNet database, www.image-net.org.
22. Y. Shen *et al.*, *Nat. Photonics* **11**, 441–446 (2017).
23. D. Psaltis, D. Brady, X.-G. Gu, S. Lin, *Nature* **343**, 325–330 (1990).
24. K. H. Wagner, in *Proceedings of Frontiers in Optics 2017* (OSA, 2017), FW2C.1.
25. B. J. Shastri *et al.*, arXiv:1801.00016 [cs.ET] (29 December 2017).
26. M. Hermans, M. Burm, T. Van Vaerenbergh, J. Dambre, P. Bienstman, *Nat. Commun.* **6**, 6729 (2015).
27. D. Brunner, M. C. Soriano, C. R. Mirasso, I. Fischer, *Nat. Commun.* **4**, 1364 (2013).
28. A. Greenbaum *et al.*, *Nat. Methods* **9**, 889–895 (2012).
29. A. Ozcan, E. McLeod, *Annu. Rev. Biomed. Eng.* **18**, 77–102 (2016).
30. M. Emons *et al.*, *Opt. Mater. Express* **2**, 942–947 (2012).

SUPPLEMENTARY MATERIALS

www.sciencemag.org/content/361/6406/1004/suppl/DC1
Materials and Methods
Figs. S1 to S16
References (*31–46*)

Science

All-optical machine learning using diffractive deep neural networks

Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi and Aydogan Ozcan

All-optical deep learning

Deep learning uses multilayered artificial neural networks to learn digitally from large datasets. It then performs advanced identification and classification tasks. To date, these multilayered neural networks have been implemented on a computer. Lin *et al.* demonstrate all-optical machine learning that uses passive optical components that can be patterned and fabricated with 3D-printing. Their hardware approach comprises stacked layers of diffractive optical elements analogous to an artificial neural network that can be trained to execute complex functions at the speed of light.
Science, this issue p. 1004

ARTICLE TOOLS	http://science.sciencemag.org/content/361/6406/1004
SUPPLEMENTARY MATERIALS	http://science.sciencemag.org/content/suppl/2018/07/25/science.aat8084.DC1
REFERENCES	This article cites 29 articles, 2 of which you can access for free http://science.sciencemag.org/content/361/6406/1004#BIBL
PERMISSIONS	http://www.sciencemag.org/help/reprints-and-permissions

Use of this article is subject to the Terms of Service

Science

AAAS

Supplementary Material for

All-optical machine learning using diffractive deep neural networks

Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, Aydogan Ozcan*

***Corresponding author. Email: ozcan@ucla.edu

This PDF file includes:

    Materials and Methods
    Figs. S1 to S16
    References

Correction: In the top panel of fig. S2, the left *y* axis contained a labeling error: "0.4 Million Neurons" has been corrected to "0.2 Million Neurons."

Materials and Methods

TensorFlow-based design and 3D-printing of a D^{2}NN. We implemented D^{2}NN design using TensorFlow (Google Inc.) framework, as shown in Fig. S11. Because we consider coherent illumination, the input information can be encoded in the amplitude and/or phase channels of the input plane. The free-space propagation module is implemented using the angular spectrum method. To help with the 3D-printing and fabrication of the D^{2}NN design, a sigmoid function was used to limit the phase value of each neuron to 0-2π and 0-π, for imaging and classifier networks, respectively. For each layer of the D^{2}NN, we set the neuron size to be 400 µm and 300 µm, for the classifier networks and the imaging network, respectively. With a higher resolution 3D-printer or fabrication method, smaller neurons can also be used in our D^{2}NN design to increase the number of neurons and connections to learn more complicated tasks. Furthermore, as illustrated in Fig. S7A, the number of the network layers and the axial distance between the layers are also design parameters.

At the detector/output plane, we measured the intensity of the network output, and as a loss function to train the imaging D^{2}NN, we used its mean square error (MSE) against the target image. The classification D^{2}NNs were also trained using a nonlinear loss function, where we aimed to maximize the normalized signal of each target's corresponding detector region, while minimizing the total signal outside of all the detector regions (see e.g., Fig. 3A). We used the stochastic gradient descent algorithm, Adam (*31*), to back-propagate the errors and update the layers of the network to minimize the loss function. The digit classifier and lens D^{2}NNs were trained with MNIST (*15*) and ImageNet (*21*) datasets, respectively, and achieved the desired mapping functions between the input and output planes after 10 and 50 epochs, respectively. The training batch size was set to be 8 and 4, for the digit classifier network and the imaging network, respectively. The training phase of the fashion product classifier network shared the same details as the digit classifier network, except using the Fashion MNIST dataset (*19*). The networks were implemented using Python version 3.5.0. and TensorFlow framework version 1.4.0 (Google Inc.). Using a desktop computer (GeForce GTX 1080 Ti Graphical Processing Unit, GPU and Intel(R) Core(TM) i7-7700 CPU @3.60GHz and 64GB of RAM, running a Windows 10 operating system, Microsoft), the above-

outlined TensorFlow based design of a D²NN architecture took approximately 8 hours and 10 hours to train for the classifier and the lens networks, respectively.

After the training phase of the optimized D²NN architecture, the 3D model of the network layers to be 3D-printed was generated by Poisson surface reconstruction (*32*) (see Fig. S12). First, neurons' phase values were converted into a relative height map ($\Delta z = \lambda \phi / 2\pi \Delta n$), where Δn is the refractive index difference between the 3D printing material (VeroBlackPlus RGD875) and air. The refractive index n and the extinction coefficient (k) of this 3D-printing material at 0.4 THz were measured as 1.7227 and 0.0311, respectively, which corresponds to an attenuation coefficient of $\alpha = 520.7177 \ m^{-1}$. Before the 3D-printing process, we also added a uniform substrate thickness of 0.5 mm to each layer of a D²NN. A 3D mesh processing software, Meshlab (*33-34*), was used to calculate the 3D structure, which was then used as input to a 3D-printer (Objet30 Pro 3D, Stratasys Ltd, Eden Prairie, Minnesota USA). For the training of MNIST digit classifier D²NN and Fashion-MNIST classifier D²NN, we padded input images with zeros to fit the input aperture of the diffractive network (8 cm x 8 cm). In our THz experiments we used aluminum foil to create zero transmission regions at the input plane, to match our training settings for each D²NN design.

Following the corresponding D²NN design, the axial distance between two successive 3D-printed layers was set to be 3.0 cm and 4.0 mm for the classifier and lens networks, respectively. The larger axial distance between the successive layers of the classifier D²NNs increased the number of neuron connections to ~8 billion, which is approximately 100-fold larger compared to the number of the neuron connections of the imaging D²NN, which is much more compact in depth (see Figs. 2(A, B)).

Terahertz Set-up. The schematic diagram of the experimental setup is given in Fig. 2C. The electromagnetic wave was generated through a WR2.2 modular amplifier/multiplier chain (AMC) made by Virginia Diode Inc. (VDI). A 16 dBm sinusoidal signal at 11.111 GHz (f_{RF1}) was sent as RF input signal and multiplied 36 times by AMC to generate continuous-wave (CW) radiation at 0.4 THz. We used a horn antenna compatible with WR 2.2 modular AMC. The source was electrically-modulated at 1 KHz. The illumination beam profile was characterized

as a Gaussian (Fig. S13), and the distance between the object and the source planes was selected as approximately 81 mm, 173 mm, and 457 mm to provide a beam spot size of ~20 mm, ~40 mm, and ~104 mm, full-width half-maximum (FWHM), for the imaging D²NN, the digit classification D²NN, and the fashion product classification D²NN, respectively. The beam passed through the input object and then the optical neural network, before reaching the output plane, which was scanned by a single-pixel detector placed on an XY positioning stage. This XY stage was built by placing two linear motorized stages (Thorlabs NRT100) vertically to allow precise control of the position of the detector. The detector scanning step size was set to be ~600 µm, ~1.2 mm, and ~1.6 mm for the imaging lens D²NN, the digit classifier D²NN, and the fashion classifier D²NN, respectively. The distance between detector/output plane and the last layer of the optical neural network was adjusted as 3 cm and 7 mm for the classifier D²NNs and the lens D²NN, respectively. We used a Mixer/AMC made by VDI to detect the amplitude of the transmitted wave (f_{opt}). A 10-dBm sinusoidal signal at 11.138 GHz (fRF2) was used as a local oscillator. This signal was multiplied by 36 through the multiplier and mixed with the detected signal. The mixing product ($f_{IR} = |f_{RF1} - f_{opt}|$) was obtained at 1 GHz frequency. This down-converted signal passed through an amplification stage which consisted of two low-noise amplifiers (Mini-Circuits ZRL-1150-LN+) to amplify the signal by 80 dBm and a 1 GHz (+/-10 MHz) bandpass filter (KL Electronics 3C40-1000/T10-O/O) to get rid of the noise coming from unwanted frequency bands. After this, the signal went through a low-noise power detector (Mini-Circuits ZX47-60) and the output voltage was read by a lock-in amplifier (Stanford Research SR830). The modulation signal was used as the reference signal for the lock-in amplifier. The dynamic range of the setup was measured as 80 dB.

Wave analysis in a D²NN. Following the Rayleigh-Sommerfeld diffraction equation (*35*), one can consider every single neuron of a given D²NN layer as a secondary source of a wave that is composed of the following optical mode:

$$w_i^l(x, y, z) = \frac{z - z_i}{r^2} \left(\frac{1}{2\pi r} + \frac{1}{j\lambda} \right) exp \left(\frac{j2\pi r}{\lambda} \right), \tag{1}$$

where l represents the l-th layer of the network, i represents the i-th neuron located at (x_i, y_i, z_i) of layer l, λ is the illumination wavelength, $r = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}$ and $j = \sqrt{-1}$. The amplitude and relative phase of this secondary wave are determined by the product of the input wave to the neuron and its transmission coefficient (t), both of which are complex-valued functions. Based on this, for the l-th layer of the network, one can write the output function (n_i^l) of the i-th neuron located at (x_i, y_i, z_i) as:

$$n_i^l(x, y, z) = w_i^l(x, y, z) \cdot t_i^l(x_i, y_i, z_i) \cdot \sum_k n_k^{l-1}(x_i, y_i, z_i) = w_i^l(x, y, z) \cdot |A| \cdot e^{j\Delta\theta} , \qquad (2)$$

where we define $m_i^l(x_i, y_i, z_i) = \sum_k n_k^{l-1}(x_i, y_i, z_i)$ as the input wave to i-th neuron of layer l, $|A|$ refers to the relative amplitude of the secondary wave, and $\Delta\theta$ refers to the additional phase delay that the secondary wave encounters due to the input wave to the neuron and its transmission coefficient. These secondary waves diffract between the layers and interfere with each other forming a complex wave at the surface of the next layer, feeding its neurons. The transmission coefficient of a neuron is composed of amplitude and phase terms, i.e., $t_i^l(x_i, y_i, z_i) = a_i^l(x_i, y_i, z_i) exp(j\phi_i^l(x_i, y_i, z_i))$, and for a phase-only D²NN architecture the amplitude $a_i^l(x_i, y_i, z_i)$ is assumed to be a constant, ideally 1, ignoring the optical losses, which is discussed in this Supplementary Materials document under the sub-section "Optical Losses in a D²NN". In general, a complex-valued modulation at each network layer improves the inference performance of the diffractive network (see e.g., figs. S1 and S4).

Forward Propagation Model. The forward model of our D²NN architecture is illustrated in Fig. 1A and its corresponding TensorFlow implementation is summarized in Fig. S11A. To simplify the notation of the forward model, we can rewrite Eq. (2) as follows:

$$\begin{cases} n_{i,p}^l = w_{i,p}^l \cdot t_i^l \cdot m_i^l \\ m_i^l = \sum_k n_{k,i}^{l-1} \\ t_i^l = a_i^l exp(j\phi_i^l) \end{cases} ,$$

$$(3)$$

where i refers to a neuron of the l-th layer, and p refers to a neuron of the next layer, connected to neuron i by optical diffraction. The same expressions would also apply for a reflective D^{2}NN with a reflection coefficient per neuron: r_i^l. The input pattern h_k^0, which is located at layer 0 (i.e., the input plane), is in general a complex-valued quantity and can carry information in its phase and/or amplitude channels. The resulting wave function due to the diffraction of the illumination plane-wave interacting with the input can be written as:

$$n_{k,p}^0 = w_{k,p}^0 \cdot h_k^0,$$

$$(4)$$

which connects the input to the neurons of layer 1. Assuming that the D^{2}NN design is composed of M layers (*excluding* the input and output planes), then a detector at the output plane measures the intensity of the resulting optical field:

$$s_i^{M+1} = \left| m_i^{M+1} \right|^2. \tag{5}$$

The comparison of the forward model of a conventional artificial neural network and a diffractive neural network is summarized in Fig. 1D of main text. Based on this forward model, the results of the network output plane are compared with the targets (for which the diffractive network is being trained for) and the resulting errors are back-propagated to iteratively update the layers of the diffractive network, which will be detailed next.

