Supplementary Information for Thin-film neural networks for optical inverse problem

Lingjie Fan[1,2,†], Ang Chen[2,†], Tongyu Li[1,2,†], Jiao Chu[1], Yang Tang[1], Jiajun Wang[1], Maoxiong Zhao[1,2], Tangyao Shen[1,2], Minjia Zheng[1,2], Fang Guan[3], Haiwei Yin[2], Lei Shi[1,2,3,4,*], and Jian Zi[1,2,3,4,*]

[1]*State Key Laboratory of Surface Physics, Key Laboratory of Micro- and Nano-Photonic Structures (Ministry of Education) and Department of Physics, Fudan University, Shanghai 200433, China*

[2]*Shanghai Engineering Research Center of Optical Metrology for Nano-fabrication (SERCOM), Shanghai 200433, China.*

[3]*Institute for Nanoelectronic devices and Quantum computing, Fudan University, Shanghai 200438, China.*

[4]*Collaborative Innovation Center of Advanced Microsstructures, Nanjing University, Nanjing 210093, China.*

Keywords: neural networks, backpropagation, deep learning, thin-film metrology, optical inverse design.

[†]These authors contributed equally to this work.

[*]Correspondence: lshi@fudan.edu.cn, and jzi@fudan.edu.cn.

Content

Backpropagation process in thin-film neural networks

In this part, we mainly introduce the backpropagation process in thin-film neural networks (TFNNs). The theoretical model of multilayer thin films is shown in Fig. S1. There are n+1 layers in the multilayer thin films. Each layer has the thickness d_i and could be described by the layer matrix $\boldsymbol{M_i} = \boldsymbol{D_{i-1} P_i D_i^{-1}}$. The + and the - signs distinguish between forward and backward field amplitudes. Therefor the optical responses of the multilayer could be described by the product of n+1 matrices:

$$\begin{bmatrix} E_0^+ \\ E_0^- \end{bmatrix} = \boldsymbol{D_0^{-1}} \left[\prod_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{D_i P_i D_i^{-1}} \right] \boldsymbol{D_{n+1}} \begin{bmatrix} E_{n+1}^+ \\ E_{n+1}^- \end{bmatrix}$$
$$= \boldsymbol{M_0 M_1 \cdots M_{n+1}} \begin{bmatrix} E_{n+1}^+ \\ E_{n+1}^- \end{bmatrix} \qquad \text{(S1)}$$

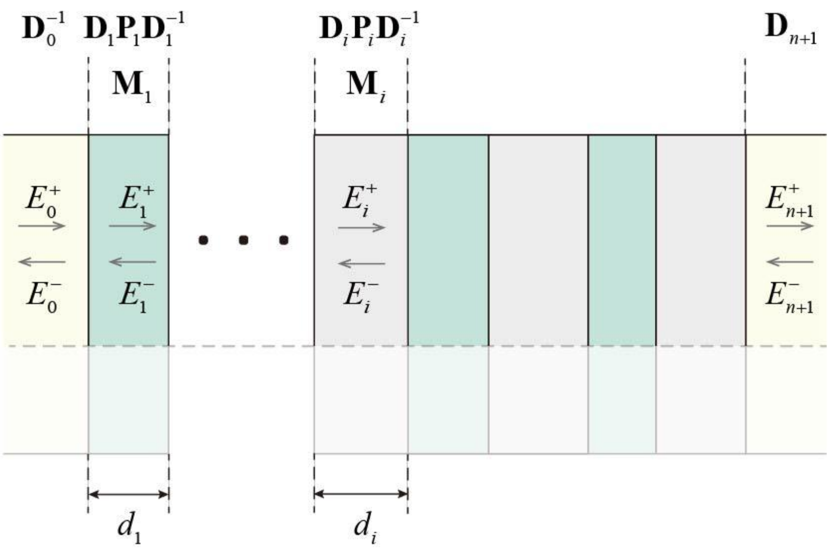

Fig. S1. The schematic view of multilayer thin films.