Error Backpropagation. To train a D^{2}NN design, we used the error back-propagation algorithm along with the stochastic gradient descent optimization method. A loss function was defined to evaluate the performance of the D^{2}NN output with respect to the desired target, and the algorithm iteratively optimized the diffractive neural network parameters to minimize the loss function. Without loss of generality, here we focus on our imaging D^{2}NN architecture, and define the loss function (E) using the mean square error between the output plane intensity s_i^{M+1} and the target, g_i^{M+1}:

$$E(\phi_i^l) = \frac{1}{K}\sum_k (s_k^{M+1} - g_k^{M+1})^2, \tag{6}$$

where K refers to the number of measurement points at the output plane. Different loss functions can also be used in D²NN. Based on this error definition, the optimization problem for a D²NN design can be written as:

$$\min_{\phi_i^l} E\left(\phi_i^l\right), \ \ s.t. \ \ \ \ 0 \le \phi_i^l < 2\pi. \tag{7}$$

To apply the backpropagation algorithm for training a D²NN, the gradient of the loss function with respect to all the trainable network variables needs to be calculated, which is then used to update the network layers during each cycle of the training phase. The gradient of the error with respect to ϕ_i^l of a given layer l can be calculated as:

$$\frac{\partial E(\phi_i^l)}{\partial \phi_i^l} = \frac{4}{K}\sum_k (s_k^{M+1} - g_k^{M+1}) \cdot Real\{(m_k^{M+1})^* \cdot \frac{\partial m_k^{M+1}}{\partial \phi_i^l}\}.$$

(8)

In Eq. (8), $\frac{\partial m_k^{M+1}}{\partial \phi_i^l}$ quantifies the gradient of the complex-valued optical field at the output layer ($m_k^{M+1} = \sum_{k_1} n_{k_1,k}^M$) with respect to the phase values of the neurons in the previous layers, $l \le M$. For every layer, l, this gradient can be calculated using:

$$\frac{\partial m_k^{M+1}}{\partial \phi_i^{l=M}} = j \cdot t_i^M \cdot m_i^M \cdot w_{i,k}^M, \tag{9}$$

$$\frac{\partial m_k^{M+1}}{\partial \phi_i^{l=M-1}} = j \cdot t_i^{M-1} \cdot m_i^{M-1} \cdot \sum_{k_1} w_{k_1,k}^M \cdot t_{k_1}^M \cdot w_{i,k_1}^{M-1}, \tag{10}$$

$$\frac{\partial m_k^{M+1}}{\partial \phi_i^{l=M-2}} = j \cdot t_i^{M-2} \cdot m_i^{M-2} \cdot \sum_{k_1} w_{k_1,k}^M \cdot t_{k_1}^M \cdot \sum_{k_2} w_{k_2,k_1}^{M-1} \cdot t_{k_2}^{M-1} \cdot w_{i,k_2}^{M-2}, \tag{11}$$

....

$$\frac{\partial m_k^{M+1}}{\partial \phi_i^{l=M-L}} = j \cdot t_i^{M-L} \cdot m_i^{M-L} \cdot \sum_{k_1} w_{k_1,k}^M \cdot t_{k_1}^M \ \cdots \cdots \ \sum_{k_L} w_{k_L,k_{L-1}}^{M-L+1} \cdot t_{k_L}^{M-L+1} \cdot w_{i,k_L}^{M-L}, \tag{12}$$

where, $3 \le L \le M$ -1. In the derivation of these partial derivatives, an important observation is that, for an arbitrary neuron at layer $l \le M$, one can write:

7

$$\frac{\partial n_{k_2,k_1}^l}{\partial \phi_i^l} = \begin{cases} j \cdot t_i^l \cdot m_i^l \cdot w_{i,k_1}^l, & for\ k_2 = i \\ 0, & for\ k_2 \neq i \end{cases}, \tag{13}$$

where $k_{1,2}$ represent dummy variables. During each iteration of the error backpropagation, a small batch of the training data is fed into the diffractive neural network to calculate the above gradients for each layer and accordingly update the D²NN.

Comparison with standard deep neural networks. Compared to standard deep neural networks, a D²NN is not only different in that it is a physical and all-optical deep network, but also it possesses some unique architectural differences. First, the inputs for neurons are complex-valued, determined by wave interference and a multiplicative bias, i.e., the transmission/reflection coefficient. Complex-valued deep neural networks (implemented in a computer) with additive bias terms have been recently reported as an alternative to real-valued networks, achieving competitive results on e.g., music transcription (*36*). In contrast, this work considers a coherent diffractive network modelled by physical wave propagation to connect various layers through the phase and amplitude of interfering waves, controlled with multiplicative bias terms and physical distances. Second, the individual function of a neuron is the phase and amplitude modulation of its input to output a secondary wave, unlike e.g., a sigmoid, a rectified linear unit (ReLU) or other nonlinear neuron functions used in modern deep neural networks. Although not implemented here, optical nonlinearity can also be incorporated into a diffractive neural network in various ways; see the sub-section "Optical Nonlinearity in Diffractive Neural Networks" (*14*). Third, each neuron's output is coupled to the neurons of the next layer through wave propagation and coherent (or partially-coherent) interference, providing a unique form of interconnectivity within the network. For example, the way that a D²NN adjusts its receptive field, which is a parameter used in convolutional neural networks, is quite different than the traditional neural networks, and is based on the axial spacing between different network layers, the signal-to-noise ratio (SNR) at the output layer as well as the spatial and temporal coherence properties of the illumination source. The secondary wave of each neuron will in theory diffract in all angles, affecting in principle all the neurons of the following layer. However, for a given spacing between the successive layers, the intensity of the wave from a neuron will decay below the detection noise floor after a certain propagation distance;

the radius of this propagation distance at the next layer practically sets the receptive field of a diffractive neural network and can be physically adjusted by changing the spacing between the network layers, the intensity of the input optical beam, the detection SNR or the coherence length and diameter of the illumination source.

Imaging D^{2}NN Architecture. Structural similarity index, SSIM (*37*), values between the D^{2}NN output plane and the ground truth (i.e., target images) were calculated to optimize the architecture of the diffractive neural network. This way, we optimized the number of network layers and the axial distance between two consecutive layers as shown in Fig. S7A. The SSIM plots in Fig. S7A were calculated by averaging the results of 100 test images randomly selected from ImageNet dataset.

Note also that, based on the large area of the 3D-printed imaging network layers (9 × 9 cm) and the short axial distance between the input (output) plane and the first (last) layer of the network, i.e., 4 mm (7 mm), one can infer that the theoretical numerical aperture of our system approaches 1 in air (see Fig. 2B of main text). During the training phase, however, our diffractive network learned to utilize only part of this spatial frequency bandwidth, which should be due to the relatively large-scale of the image features that we used in the training image set (randomly selected from ImageNet database). If a higher resolution imaging system is desired, images that contain much finer spatial features can be utilized as part of the training phase to design a D^{2}NN that can approach the theoretical diffraction-limited numerical aperture of the system. One can also change the loss function definition used in the training phase to teach the diffractive neural network to enhance the spatial resolution; in fact deep learning provides a powerful framework to improve image resolution by engineering the loss function used to train a neural network (*8, 13*).

Dataset Preprocessing. To train and test the D^{2}NN as a digit classifier, we utilized MNIST handwritten digit database (*15*), which is composed of 55,000 training images, 5,000 validation images and 10,000 testing images. Images were up-sampled to match the size of the D^{2}NN model. For the training and testing of the imaging D^{2}NN, we used ImageNet (*21*) where we randomly selected a subset of 2,000 images. We converted each color image into grayscale and resized it to match our D^{2}NN design. (We should note that color image data can also be applied to D^{2}NN framework using different approaches although we did not consider it in our work since we utilized a

9

single wavelength THz system for testing. For colorful images, as an example, Red, Green and Blue channels of an image can be used as separate parallel input planes to a diffractive neural network.) The selected images were then randomly divided into 1500 training images, 200 validation images and 300 testing images. We also obtained very similar imaging performance by using 10,000 images in the training phase (instead of 2,000 images); this is expected since each training image contains various spatial features at different parts of the image, all of which provide valuable patches of information for successfully training our diffractive imaging network.

To test the performance of the D^2NN digit classifier experimentally, 50 handwritten digits were extracted from MNIST test database. To solely quantify the match between our numerical testing results and experimental testing, these 3D-printed handwritten digits were selected among the same 91.75% of the test images that numerical testing was successful. The digits were up-sampled and binarized, as implemented during the training process. Binarized digits were stored as a vector image, in .svg format, before they were 3D printed. The images were then fed into Autodesk Fusion Software (Autodesk Inc.) to generate their corresponding 3D model. To provide amplitude only image inputs to our digit classifier D^2NN, the 3D-printed digits were coated with aluminum foil to block the light transmission in desired regions.

In addition to MNIST digit classification, to test our D^2NN framework with a more challenging classification task, we used the Fashion MNIST database which has more complicated targets as exemplified in Fig. S3. Some of these target classes, such as pullovers (class 2), coats (class 4) and shirts (class 6), are very similar to each other, making it difficult for different classification methods. For example, the state-of-the-art DENSER convolutional neural network achieves 95.3% classification accuracy on Fashion MNIST dataset compared with 99.7% for MNIST dataset (*19*). In order to train a D^2NN with Fashion MNIST database, we encoded the target fashion product images into the phase channel of the input plane instead of the amplitude channel. Grayscale images corresponding to fashion products were scaled between 0 and 2π as the phase-only input to the diffractive neural network, and other details of the Fashion MNIST experiments were similar as in MNIST classification experiments.

D²NN Neuron Numbers and Connectivity. D²NN uses optical diffraction to connect the neurons at different layers of the network. The maximum half-cone diffraction angle can be formulated as $\varphi_{max} = \sin^{-1}(\lambda f_{max})$, where $f_{max} = 1/2d_f$ is the maximum spatial frequency and d_f is the layer feature size (*35*). In this work, we demonstrated the proof-of-concept of D²NN architecture at 0.4 THz by using low-cost 3D-printed layers. The 3D printer that we used has a spatial resolution of 600 dpi with 0.1 mm accuracy and the wavelength of the illumination system is 0.75 mm in air.

For the digit and fashion product classification D²NNs, we set the pixel size to 400 µm for packing 200×200 neurons over each layer of the network, covering an area of 8 cm × 8 cm per layer. We used 5 transmissive diffraction layers with the axial distance between the successive layers set to be 3cm. These choices mean that we have a fully-connected diffractive neural network structure because of the relatively large axial distance between the two successive layers of the diffractive network. This corresponds to 200×200×5=0.2 million neurons (each containing a trainable phase term) and $(200 \times 200)^2 \times 5 = 8.0$ billion connections (including the connections to the output layer). This large number of neurons and their connections offer a large degree-of-freedom to train the desired mapping function between the input amplitude (handwritten digit classification) or input phase (fashion product classification) and the output intensity measurement for classification of input objects.

For the imaging lens D²NN design, the smallest feature size was ~0.9 mm with a pixel size set of 0.3 mm, which corresponds to a half-cone diffraction angle of ~25°. The axial distance between two successive layers is set to be 4 mm for 5 layers, and the width of each layer was 9 cm × 9 cm. This means the amplitude imaging D²NN design had 300×300×5=0.45 million neurons, each having a trainable phase term. Because of the relatively small axial distance (4 mm) between the successive layers and the smaller diffraction angle due to the larger feature size, we have <0.1 billion connections in this imaging D²NN design (including the connections to the output layer, which is 7 mm away from the 5th layer of the diffractive network). Compared to the classification D²NNs, this amplitude imaging one is much more compact in the axial direction as also pictured in Fig. 2(A, B) of the main text.

Finally, we would like to emphasize that there are some unique features of a D^{2}NN that make it easier to handle large scale connections (e.g., 8 billion connections as reported in Fig. 2A of our main text). The connectivity of a D^{2}NN is controlled by the size of each neuron of a given layer (defining the diffraction angle) and the axial spacing between the layers. For example, consider a 5-layer D^{2}NN design with a certain fixed neuron size; for this design, one can have a very low number of neural connections by closely placing the layers, one after another. On the other hand, one can also make the same design fully-connected by simply increasing the axial spacing between the layers, significantly increasing the number of connections. Interestingly, these two extreme designs (that vary considerably in their number of connections) would be identical in terms of training complexity because the computation time and complexity of digital wave propagation between layers is a not a function of the axial distance. Therefore largely spaced D^{2}NN layers that form a fully connected network would be identical (in terms of their computational implementation complexity) to partially-connected D^{2}NN designs that have shorter axial distance between the layers (also see Fig. S4, top two rows, for an example of this comparison).

Performance analysis of D^{2}NN as a function of the number of layers and neurons. A single diffractive layer cannot achieve the same level of inference that a multi-layer D^{2}NN structure can perform. Multi-layer architecture of D^{2}NN provides a large degree-of-freedom within a physical volume to train the transfer function between its input and the output planes, which, in general, cannot be replaced by a single phase-only or complex modulation layer (employing phase and amplitude modulation at each neuron).

To expand on this, we would like to first show that, indeed, a single diffractive layer performance is quite primitive compared to a multi-layered D^{2}NN. As shown in Fig. S1, a single phase-only modulation layer or even a complex modulation layer (where both phase and amplitude of each neuron are learnable parameters) cannot present enough degrees of freedom to establish the desired transfer function for classification of input images (MNIST) and achieves a much lower performance compared to a 5-layer D^{2}NN network, the one that we demonstrated in the main text. In these results reported in Fig. S1, the same physical neuron size was used in each case, representing our 3D-printing resolution. Fig. S1 shows that a single layer diffractive network can only achieve 55.64% and 64.84% blind testing accuracy for phase-only and complex modulation D^{2}NN designs, respectively,

whereas N=5 layers (with everything else being the same) can achieve 91.75% and 93.23% blind testing accuracy, respectively. The same conclusion also applies for a single layer D^{2}NN (N=1) that has 0.2 million neurons over the same area (assuming a higher resolution 3D-printer was available for defining smaller neurons).

Figure S2 further demonstrates that by using a patch of 2 layers added to an existing/fixed D^{2}NN (N=5), we improved our MNIST classification accuracy to 93.39%; the state of the art convolutional neural net performance varies between 99.60%-99.77% depending on the network design (*16-18*). We have obtained similar results for the Fashion MNIST dataset using N=5, 10 layers (see Figs. S4-S5).