For the forward propagation process in TFNNs, the proceedings could be viewed as a sequence of matrix multiplier calculations, as illustrated in Eq. (S2) and Fig. S2.

$$
\begin{aligned}
&[\boldsymbol{M_0}\boldsymbol{M_1}\cdots\boldsymbol{M_{i-1}}]\boldsymbol{M_i} \\
&= \begin{bmatrix} A_{i-1} & B_{i-1} \\ C_{i-1} & D_{i-1} \end{bmatrix} \begin{bmatrix} a_{i-1} & b_{i-1} \\ c_{i-1} & d_{i-1} \end{bmatrix} \\
&= \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix} \qquad \text{(S2)}
\end{aligned}
$$

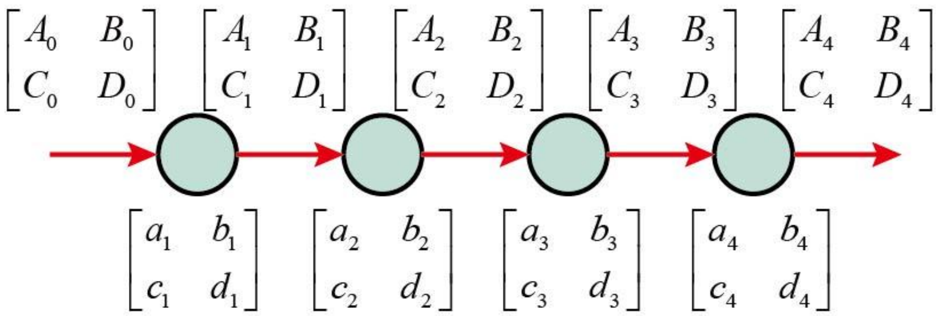

Fig. S2. The forward propagation process in TFNNs.

For the stem of the backpropagation process in TFNNs, the process starts from the Fresnel coefficients r and ended in $\boldsymbol{A_i}$, $\boldsymbol{B_i}$, $\boldsymbol{C_i}$, and $\boldsymbol{D_i}$. And the gradient of each layer is calculated in turn. The question is, given the gradients of the next layer ($\partial r/\partial \boldsymbol{A_{i+1}}$, $\partial r/\partial \boldsymbol{B_{i+1}}$, $\partial r/\partial \boldsymbol{C_{i+1}}$, $\partial r/\partial \boldsymbol{D_{i+1}}$), how to obtain the gradients of this layer ($\partial r/\partial \boldsymbol{A_i}$, $\partial r/\partial \boldsymbol{B_i}$, $\partial r/\partial \boldsymbol{C_i}$, $\partial r/\partial \boldsymbol{D_i}$). The following equations establish the relations in the backpropagation through chain rule as shown in Eq. (S3). The repeated application of the above transformations for the n+1 layers leads to the stem of the backpropagation in TFNNs as illustrated in Fig. S3.

$$
\begin{bmatrix} \frac{\partial r}{\partial A_i} & \frac{\partial r}{\partial B_i} \\ \frac{\partial r}{\partial C_i} & \frac{\partial r}{\partial D_i} \end{bmatrix} = \begin{bmatrix} \frac{\partial r}{\partial A_{i+1}} & \frac{\partial r}{\partial B_{i+1}} \\ \frac{\partial r}{\partial C_{i+1}} & \frac{\partial r}{\partial D_{i+1}} \end{bmatrix} \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix}^T \qquad \text{(S3)}
$$