These results, summarized above, highlight that a single diffractive layer stagnates at its inference performance to modest accuracy values, and increasing the number of layers, neurons and connections of a D^{2}NN design provides significant improvements in its inference capability.

Error sources and mitigation strategies. There are five main sources of error that contribute to the performance of a 3D-printed D^{2}NN: (1) Poisson surface reconstruction is the first error source. After the transmission layers are trained, 3D structure of each layer is generated through the Poisson surface reconstruction as detailed in earlier. However, for practical purposes, we can only use a limited number of sampling points, which distorts the 3D structure of each layer. (2) Alignment errors during the experiments form the second source of error. To minimize the alignment errors, the transmission layers and input objects are placed into single 3D printed holder. However, considering the fact that 3D printed materials have some elasticity, the thin transmission layers do not perfectly stay flat, and they will have some curvature. Alignment of THz source and detector with respect to the transmission layers also creates another error source in our experiments. (3) 3D-printing is the third and one of the most dominant sources of error. This originates from the lack of precision and accuracy of the 3D-printer used to generate network layers. It smoothens the edges and fine details on the transmission layers. (4) Absorption of each transmissive layer is another source that can deteriorate the performance of a D^{2}NN design. (5) The measurements of the material properties that are extensively used in our simulations such as refractive index and extinction coefficient of the 3D printed material might have some additional sources of error, contributing to a

reduced experimental accuracy. It is hard to quantitatively evaluate the overall magnitude of these various sources of errors; instead we incorporated the Poisson surface reconstruction errors, absorption related losses at different layers and 0.1 mm random misalignment error for each network layer during the testing phase of the D^2NNs as shown in Figs. S7 and S14. These errors showed minor influence on the performance of the diffractive networks.

To minimize the impact of the 3D printing error, we set a relatively large pixel size, i.e. 0.4 mm and 0.3 mm for the classification and imaging D^2NNs, respectively. Furthermore, we designed a 3D-printed holder (Figs. 2(A, B)) to self-align the multi-layer structure of a 3D-printed D^2NN, where each network layer and the input object were inserted into their specific slots. Based on the resolution of our 3D-printer, the misalignment error of a 3D-printed D^2NN (including its holder) is estimated to be smaller than 0.1 mm compared to the ideal positions of the neurons of a given layer, and this level of error was found to have a minor effect on the network performance as illustrated in Figs. S7 and S14.

For an inexpensive 3D-printer or fabrication method, printing/fabrication errors and imperfections, and the resulting alignment problems can be further mitigated by increasing the area of each layer and the footprint of the D^2NN. This way, the feature size at each layer can be increased, which will partially release the alignment requirements. A minor disadvantage of such an approach of printing larger diffractive networks, with an increased feature size, would be an increase in the physical size of the system and its input illumination power requirements. Furthermore, to avoid bending of the network layers over larger areas, an increase in layer thickness and hence its stiffness would be needed, which can potentially also introduce additional optical losses (discussed next), depending on the illumination wavelength and the material properties.

Optical Losses in a D^2NN. For a D^2NN, after all the parameters are trained and the physical diffractive network is fabricated or 3D-printed, the computation of the network function (i.e., inference) is implemented all-optically using a light source and optical diffraction through passive components. Therefore, the energy efficiency of a D^2NN depends on the reflection and/or transmission coefficients of the network layers. Such optical losses can be made negligible, especially for phase-only networks that employ e.g., transparent materials that are structured using e.g., optical lithography, creating D^2NN designs operating at the visible part of the spectrum. In our

experiments, we used a standard 3D-printing material (VeroBlackPlus RGD875) to provide phase modulation, and each layer of the networks shown in Fig. 2 (main text) had on average ~51% power attenuation at 0.4 THz for an average thickness of ~1 mm (see Fig. S15). This attenuation could be further decreased by using thinner substrates or by using other materials (e.g., polyethylene, polytetrafluoroethylene) that have much lower losses in THz wavelengths. One might also use the absorption properties of the neurons of a given layer as another degree of freedom in the network design to control the connectivity of the network, which can be considered as a physical analog of the dropout rate in deep network training (*38*). In principle, a phase-only D^{2}NN can be designed by using the correct combination of low-loss materials and appropriately selected illumination wavelengths, such that the energy efficiency of the diffractive network is only limited by the Fresnel reflections that happen at the surfaces of different layers. Such reflection related losses can also be engineered to be negligible by using anti-reflection coatings on the substrates. So far, the consideration of multiple-reflections between the layers has been neglected since such waves are much weaker compared to the directly transmitted forward-propagating waves. The strong match between the experimental results obtained with our 3D-printed D^{2}NNs and their numerical testing also supports this (see Figs. 3 and 4 of the main text).

Although not considered in this manuscript since we are dealing with *passive* diffractive neural networks, diffractive networks can be created that use a physical gain (e.g., through optical or electrical pumping, or nonlinear optical phenomena, including but not limited to plasmonics and metamaterials) to explore the domain of *amplified bias terms*, i.e., $|t_i^l| > 1$ or $|r_i^l| > 1$. At the cost of additional complexity, such amplifying layers can be useful for the diffractive neural network to better handle its photon budget and can be used after a certain number of passive layers to boost up the diffracted signal, intuitively similar to e.g., optical amplifiers used in fiber optic communication links.

Transmission and reflection modes of operation in D^{2}NNs. The architecture of our D^{2}NN can be implemented in transmission or reflection modes by using multiple layers of diffractive surfaces; in transmission (or reflection) mode of operation, the information that is transferred from one diffractive layer to the other is carried with the transmitted (or reflected) optical wave. The operation principles of D^{2}NN can be easily extended to amplitude-

only or phase/amplitude-mixed network designs. Whether the network layers perform phase-only or amplitude-only modulation, or a combination of both, what changes from one design to another is only the nature of the multiplicative bias terms, t_i^l or r_i^l for a transmissive or reflective neuron, respectively, and each neuron of a given layer will still be connected to the neurons of the former layer through a wave-interference process, $\sum_k n_k^{l-1}(x_i, y_i, z_i)$, which provides the complex-valued input to a neuron. Compared to a phase-only D^{2}NN design, where $|t_i^l| = |r_i^l| = 1$, a choice of $|t_i^l| < 1$ or $|r_i^l| < 1$ would introduce additional optical losses, and would need to be taken into account for a given illumination power and detection SNR at the network output plane.

Reconfigurable D^{2}NN Designs. One important avenue to consider is the use of spatial light modulators (SLMs) as part of a diffractive neural network. This approach of using SLMs in D^{2}NNs has several advantages, at the cost of an increased complexity due to deviation from an entirely passive optical network to a reconfigurable electro-optic one. First, a D^{2}NN that employs one or more SLMs can be used to learn and implement various tasks because of its reconfigurable architecture. Second, this reconfigurability of the physical network can be used to mitigate alignment errors or other imperfections in the optical system of the network. Furthermore, as the optical network statistically fails, e.g., a misclassification or an error in its output is detected, it can mend itself through a transfer learning based re-training with appropriate penalties attached to some of the discovered errors of the network as it is being used. For building a D^{2}NN that contains SLMs, both reflection and transmission based modulator devices can be used to create an optical network that is either entirely composed of SLMs or a hybrid one, i.e., employing some SLMs in combination with fabricated (i.e., passive) layers.

In addition to the possibility of using SLMs as part of a reconfigurable D^{2}NN, another option to consider is to use a given 3D-printed or fabricated D^{2}NN design as a fixed input block of a new diffractive network where we train only the additional layers that we plan to fabricate. Assume for example that a 5-layer D^{2}NN has been printed/fabricated for a certain inference task. As its prediction performance degrades or slightly changes, due to e.g., a change in the input data, etc., we can train a few new layers to be physically added/patched to the existing printed/fabricated network to improve its inference performance. In some cases, we can even peel off (i.e., discard)

some of the existing layers of the printed network and assume the remaining fabricated layers as a fixed (i.e., non-learnable) input block to a new network where the new layers to be added/patched are trained for an improved inference task (coming from the entire diffractive network: old layers and new layers).

Intuitively, we can think of each D^2NN as a "Lego" piece (with several layers following each other); we can either add a new layer (or layers) on top of existing (i.e., already fabricated) ones, or peel off some layers and replace them with the new trained diffractive blocks. This provides a unique physical implementation (like blocks of Lego) for transfer learning or mending the performance of a printed/fabricated D^2NN design.

We implemented this concept of Lego design for our Fashion MNIST diffractive network and our results are summarized in Fig. S16, demonstrating that, for example, the addition of a 6th layer (learnable) to an already trained and fixed D^2NN with N=5 improves its inference performance, performing slightly better than the performance of a D^2NN with N=6 layers that were simultaneously trained. Also see Fig. S2 for an implementation of the same concept for MNIST: using a patch of 2 layers added to an existing/fixed D^2NN (N=5), we improved our MNIST classification accuracy to 93.39%. The advantage of this Lego-like transfer learning or patching approach is that already fabricated and printed D^2NN designs can be improved in performance by adding additional printed layers to them or replacing some of the existing diffractive layers with newly trained ones. This can also help us with the training process of very large network designs (e.g., N $\geqslant$ 25) by training them in patches, making it more tractable with state of the art computers.

Discussion of Unique Imaging Functionalities using D^2NNs. We believe that the D^2NN framework will help imaging at the macro and micro/nano scale by enabling all-optical implementation of some unique imaging tasks. One possibility for enhancing imaging systems could be to utilize D^2NN designs to be integrated with sample holders or substrates used in microscopic imaging to enhance certain bands of spatial frequencies and create new contrast mechanisms in the acquired images. In other words, as the sample on a substrate (e.g., cells or tissue samples, etc.) diffracts light, a D^2NN can be used to project magnified images of the cells/objects onto a CMOS/CCD chip with certain spatial features highlighted or enhanced, depending on the training of the

diffractive network. This could form a very compact chip-scale microscope (just a passive D^2NN placed on top of an imager chip) that implements, all-optically, task specific contrast imaging and/or object recognition or tracking within the sample. Similarly, for macro-scale imaging, face recognition, as an example, could be achieved as part of a sensor design, without the need for a high mega-pixel imager. For instance, tens to hundreds of different classes can potentially be detected using a modest (e.g., <1 Mega-pixel) imager chip placed at the output plane of a D^2NN that is built for this inference task.

For THz part of the spectrum, as another possible use example, various biomedical applications that utilize THz imagers for looking into chemical sensing or the composition of drugs to detect e.g., counterfeit medicine, or for assessing the healing of wounds etc. could benefit from D^2NN designs to automate predictions in such THz-based analysis of specimen using a diffractive neural network.

Optical Nonlinearity in Diffractive Deep Neural Networks. Optical nonlinearity can be incorporated into our deep optical network design using various optical non-linear materials (crystals, polymers, semiconductor materials, doped glasses, among others as detailed below). A D^2NN is based on controlling the diffraction of light through complex-valued diffractive elements to perform a desired/trained task. Augmenting nonlinear optical components is both practical and synergetic to our D^2NN framework.

Assuming that the input object, together with the D^2NN diffractive layers, create a spatially varying complex field amplitude E(x,y) at a given network layer, then the use of a nonlinear medium (e.g., optical Kerr effect based on third-order optical nonlinearity, $\chi^{(3)}$) will introduce an all-optical refractive index change which is a function of the input field's intensity, $\Delta n \propto \chi^{(3)} E^2$. This intensity dependent refractive index modulation and its impact on the phase and amplitude of the resulting waves through the diffractive network can be numerically modeled and therefore is straightforward to incorporate as part of our network training phase. Any third-order nonlinear material with a strong $\chi^{(3)}$ could be used to form our nonlinear diffractive layers: glasses (e.g., As_2S_3, metal nanoparticle doped glasses), polymers (e.g., polydiacetylenes), organic films, semiconductors (e.g., GaAs, Si,

CdS), graphene, among others. There are different fabrication methods that can be employed to structure each nonlinear layer of a diffractive neural network using these materials.

In addition to third-order all-optical nonlinearity, another method to introduce nonlinearity into a D^2NN design is to use saturable absorbers that can be based on materials such as semiconductors, quantum-dot films, carbon nanotubes or even graphene films. There are also various fabrication methods, including standard photo-lithography, that one can employ to structure such materials as part of a D^2NN design; for example, in THz wavelengths, recent research has demonstrated inkjet printing of graphene saturable absorbers (*39*). Graphene-based saturable absorbers are further advantageous since they work well even at relatively low modulation intensities (*40*).

Another promising avenue to bring non-linear optical properties into D^2NN designs is to use nonlinear metamaterials. These materials have the potential to be integrated with diffractive networks owing to their compactness and the fact that they can be manufactured with standard fabrication processes. While a significant part of the previous work in the field has focused on second and third harmonic generation, recent studies have demonstrated very strong optical Kerr effect for different parts of the electromagnetic spectrum (*41-42*), which can be incorporated into our deep diffractive neural network architecture to bring all-optical nonlinearity into its operation.

Finally, one can also use the DC electro-optic effect to introduce optical nonlinearity into the layers of a D^2NN although this would deviate from all-optical operation of the device and require a DC electric-field for each layer of the diffractive neural network. This electric-field can be externally applied to each layer of a D^2NN; alternatively one can also use poled materials with very strong built-in electric fields as part of the material (e.g., poled crystals or glasses). The latter will still be all-optical in its operation, without the need for an external DC field.

To summarize, there are several practical approaches that can be integrated with diffractive neural networks to bring physical all-optical nonlinearity to D^2NN designs.