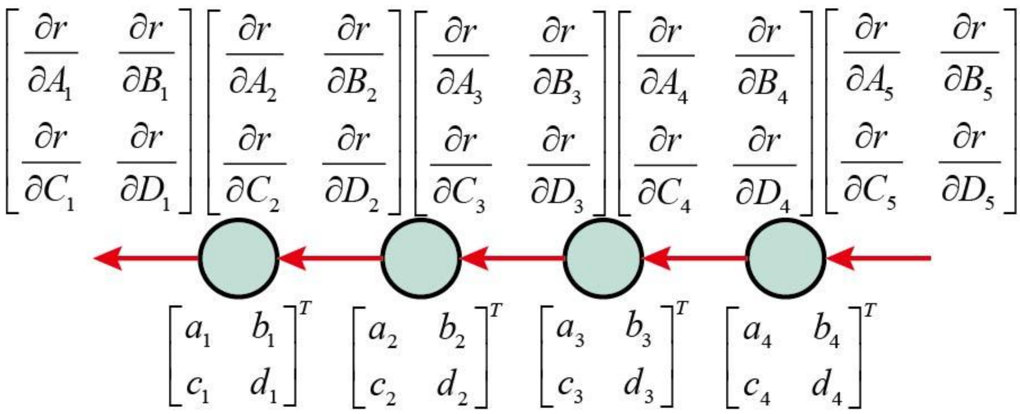

Fig. S3. The stem of the backpropagation process in TFNNs.

For the branches of the backpropagation process in TFNNs, the process starts from the stem in each layer and ended in a_i, b_i, c_i, and d_i as shown in Eq. (S4) and Fig. S4.

$$\begin{bmatrix} \frac{\partial r}{\partial a_i} & \frac{\partial r}{\partial b_i} \\ \frac{\partial r}{\partial c_i} & \frac{\partial r}{\partial d_i} \end{bmatrix} = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix}^T \begin{bmatrix} \frac{\partial r}{\partial A_i} & \frac{\partial r}{\partial B_i} \\ \frac{\partial r}{\partial C_i} & \frac{\partial r}{\partial D_i} \end{bmatrix} \qquad \text{(S4)}$$

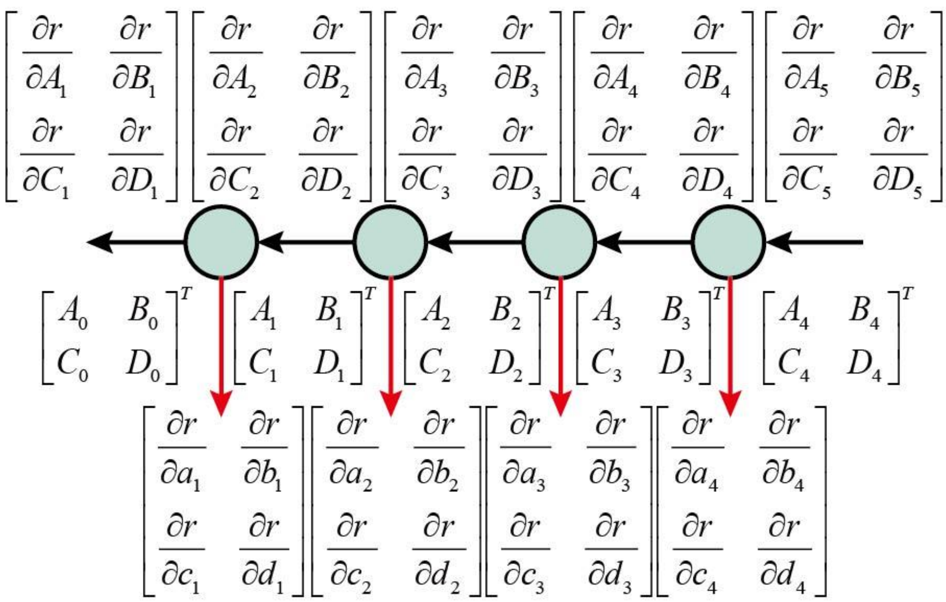

Fig. S4. The branches of the backpropagation process in TFNNs

The propagation of the gradients from the Fresnel coefficients r to the energy coefficients R is constructed from the complex number space to the real number space as shown in Fig. S5, The basic idea is to calculate the value of $\partial R/\partial r$. However, this derivative doesn't exist for its different value on each direction on the complex plane. Therefore a more robust connection between the complex number space to real number space is proposed as shown in Eq. (S5)

$$\frac{\partial R}{\partial A_i} = \frac{\partial |r|^2}{\partial A_i} = r\frac{\partial r^*}{\partial A_i} + r^*\frac{\partial r}{\partial A_i} = 2Re(r^*\frac{\partial r}{\partial A_i}) \qquad \text{(S5)}$$

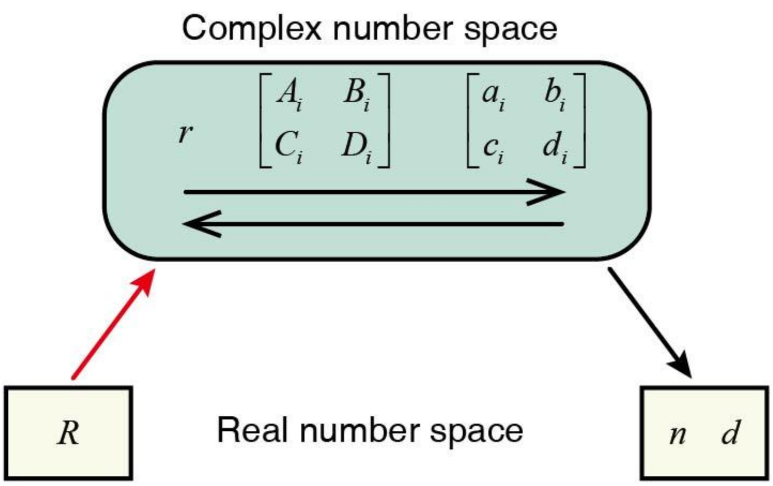

Fig. S5. Backpropagation from real number space to complex number space.

The Cauchy equations and Forouhi-Bloomer dispersion relations are used as the dispersion relations for the wavelength-dependent refractive indices in TFNNs. Cauchy equations are suited to model SiO_2 in monolayer thin films and TiO_2, Si_3N_4, and K9 glass in multilayer thin films:

$$n(\lambda) = a + \frac{b}{\lambda^2} + \frac{c}{\lambda^4} \qquad \text{(S6a)}$$

$$k(\lambda) = 0 \qquad \text{(S6b)}$$

where a_n, b_n, and c_n are the fitting parameters in Cauchy equations. Forouhi-Bloomer dispersion relations have been developed for modelling the complex index of refraction of Si in monolayer and multilayer thin films:

$$n(E) = n_{inf} + \sum_{i=1}^{q} \frac{B_{oi}E + C_{oi}}{E^2 - B_{fi}E + C_{fi}} \qquad \text{(S7a)}$$

$$k(\lambda) = \sum_{i=1}^{q} \frac{A_{fi}(E - E_g)^2}{E^2 - B_{fi}E + C_{fi}} \qquad \text{(S7b)}$$

with

$$B_{oi} = \frac{A_{fi}}{Q_{fi}}\left(-\frac{B_{fi}^2}{2} + E_g B_{fi} - E_g^2 + C_{fi}\right) \qquad \text{(S8a)}$$

$$C_{oi} = \frac{A_{fi}}{Q_{fi}}\left((E_g^2 + C_{fi})\frac{B_{fi}}{2} - 2E_g C_{fi}\right) \qquad \text{(S8b)}$$

$$Q_{fi} = \frac{1}{2}\left(4C_{fi} - B_{fi}^2\right)^{1/2} \qquad \text{(S8c)}$$

Through the backpropagation propagation process in TFNNs, the gradients $\partial R/\partial a$, $\partial R/\partial b$, $\partial R/\partial c$ $\cdots$ are obtained for optimization. Thus, the derivatives of the dispersion relations should also be given in the backpropagation process for calculating the gradients.

Comparison with gradient based differential method

If gradient based differential method is applied to the optical inverse problem of 232-layer thin films, the following process should be completed.

Step 1: Initial values of the thicknesses of 232-layer thin films is selected as the initial point for optimization. The spectrum of the thin films with initial thicknesses could be obtained by one TMM calculation.

Step 2: The thickness of i-th layer is changed a little bit, while the thicknesses of the rest layer remain same. The spectrum of the thin films with current thicknesses could be obtained by one TMM calculation.

Step 3: By comparing the spectra obtained from Step 1 and Step 2, the derivative of spectra with respect to the thickness of i-th layer is obtained.