Supplementary Figures
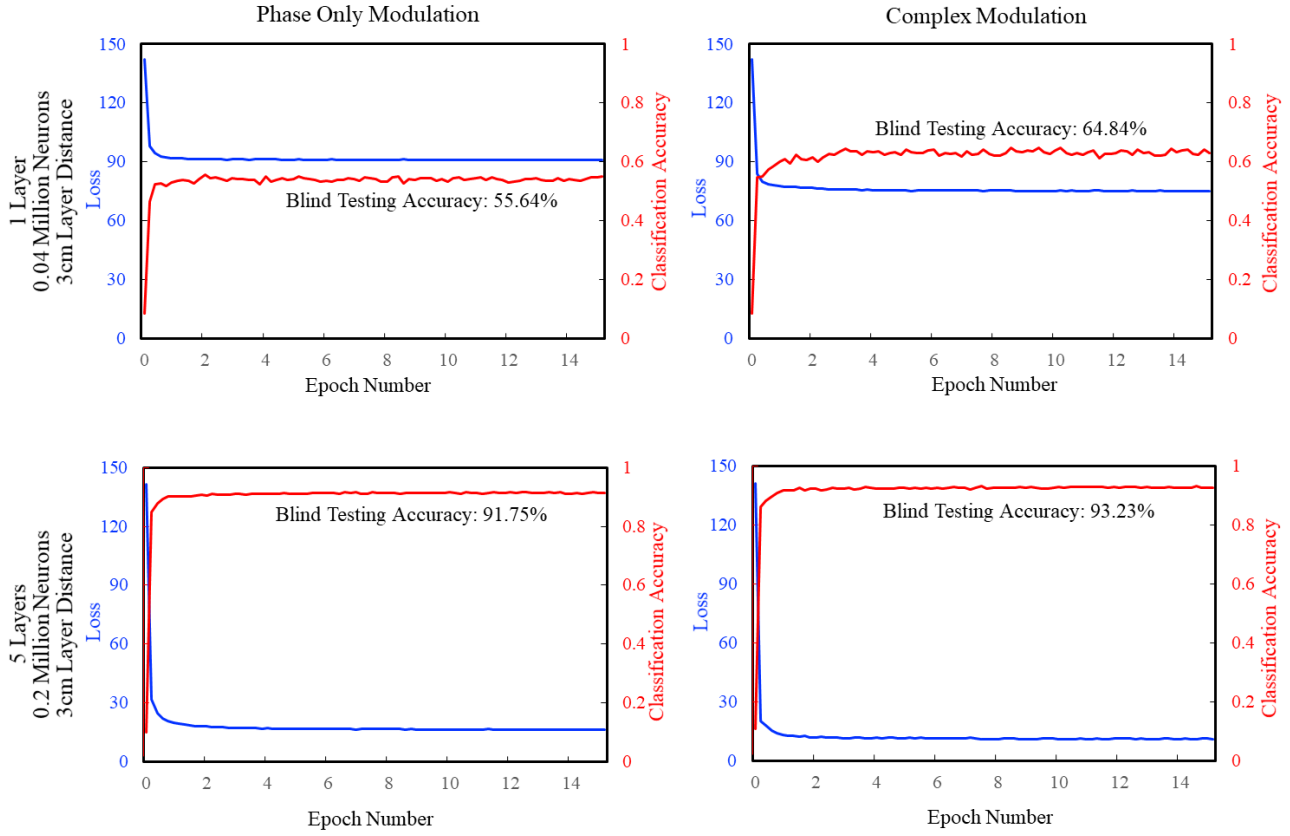
Figure S1: MNIST training convergence plots of a phase-only modulation D^2NN (left column) and a complex-valued (i.e., phase and amplitude) modulation D^2NN (right column) as a function of the number of diffractive layers (N = 1 and 5) and the number of neurons used in the network. The y-axis values in each plot report the MNIST digit classification accuracy and the loss values as a function of the epoch number for the testing datasets. For the same number of diffractive layers, using complex-valued modulation and increasing the spacing between each layer increase the number of connections of the diffractive network, further helping to improve its inference success (also see Fig. S4, top two rows). For N=1, layer distance (3cm) refers to the distance between the sample/output plane and the diffractive layer. The same physical neuron size was used in each case, matching the MNIST D^2NN design reported in our main text. For each class, the detector width was 4.8 mm. We also obtained similar conclusions for the Fashion MNIST dataset results reported in Fig. S4.
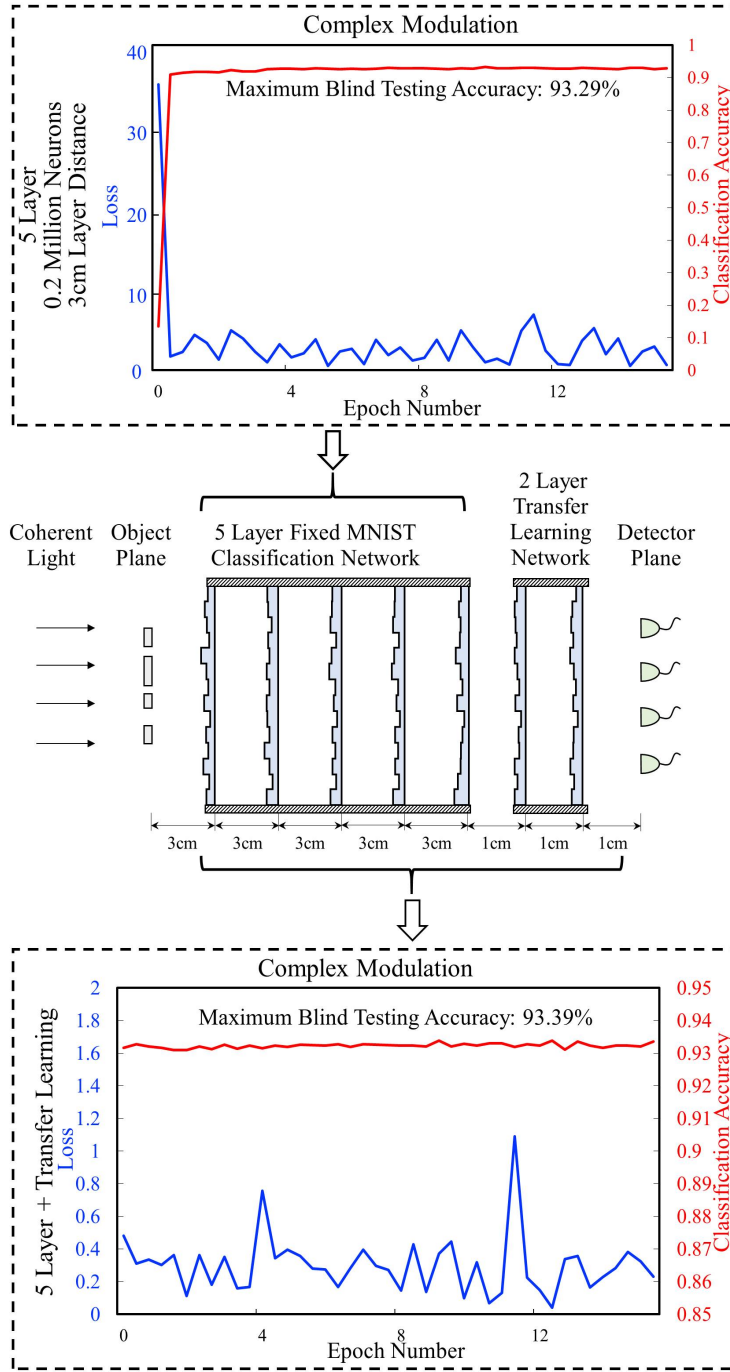
Figure S2: (**Top**) MNIST training convergence plot of a complex-valued modulation D²NN for N = 5 layers and 0.2 million neurons in total. The y-axis values report the MNIST digit classification accuracy and the loss values as a function of the epoch number for the testing dataset. (**Middle**) We illustrate a Lego-like physical transfer learning behavior for D²NN framework, i.e., additional layers are patched to an existing D²NN to improve its inference performance. In this example shown here, we trained 2 additional layers that were placed right at the