Step 4: Repeat Step 2 and Step 3 232 times. We could know how to update the thickness of each layer in this iteration.

Therefore, at least 233 TMM calculations are needed in an iteration for solving optical inverse problem of 232-layer thin films (67.498 s per iteration). What make it much worse is that it often takes dozens or even hundreds of iterations before a reasonable design can be found. Although each TMM calculation of 232-layer thin films is fast, conventional methods such as TMM for optical inverse problem, where ten thousands of or millions of simulations are needed for complex structures such as 232-layer thin films, are still not fast enough. A practical error and time for solving the optical inverse problem of 232-layer thin films by using the differential method is shown in Fig. S6.

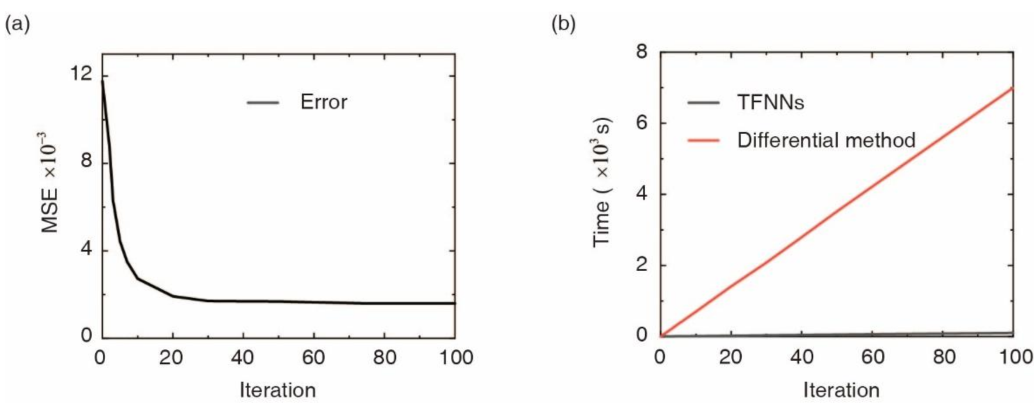

Fig. S6. **A practical error and time for solving the optical inverse problem of 232-layer thin films.** (a) The error between the target and the simulated spectra at each iteration. (b) The time needed for TFNNs (black, 102.13 seconds for 100 iterations) and differential method (red, 1.96 hours for 100 iterations).

Using ANNs for solving optical inverse problem of 232-layer thin films

We use ANNs to solve the inverse problem of 232-layer thin films. We train a 232-layer thin films model by using 1,000,000 examples of 232-layer thin films and constrain thickness of each layer in a relatively small range (80 nm to 120 nm, and 10 nm to 50 nm). It still took us $30517\ \mathrm{s} \approx 509.51\ \mathrm{min} \approx 8.49\ \mathrm{h}$ for obtaining 1,000,000 examples and another $7.44\ \mathrm{h}$ for training per 100 epochs. The results of ANNs show that the optical inverse problem of multilayer thin films with hundreds of layers is not an easy task.