exit of an existing (i.e., fixed) 5-layer D²NN. (**Bottom**) After the training of the additional 2 layers, the inference success of the resulting "patched" diffractive neural network has reached 93.39% for MNIST testing dataset. For each class, the detector width was 0.8 mm. Also see fig. S16 for a comparison of detector widths.
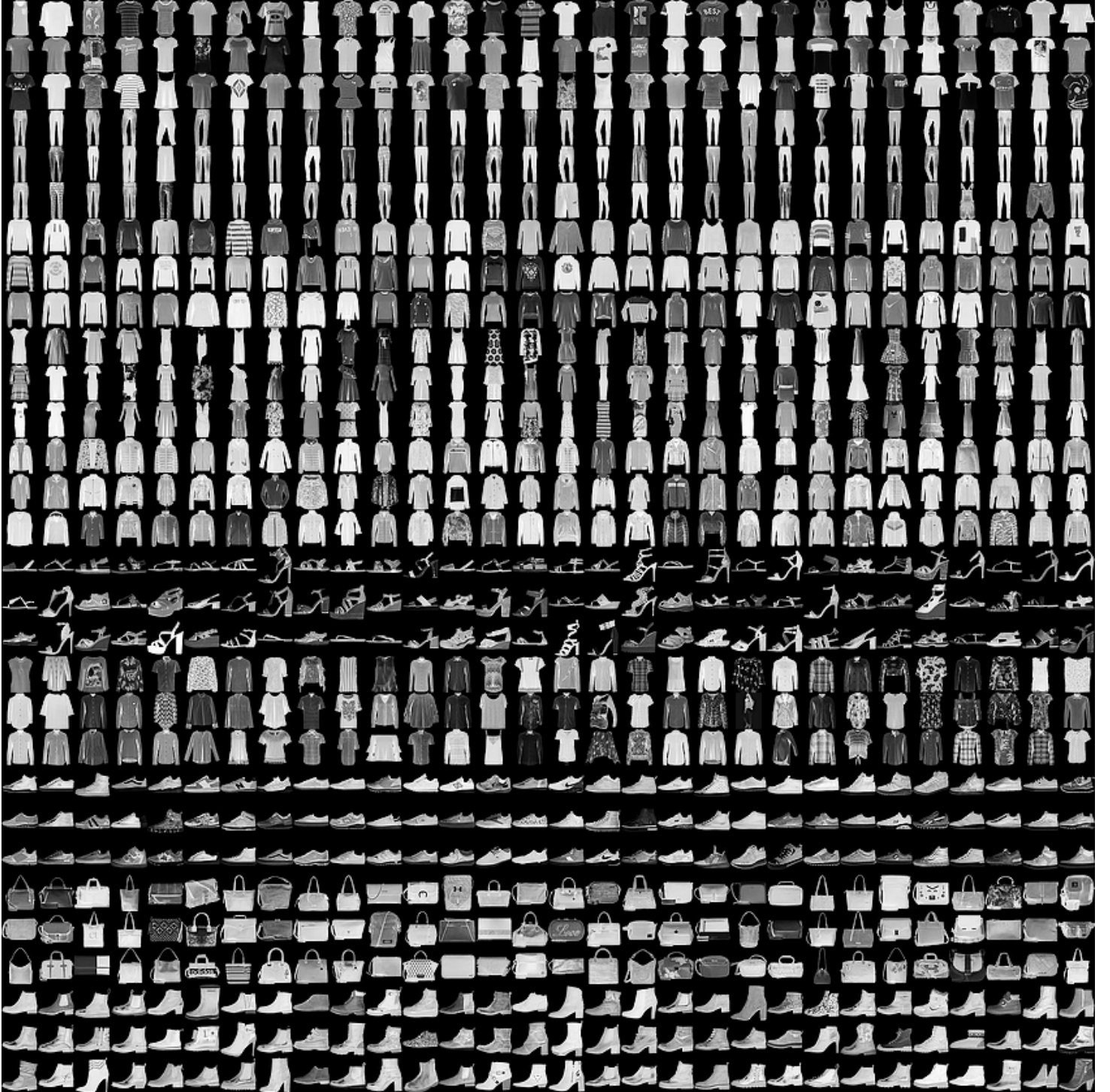
Figure S3: Some sample images for each class of the Fashion MNIST dataset.
https://github.com/zalandoresearch/fashion-mnist
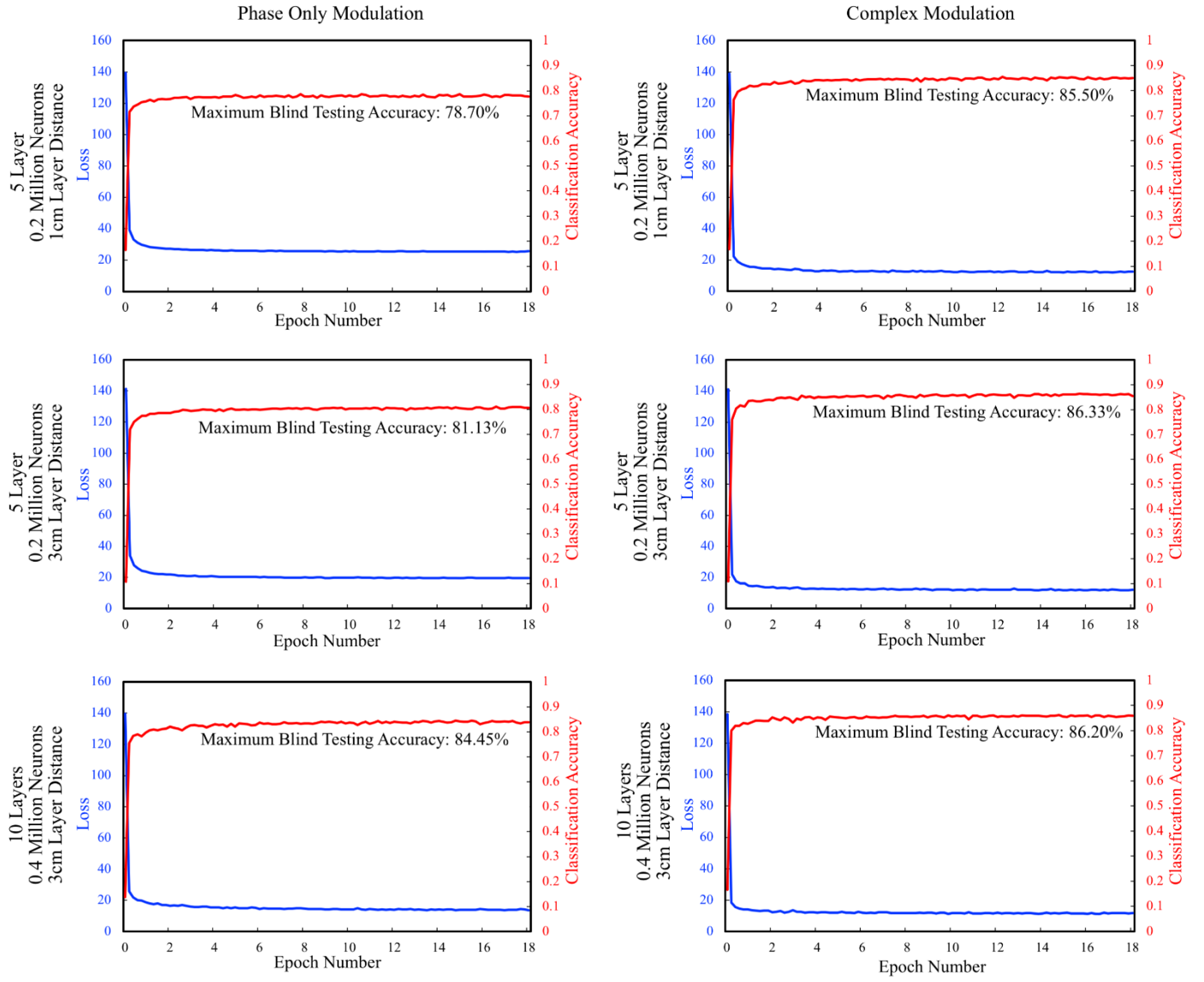
Figure S4: Fashion MNIST results achieved with D²NN framework. Training convergence plots of phase-only as well as complex-valued modulation D²NNs (for N=5 and N=10 layers). The y-axis values in each plot report the Fashion MNIST classification accuracy and the loss values as a function of the epoch number for the testing datasets. The 1ˢᵗ row and 2ⁿᵈ row refer to the same diffractive neural network design (N=5 and 0.2 million neurons in total), except with one difference, the physical space between the layers: 1 cm vs. 3cm, respectively, which affects the number of connections in the network. As expected, the fully connected networks (with 3cm layer-to-layer distance) have better inference performance compared to the 1ˢᵗ row that has 1cm layer-to-layer distance. For each class, the detector width was 4.8 mm.
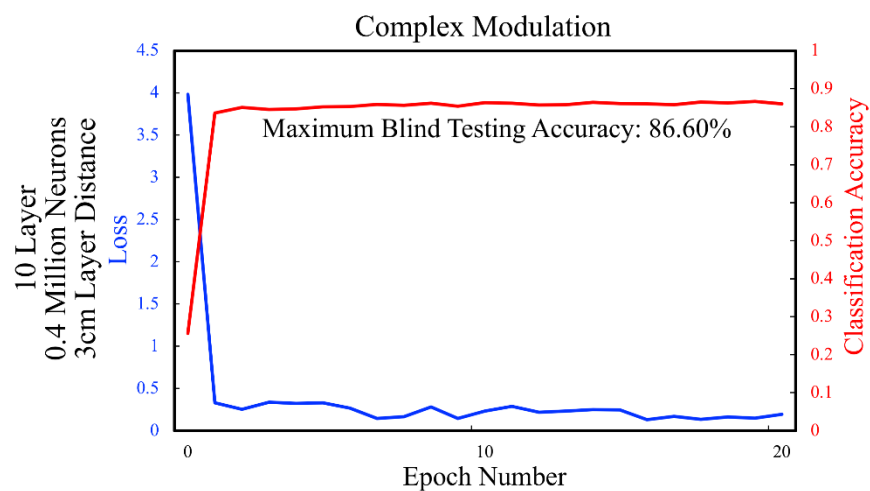
Figure S5. Convergence plot of a complex-valued modulation D^{2}NN (for N=10 and 0.4 million neurons in total) for Fashion MNIST classification that achieves a blind testing accuracy of 86.60%. For each class, the detector width was 0.8 mm.
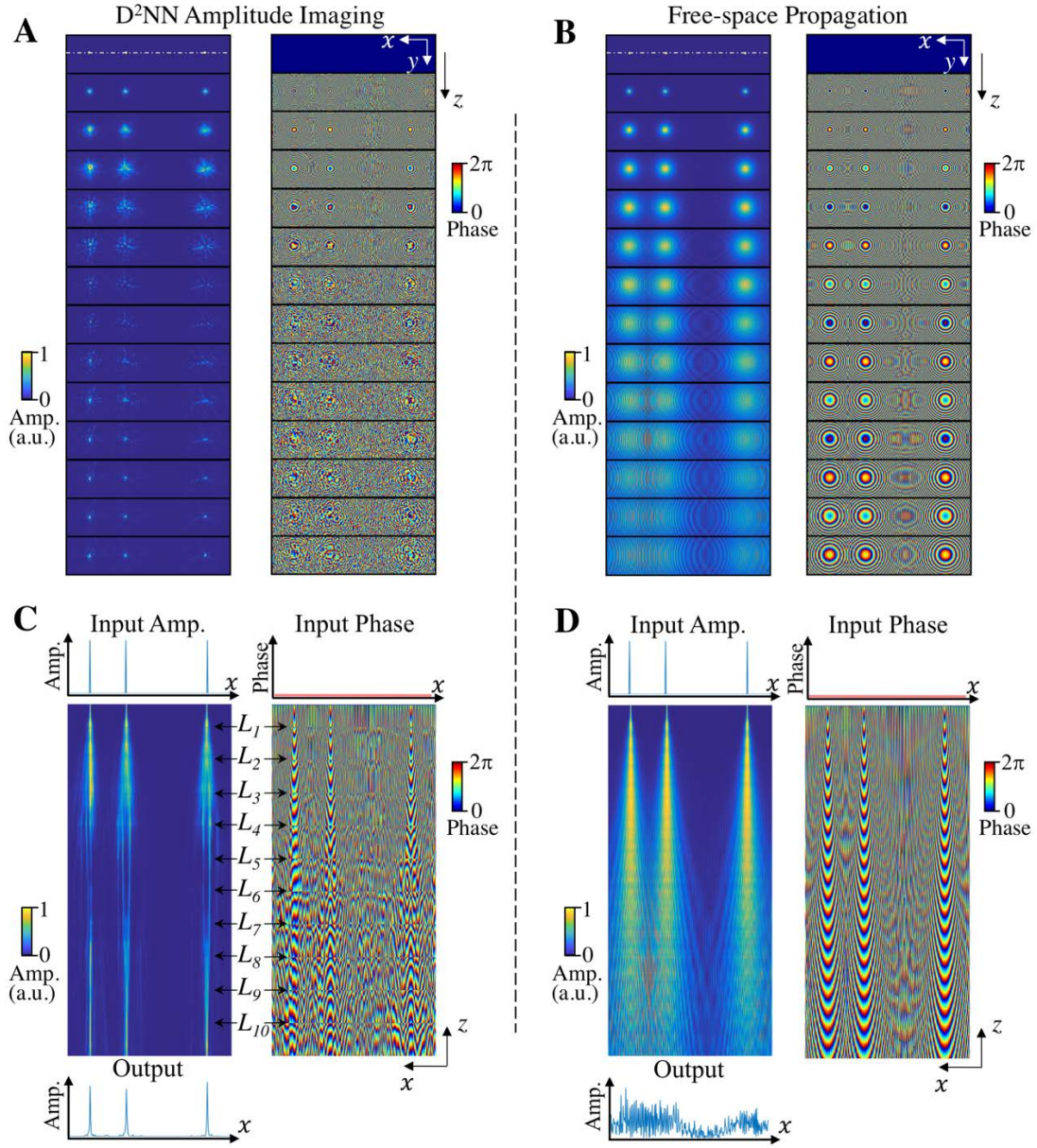
Figure S6: Wave propagation within an imaging D²NN. (**A, C**) To provide insights to the operation principles of a D²NN, we show the amplitude and phase information of the wave that is propagating within a D²NN, trained for amplitude imaging. The object was composed of 3 Dirac-delta functions spread in *x* direction. (**B, D**) Same as in (**A, C**), except without the D²NN. '*L*' refers to each diffractive layer of the network. (**C**) and (**D**) show the cross-sectional view along the *z* direction indicated by the dashed lines in (**A**) and (**B**), respectively.

A

Unit: mm

THZ Wavelength

0.10
0.40
0.70
0.75
1.00

Total Distance = 30mm

SSIM vs D^{2}NN Layer Numbers

Mask Number = 5

SSIM vs D^{2}NN Layer Distance (mm)

B

Amp. (a.u.)

Input Object   2cm

d_0 d_1 d_2 d_3 d_4 d_5

Input Plane L_1 L_2 L_3 L_4 L_5 Output Plane

3D Printed Layers

SSIM = 0.69

D^{2}NN (Amp. Imaging)

SSIM = 0.34

Free-space

C

1cm

4.2mm
3.6mm
3.0mm
2.4mm
1.8mm
1.2mm

Input Resolution Chart

D^{2}NN (Amp. Imaging)

D^{2}NN (With Errors)

Free-space

Figure S7: Design of a transmissive D^{2}NN as an imaging lens. (**A**) The performance of the imaging lens D^{2}NN is optimized by tuning the physical layout of its architecture, including the number of layers (left) and the axial distance between the two consecutive layers (right). SSIM (structural similarity index) was used in this analysis, and we selected 5 layers with an axial distance of 4mm between two successive layers in order to maximize the network performance, while also minimizing its structural complexity. (**B**) After the selection of the optimal neural network layout, the D^{2}NN was trained using ImageNet dataset. After its training, we blindly evaluated the performance of the resulting D^{2}NN with test images to demonstrate its success in imaging arbitrary input objects. (**C**) Blind testing results revealed that the trained D^{2}NN can resolve at its output plane a linewidth of 1.2 mm. As

shown in the 3rd image on the right (D^{2}NN With Errors), the Poisson surface reconstruction errors, absorption related losses at different layers and a random misalignment error of 0.1 mm for each layer of the network design have little effect on the imaging performance of the D^{2}NN. For comparison, the last image on the right shows the diffracted image at the output plane, without the presence of the D^{2}NN.
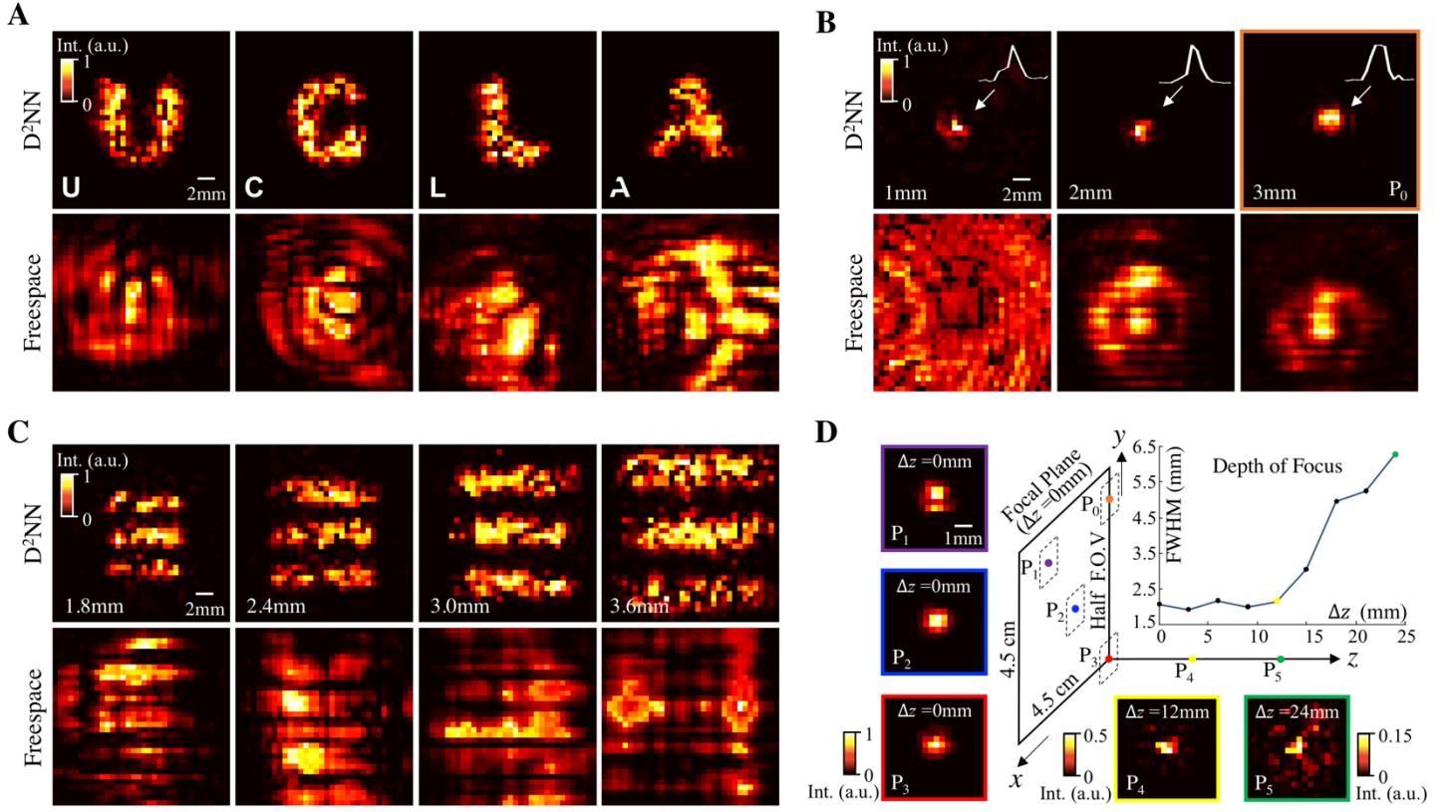
Figure S8: Experimental results for imaging lens D²NN. (A) Output images of the 3D-printed lens D²NN are shown for different input objects: 'U', 'C', 'L' and 'A'. To be able to 3D-print letter 'A', the letter was slightly modified as shown in the bottom-left corner of the corresponding image panel. For comparison, free-space diffraction results corresponding to the same objects, achieved over the same sample-output plane distance (29.5 mm) without the 3D-printed network, are also shown. **(B)** Same as in **(A)**, except the input objects were pinholes with diameters of 1 mm, 2 mm and 3 mm. **(C)** D²NN can resolve a line-width of 1.8 mm at its output plane. **(D)** Using a 3-mm pinhole that is scanned in front of the 3D-printed network, we evaluated the tolerance of the physical D²NN as a function of the axial distance. For four different locations on the input plane of the network, i.e., P_1-P_3, in **(D)** and P_0 in **(B)**, we obtained very similar output images for the same 3-mm pinhole. The 3D-printed network was found to be robust to axis defocusing up to ~12 mm from the input plane. While there are various other powerful methods to design lenses (*43-45*), the main point of these results is the introduction of the diffractive neural network as an all-optical machine learning engine that is scalable and power-efficient to implement various functions using passive optical components, which present large degrees of freedom that can be learned through training data.