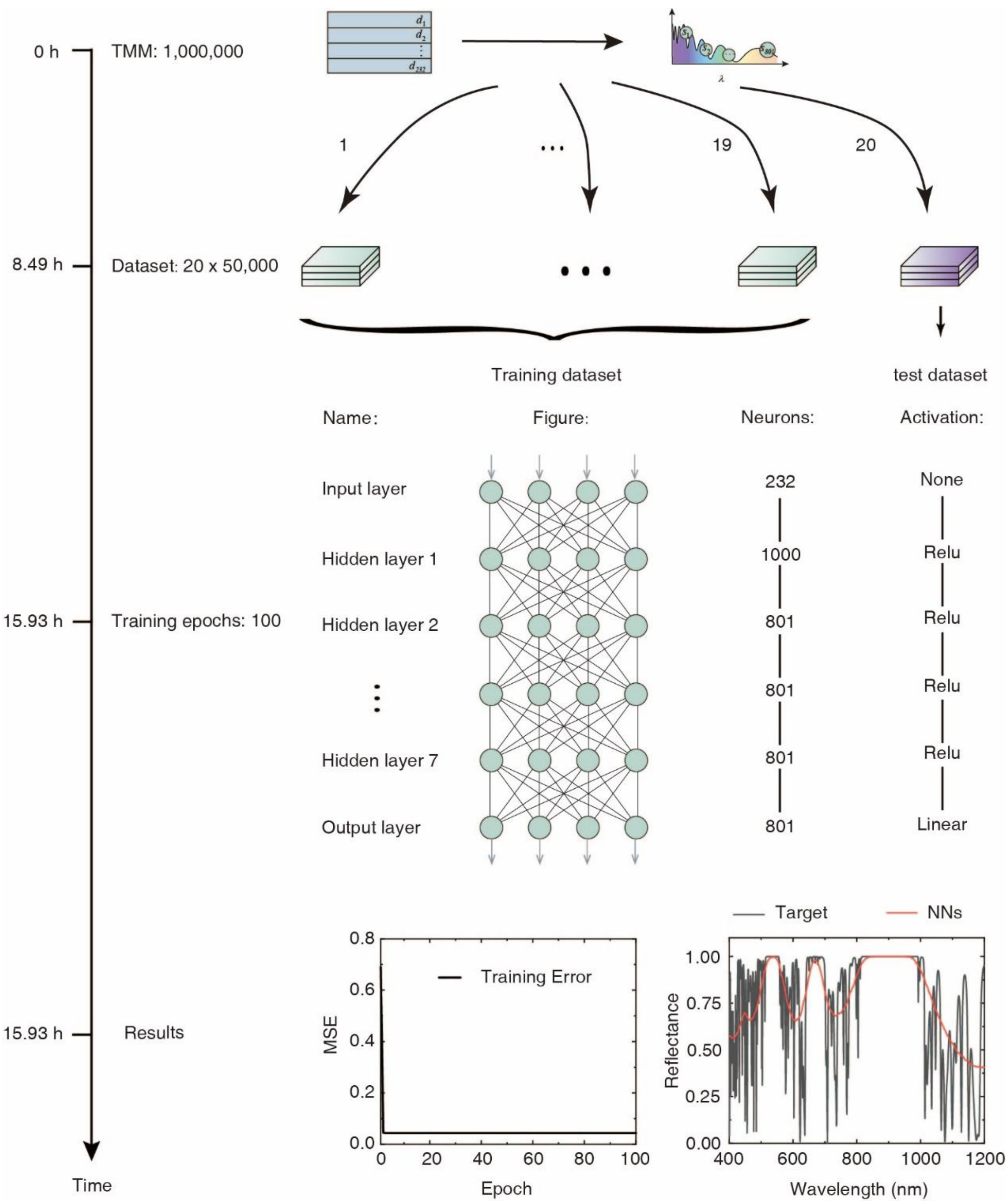

Fig. S7. Using ANNs to solve the inverse problem of 232-layer thin films with thickness between 80 nm and 120 nm.

As shown in Fig. S7, we divided 1,000,000 examples of 232-layer thin films into 20 parts. Each part contains 50,000 examples and is simulated on each core of a multicore server (Intel(R) Xeon(R) Gold 6230 CPU @2.10GHz 2.10GHz). However, it still took us $30517 \text{ s} \approx 509.51 \text{ min} \approx 8.49 \text{ h}$.

Then, we divided 1,000,000 examples into training dataset and test dataset. The training dataset includes 950,000 examples and test dataset includes 50,000 examples. The architecture of the neural networks is shown in Fig. S7, and its training results is also shown in Fig. S7. After 100 epochs of training (It took us 7.44 h on Tesla GPU (Tesla V100-PCIE-32GB, pci bus id: 0000:af:00.0, compute capability: 7.0)), the neural networks have only learned the band gaps of the multilayer thin films. the rest dense fringes are ignored as noises. The network could not reduce the training error further.