30
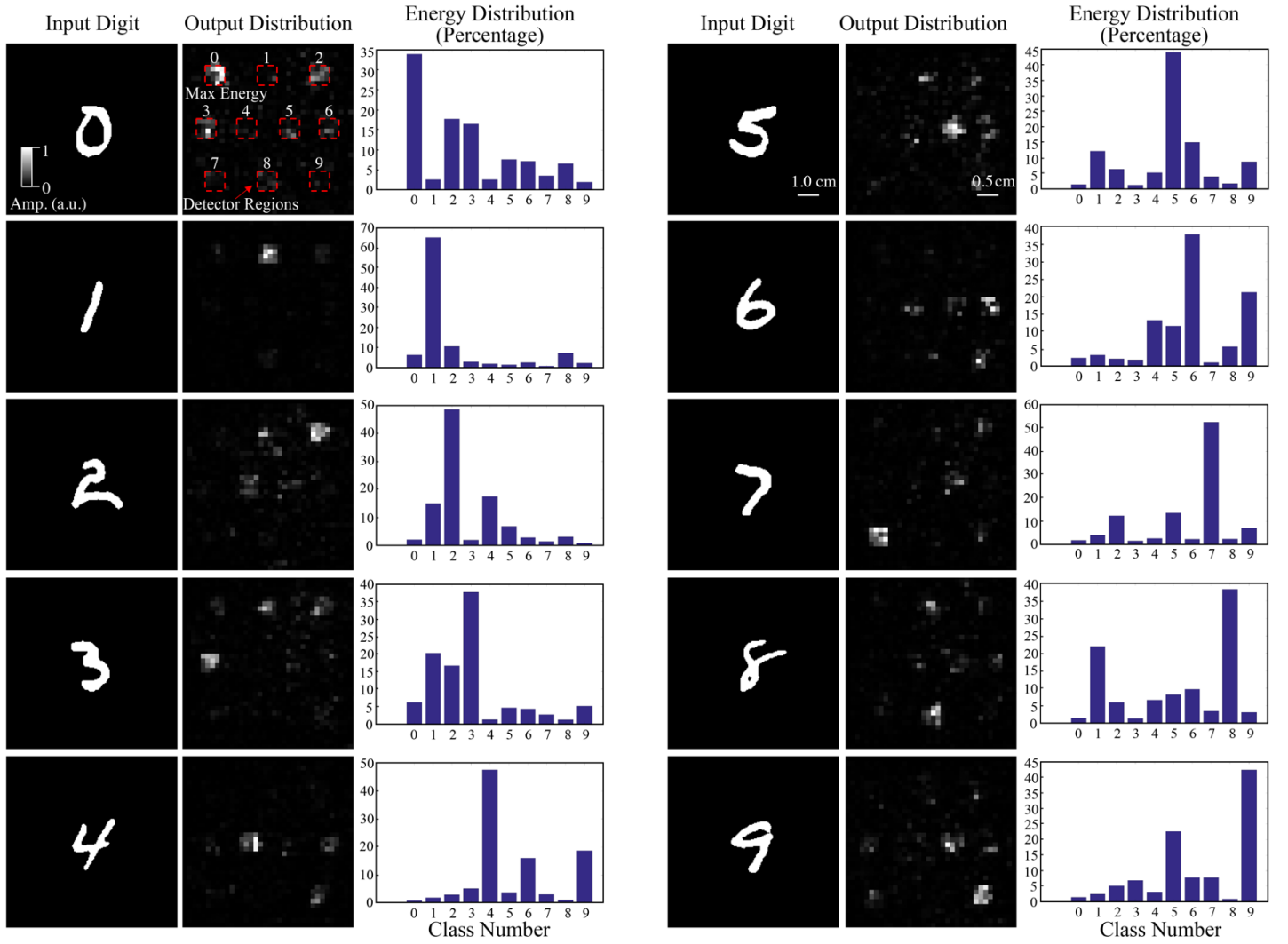
Figure S9: Sample experimental results for digit classifier D²NN. Summary of some of the experimental results achieved with our 3D-printed handwritten digit classification D²NN. The energy distribution percentage corresponding to each digit at the output plane shows that D²NN has the maximum energy focused on the target detector region of each digit (also see Fig. 3 of the main text).
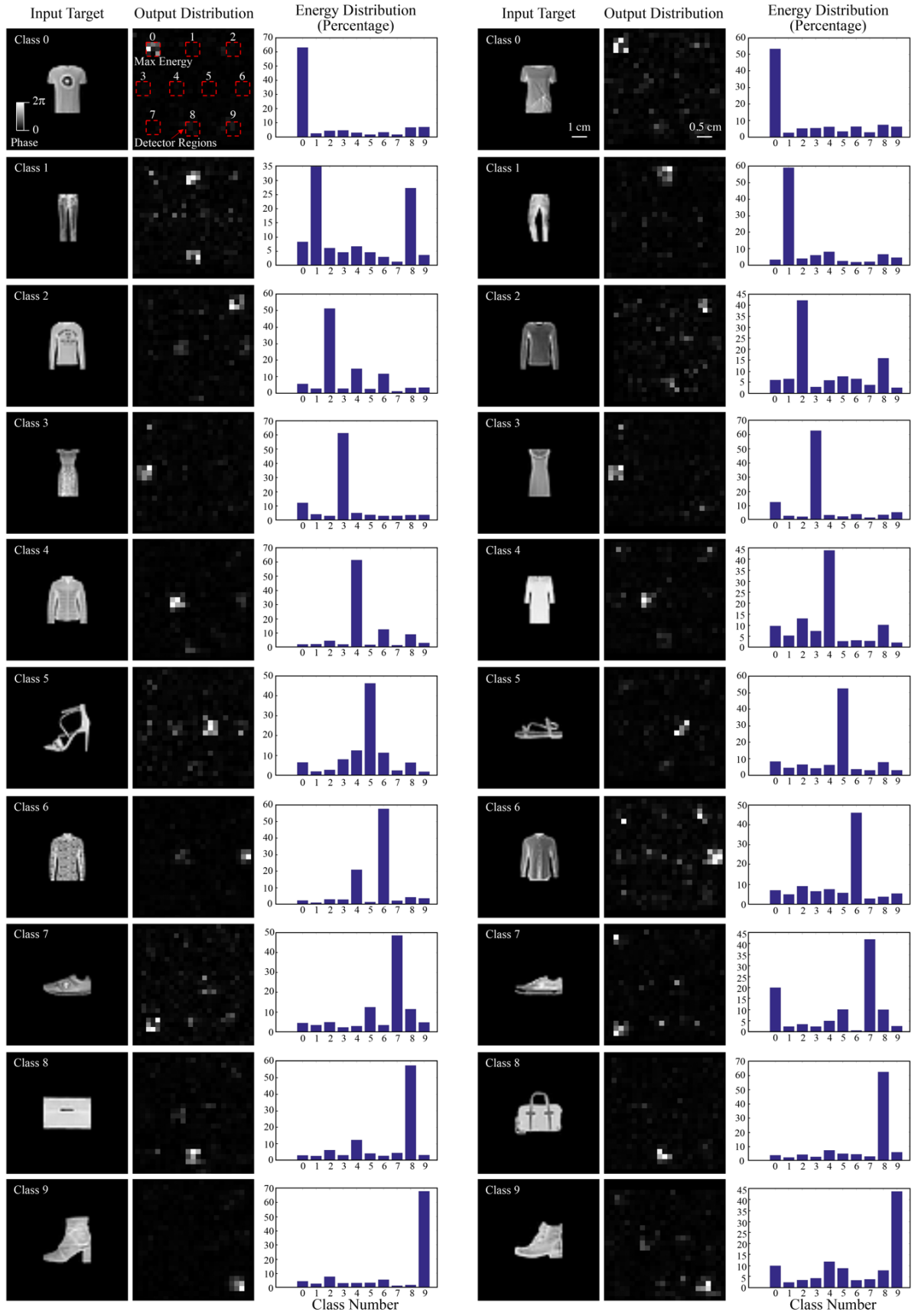
Figure S10: Sample experimental results for fashion product classifier D²NN. Summary of some of the experimental results achieved with our 3D-printed fashion product classification D²NN. The energy distribution

32

percentage corresponding to each product at the output plane shows that D^{2}NN has the maximum energy focused