Since the dense fringes prevented ANNs learning the spectra of 232-layer thin films with thickness between 80 nm and 120 nm, we simulated another 1,000,000 examples of 232-layer thin films with thickness between 10 nm and 50 nm, and training the neural networks for 300 epochs (It took us 19.17 h on Tesla GPU (Tesla V100-PCIE-32GB, pci bus id: 0000:af:00.0, compute capability: 7.0)). The following results in Fig. S8 will show that even in this cases, recently proposed ANNs solution for optical inverse problem couldn’t handle it.

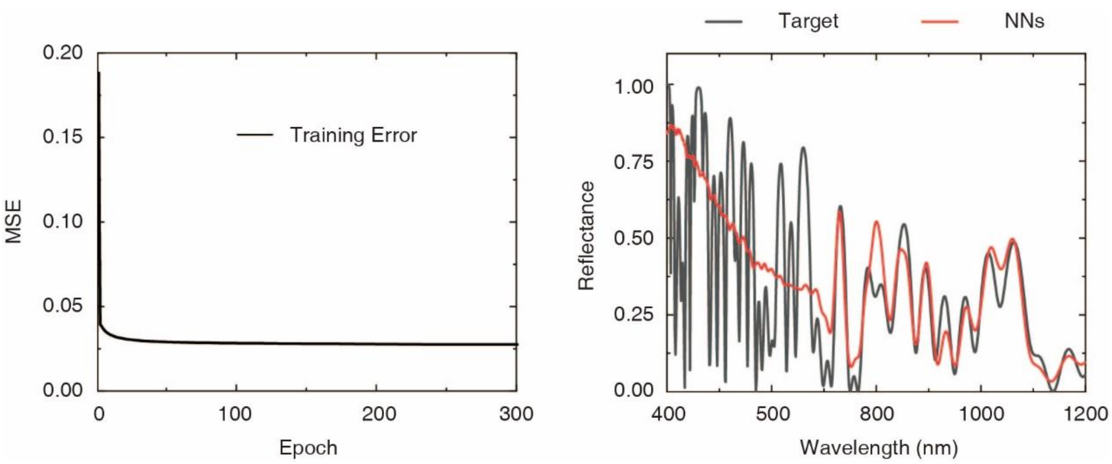

Fig. S8. **Whole architecture of the implementation of TFNNs. Using ANNs to solve the inverse problem of 232-layer thin films with thickness between 10 nm and 50 nm.** ANNs could learn the spectra in wavelength range between 800 nm and 1200 nm, while ignore dense fringes between 400 nm and 800 nm as noises.