on the target detector region of each product (also see Fig. 4 of the main text)
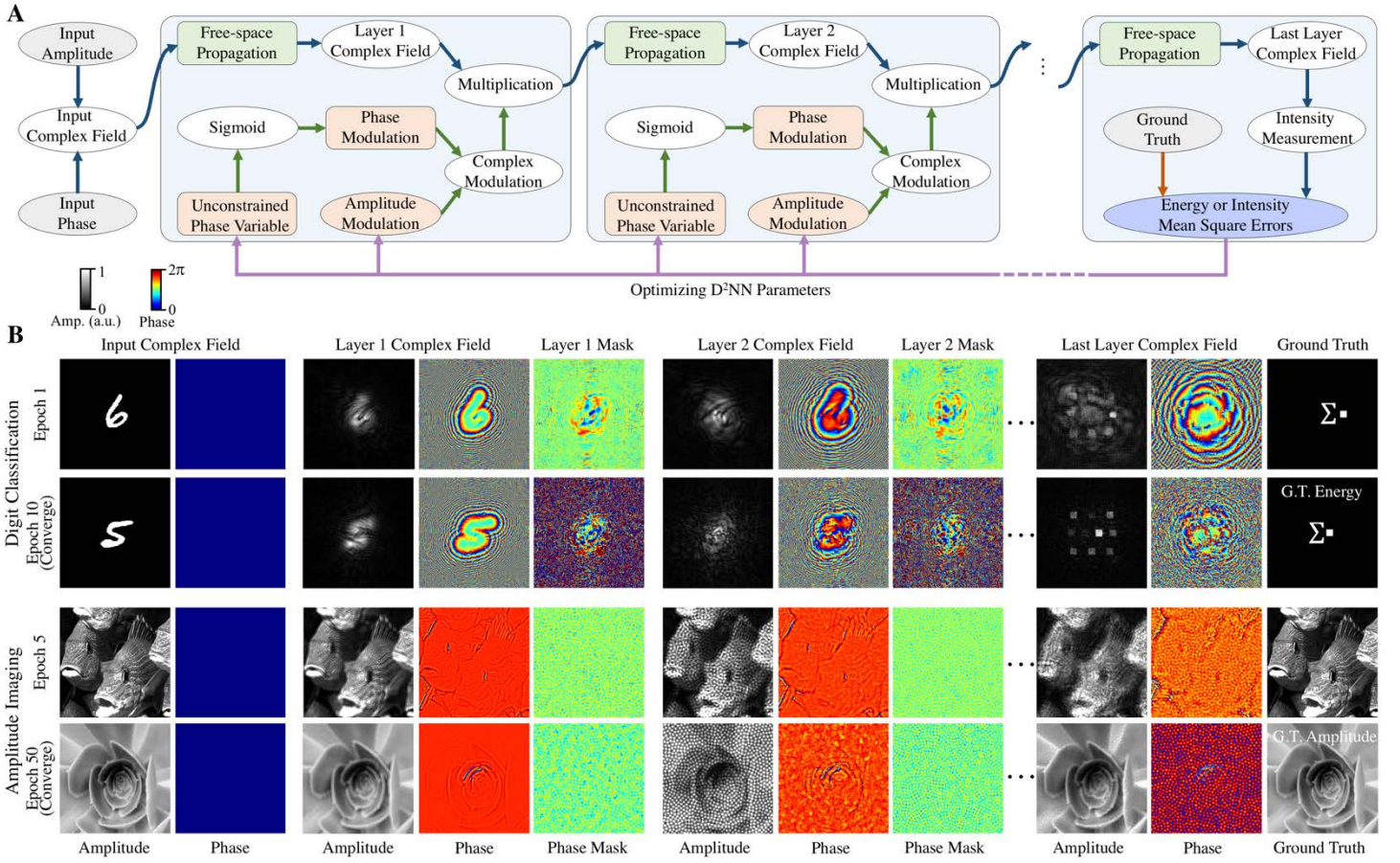
Figure S11: TensorFlow implementation of a diffractive deep neural network. (**A**) The resulting complex field of free-space propagated field is multiplied with a complex modulator at each layer and is then transferred to the next layer. To help with the 3D-printing and fabrication of the D^{2}NN design, a sigmoid function was used to constrain the phase value of each neuron. (**B**) MNIST and ImageNet datasets were used to train the D^{2}NNs for handwritten digit classification and imaging lens tasks, respectively. Fashion MNIST dataset was used for training the fashion product classifier D^{2}NN. The resulting complex fields and phase patterns of each layer are demonstrated at different epochs of the training phase.
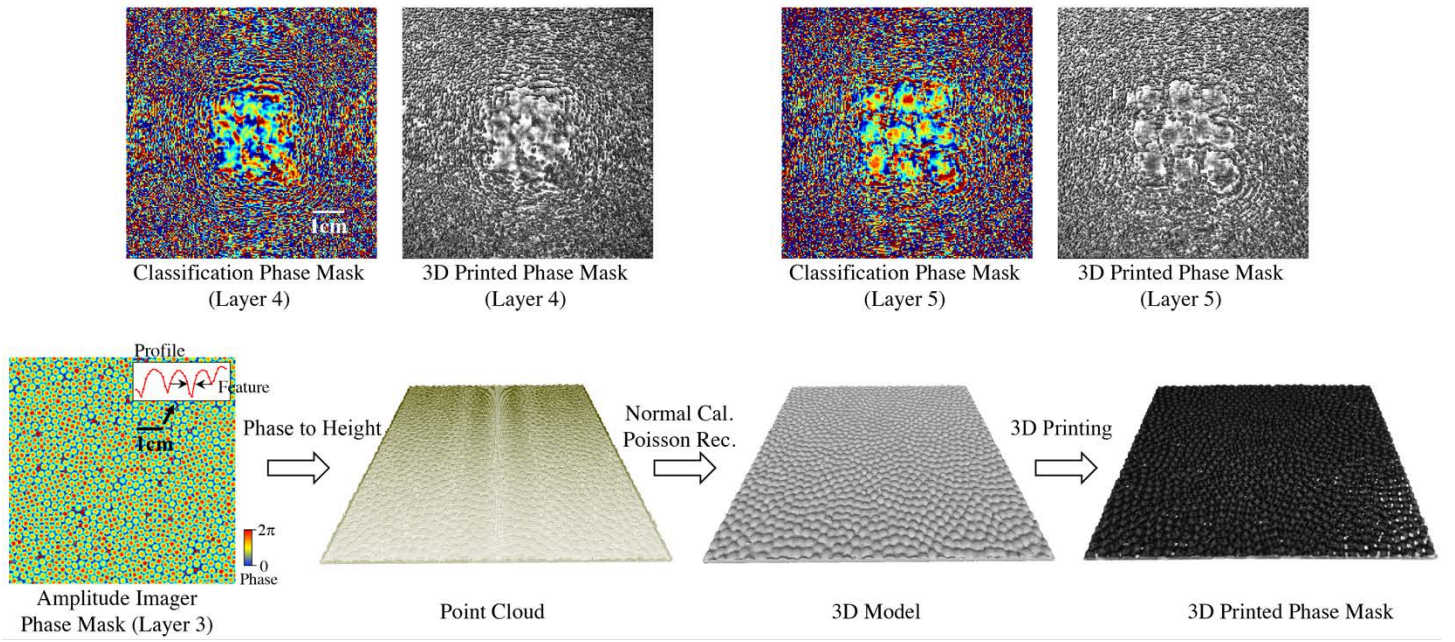
Figure S12: 3D model reconstruction of a D^{2}NN layer for 3D-printing. We apply Poisson surface reconstruction to generate the 3D model of each D^{2}NN layer for 3D printing. The phase mask is first converted to a height map with the knowledge of the material refractive index, and the enclosed point cloud is formed by adding the substrate points. The 3D model is then generated by calculating the surface normal and performing the Poisson reconstruction. The final step is the 3D-printing of the D^{2}NN model.
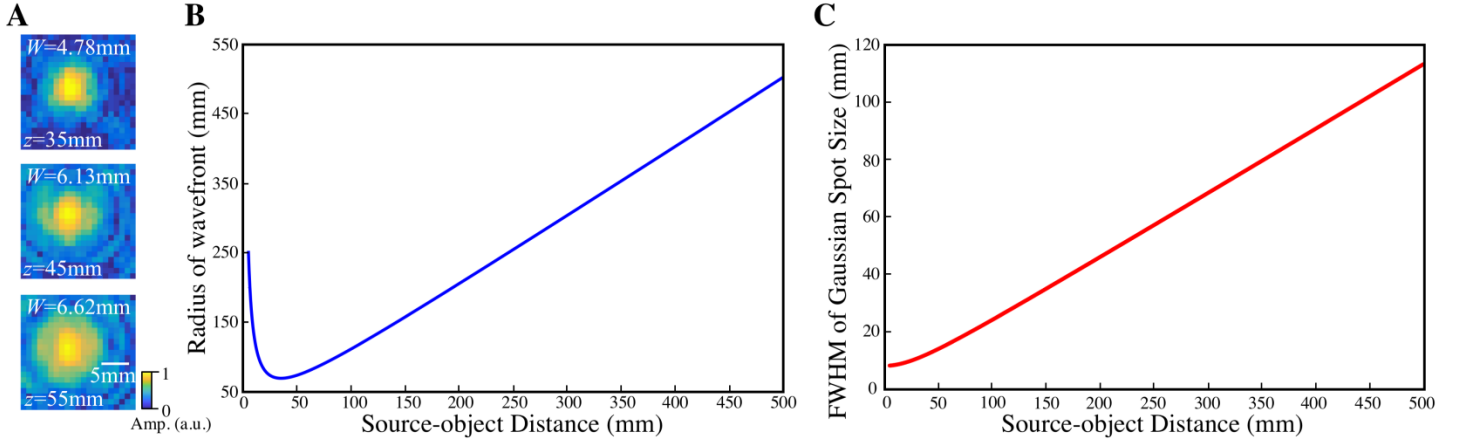
Figure S13: Terahertz Source Characterization. (**A**) Beam profiles were imaged at three different axial locations to quantify the beam parameters, based on which the Terahertz light source can be approximated as a Gaussian beam. (**B, C**) The plots show the radius of the source wavefront and its FWHM as a function of the source-object distance. For all the 3D-printed D^2NN designs of this work, the illumination at the object/input plane can be approximated as a plane wave.
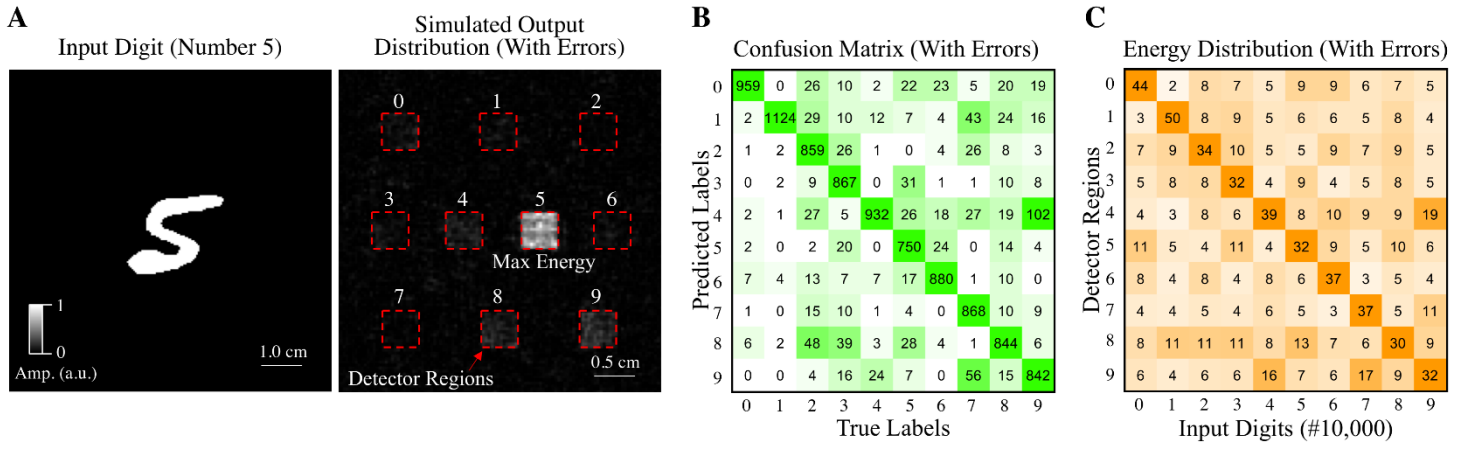
Figure S14: Numerical Test Results of the Digit Classifier D²NN Including Error Sources. (**A**) As an example, the output image of the digit classifier D²NN for a handwritten input of "5" is demonstrated, where the red squares represent the trained detector regions for each digit. (**B, C**) are the same as in Fig. 3C of the main text, except they now take into account the Poisson surface reconstruction errors, absorption related losses at different layers and a random misalignment error of 0.1 mm for each layer of the network design. All these sources of error reduced the overall performance of the diffractive network's digit classification accuracy from 91.75% (Fig. 3C) to 89.25%, evaluated over 10,000 different handwritten digits (i.e., approximately 1,000 for each digit).
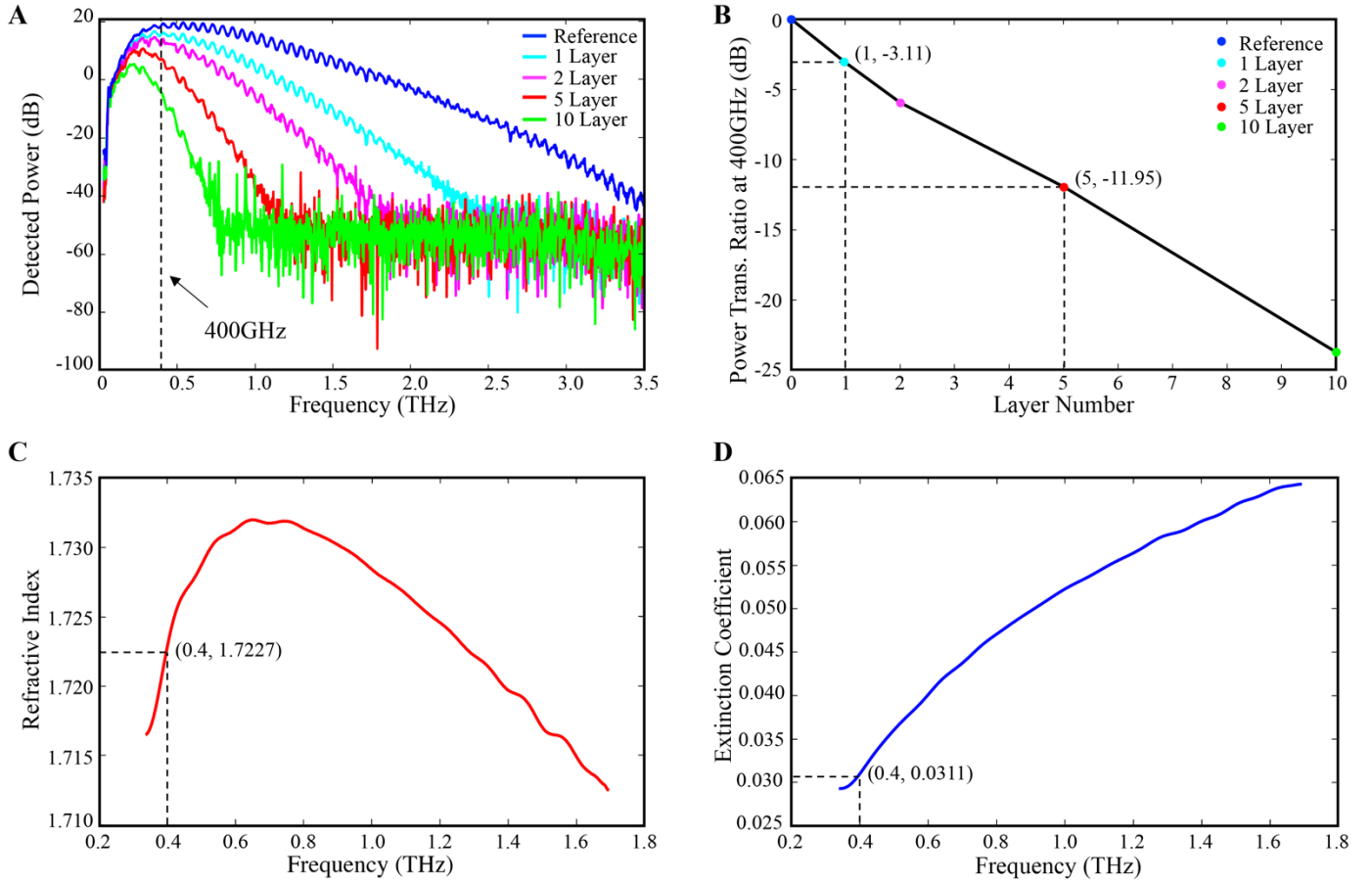
Figure S15: Characterization of the 3D-printing material properties. (**A**) Our 3D-printing material (VeroBlackPlus RGD875) was characterized with a terahertz time-domain spectroscopy setup (*46*). 1 mm-thick plastic layers were placed between the terahertz emitter and detector, and the transmitted field from the plastic layers was measured. The Fourier transform of the detected field was taken to calculate the detected power as a function of the frequency. The detected power levels for different numbers of 3D-printed layers are shown, revealing that the material loss increases at higher frequencies. Reference signal shows the detected power without any plastic layers on the beam path. (**B**) The power transmission ratio as a function of the number of layers is shown. The light transmission efficiency of a single 1mm-thick 3D-printed layer is $10^{-3.11/10} = 48.87\%$, and it drops to $10^{-11.95/10} = 6.38\%$ for five 1mm-thick 3D-printed layers. (**C, D**) At 0.4 THz, the refractive index and the extinction coefficient of the 3D-printing material can be calculated as 1.7227 and 0.0311, respectively. These numbers were used in the design and training of each D^2NN so that the final 3D-printed network works as designed.
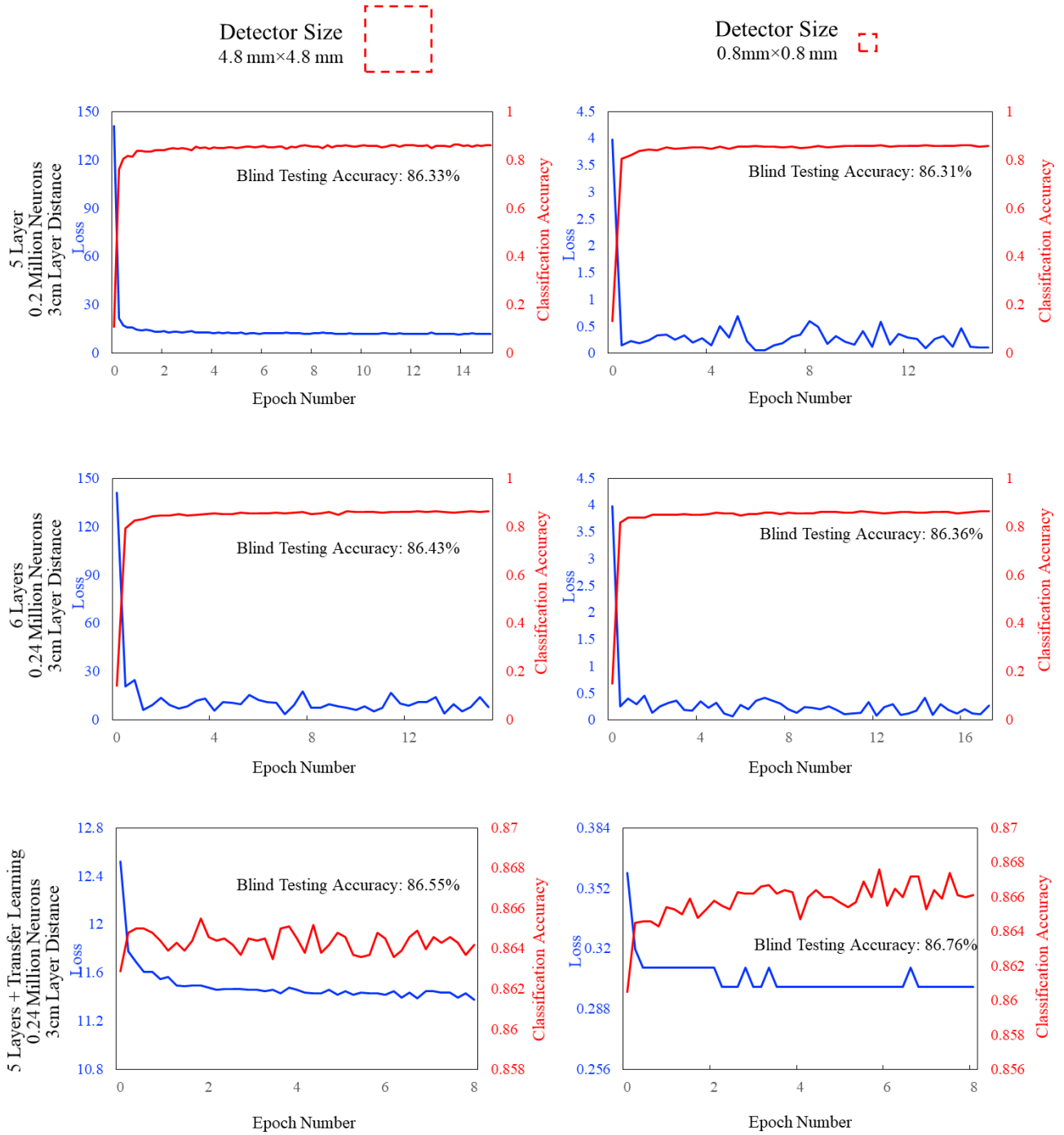
Figure S16: Fashion MNIST results achieved with complex-valued D²NN framework (also see Figs. S4 and S5). Convergence plots of D²NNs (top and middle plots for N=5 and N=6, respectively) are shown. Bottom plots show the case for training only the 6th layer, where the first 5 layers of the network were fixed (i.e., identical to the design resulting from the top case, N=5) and the new layer was added between the 5th layer and the detector plane,

at equal distance from both. The layers of the N=5 and N=6 designs were separated by 3 cm from each other and the detector plane. The y-axis values in each plot report the Fashion MNIST classification accuracy and the loss values as a function of the epoch number for the training datasets. Addition of the 6th layer (learnable) to an already trained and fixed D^{2}NN with N=5 improves its inference performance, performing slightly better than the performance of N=6 (middle plots). Also see Fig. S2.

References and Notes

1. Y. LeCun, Y. Bengio, G. Hinton, Deep learning. *Nature* **521**, 436–444 (2015). doi:10.1038/nature14539 Medline

2. G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, C. I. Sánchez, A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017). doi:10.1016/j.media.2017.07.005 Medline

3. A. Graves, A. Mohamed, G. Hinton, in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (IEEE, 2013), pp. 6645–6649.

4. K. Cho *et al*., Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078 [cs.CL] (3 June 2014).

5. A. Krizhevsky, I. Sutskever, G. E. Hinton, in *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger, Eds. (Curran Associates, 2012), pp. 1097–1105.

6. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, D. Hassabis, Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016). doi:10.1038/nature16961 Medline

7. U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, D. Psaltis, Learning approach to optical tomography. *Optica* **2**, 517 (2015). doi:10.1364/OPTICA.2.000517

8. Y. Rivenson, Z. Göröcs, H. Günaydin, Y. Zhang, H. Wang, A. Ozcan, Deep learning microscopy. *Optica* **4**, 1437 (2017). doi:10.1364/OPTICA.4.001437

9. K. H. Jin, M. T. McCann, E. Froustey, M. Unser, Deep convolutional neural network for inverse problems in imaging. *IEEE Trans. Image Process.* **26**, 4509–4522 (2017). doi:10.1109/TIP.2017.2713099 Medline

10. Y. Rivenson, Y. Zhang, H. Gunaydin, D. Teng, A. Ozcan, Phase recovery and holographic image reconstruction using deep learning in neural networks. *Light Sci. Appl.* **7**, 17141 (2018). doi:10.1038/lsa.2017.141

11. A. Sinha, J. Lee, S. Li, G. Barbastathis, Lensless computational imaging through deep learning. *Optica* **4**, 1117 (2017). doi:10.1364/OPTICA.4.001117

12. K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, F. Knoll, Learning a variational network for reconstruction of accelerated MRI data. *Magn. Reson. Med.* **79**, 3055–3071 (2018). doi:10.1002/mrm.26977 Medline

13. Y. Rivenson, H. Ceylan Koydemir, H. Wang, Z. Wei, Z. Ren, H. Günaydın, Y. Zhang, Z. Göröcs, K. Liang, D. Tseng, A. Ozcan, Deep learning enhanced mobile-phone microscopy. *ACS Photonics* **5**, 2354–2364 (2018). doi:10.1021/acsphotonics.8b00146

14. Materials and methods are available as supplementary materials.

15. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2324 (1998). doi:10.1109/5.726791

16. D. Ciregan, U. Meier, J. Schmidhuber, in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2012), pp. 3642–3649.

17. C.-Y. Lee, P. W. Gallagher, Z. Tu, in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics* (AISTATS, 2016), pp. 464–472.

18. M. A. Ranzato, C. Poultney, S. Chopra, Y. LeCun, in *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, J. C. Platt, D. Koller, Y. Singer, S. T. Roweis, Eds. (Curran Associates, 2007), pp. 1137–1144.

19. H. Xiao, K. Rasul, R. Vollgraf, Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747 [cs.LG] (25 August 2017).

20. S. Zeng, B. Zhang, J. Gou, Deep Collaborative Weight-based Classification. arXiv:1802.07589 [cs.CV] (21 February 2018).

21. ImageNet database, www.image-net.org.

22. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, M. Soljačić, Deep learning with coherent nanophotonic circuits. *Nat. Photonics* **11**, 441–446 (2017). doi:10.1038/nphoton.2017.93

23. D. Psaltis, D. Brady, X.-G. Gu, S. Lin, Holography in artificial neural networks. *Nature* **343**, 325–330 (1990). doi:10.1038/343325a0 Medline

24. K. H. Wagner, in *Proceedings of Frontiers in Optics 2017* (OSA, 2017), FW2C.1.

25. B. J. Shastri *et al*., Principles of neuromorphic photonics. arXiv:1801.00016 [cs.ET] (29 December 2017).

26. M. Hermans, M. Burm, T. Van Vaerenbergh, J. Dambre, P. Bienstman, Trainable hardware for dynamical computing using error backpropagation through physical media. *Nat. Commun.* **6**, 6729 (2015). doi:10.1038/ncomms7729 Medline

27. D. Brunner, M. C. Soriano, C. R. Mirasso, I. Fischer, Parallel photonic information processing at gigabyte per second data rates using transient states. *Nat. Commun.* **4**, 1364 (2013). doi:10.1038/ncomms2368 Medline

28. A. Greenbaum, W. Luo, T.-W. Su, Z. Göröcs, L. Xue, S. O. Isikman, A. F. Coskun, O. Mudanyali, A. Ozcan, Imaging without lenses: Achievements and remaining challenges of wide-field on-chip microscopy. *Nat. Methods* **9**, 889–895 (2012). doi:10.1038/nmeth.2114 Medline

29. A. Ozcan, E. McLeod, Lensless imaging and sensing. *Annu. Rev. Biomed. Eng.* **18**, 77–102 (2016). doi:10.1146/annurev-bioeng-092515-010849 Medline

30. M. Emons, K. Obata, T. Binhammer, A. Ovsianikov, B. N. Chichkov, U. Morgner, Two-photon polymerization technique with sub-50 nm resolution by sub-10 fs laser pulses. *Opt. Mater. Express* **2**, 942–947 (2012). doi:10.1364/OME.2.000942

31. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. arXiv:1412.6980 [cs.LG] (22 December 2014).

32. M. Kazhdan, H. Hoppe, Screened poisson surface reconstruction. *ACM Trans. Graph.* **32**, 1–13 (2013). doi:10.1145/2487228.2487237

33. MeshLab, www.meshlab.net.

34. P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia, "MeshLab: An open-source mesh processing tool" in *Sixth Eurographics Italian Chapter Conference* (2008), pp. 129–136.

35. J. W. Goodman, *Introduction to Fourier Optics* (Roberts and Company, 2005).

36. C. Trabelsi *et al*., Deep complex networks. arXiv:1705.09792 [cs.NE] (27 May 2017).

37. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **13**, 600–612 (2004). doi:10.1109/TIP.2003.819861 Medline

38. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).

39. V. Bianchi, T. Carey, L. Viti, L. Li, E. H. Linfield, A. G. Davies, A. Tredicucci, D. Yoon, P. G. Karagiannidis, L. Lombardi, F. Tomarchio, A. C. Ferrari, F. Torrisi, M. S. Vitiello, Terahertz saturable absorbers from liquid phase exfoliation of graphite. *Nat. Commun.* **8**, 15763 (2017). doi:10.1038/ncomms15763 Medline

40. A. Marini, J. D. Cox, F. J. García de Abajo, Theory of graphene saturable absorption. *Phys. Rev. B* **95**, 125408 (2017). doi:10.1103/PhysRevB.95.125408

41. X. Yin, T. Feng, Z. Liang, J. Li, Artificial Kerr-type medium using metamaterials. *Opt. Express* **20**, 8543–8550 (2012). doi:10.1364/OE.20.008543 Medline

42. Y. Xiao, H. Qian, Z. Liu, Nonlinear metasurface based on giant optical kerr response of gold quantum wells. *ACS Photonics* **5**, 1654–1659 (2018). doi:10.1021/acsphotonics.7b01140

43. N. Yu, F. Capasso, Flat optics with designer metasurfaces. *Nat. Mater.* **13**, 139–150 (2014). doi:10.1038/nmat3839 Medline

44. M. Khorasaninejad, W. T. Chen, R. C. Devlin, J. Oh, A. Y. Zhu, F. Capasso, Metalenses at visible wavelengths: Diffraction-limited focusing and subwavelength resolution imaging. *Science* **352**, 1190–1194 (2016). doi:10.1126/science.aaf6644 Medline

45. A. V. Kildishev, A. Boltasseva, V. M. Shalaev, Planar photonics with metasurfaces. *Science* **339**, 1232009 (2013). doi:10.1126/science.1232009 Medline

46. D. Grischkowsky, S. Keiding, M. Van Exter, C. Fattinger, Far-infrared time-domain spectroscopy with terahertz beams of dielectrics and semiconductors. *J. Opt. Soc. Am. B* **7**, 2006–2015 (1990). doi:10.1364/JOSAB.7.002006