Monolayer thin films with different thicknesses

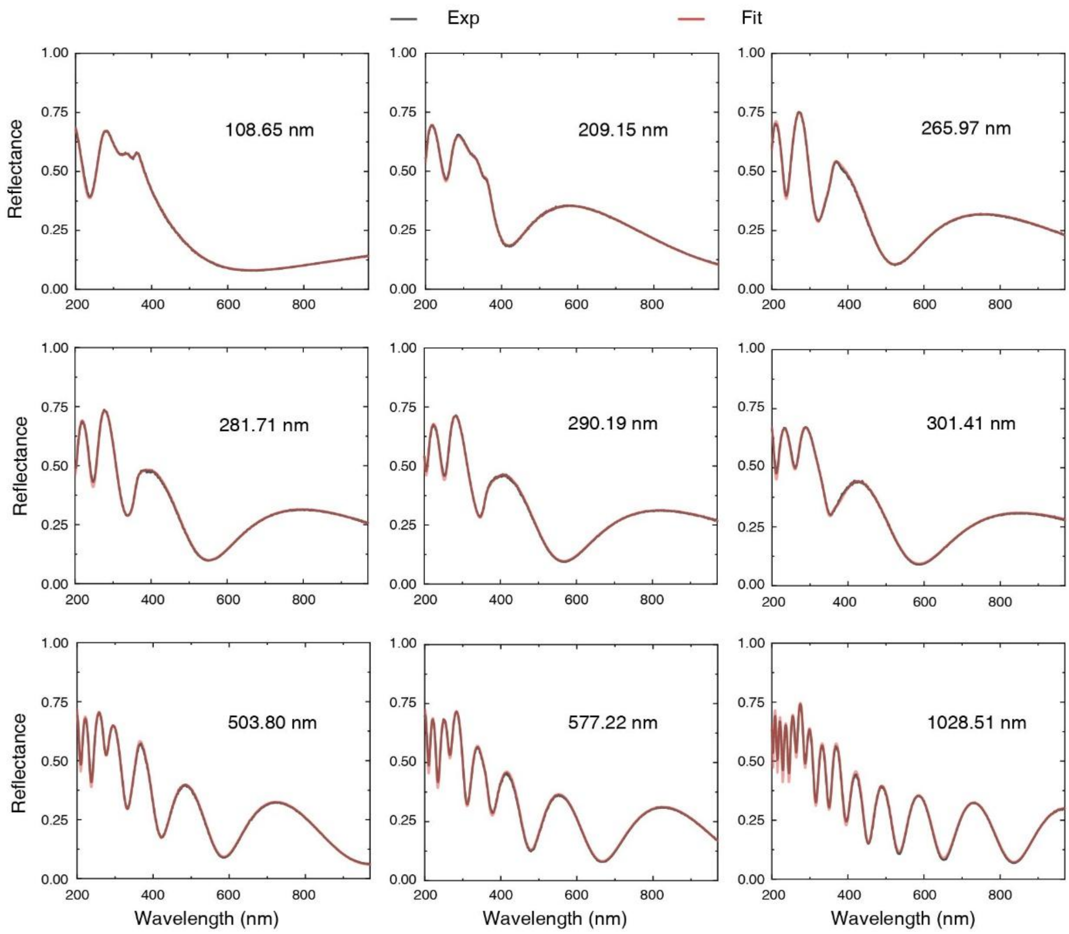

Fig. S9. Fitting results of monolayer thin films with different thicknesses.

Unnormal incidence cases

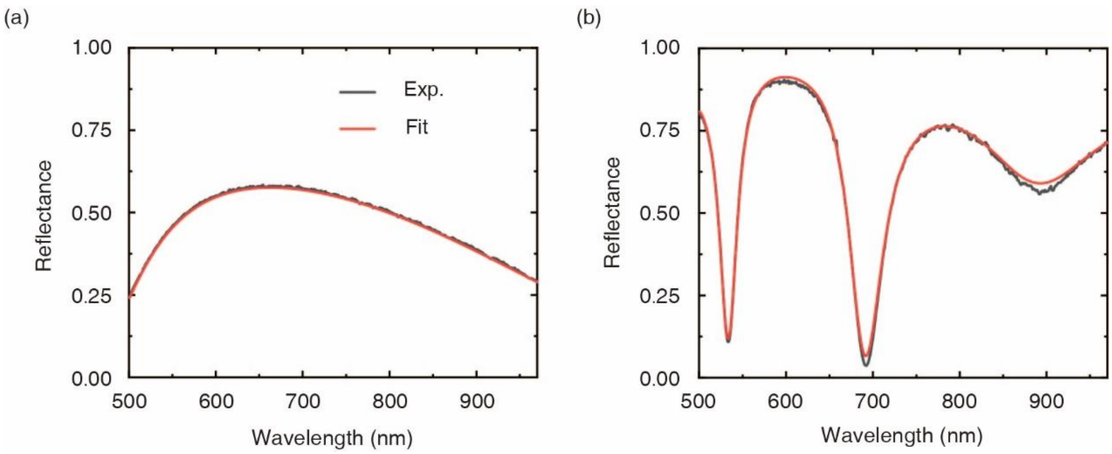

Fig. S10. **Unnormal incidence of rays for SiO2 and SOI wafer.** (a) Experimental and fitting result of SiO_2 thin films on Si with the incident beam inclining to 60 degrees. The fitting thickness of SiO_2 is 287.6 nm. (b) Experimental and fitting result of SOI wafer with the incident beam inclining to 60 degrees. The fitting thickness of top Si is 196.9 nm, and the fitting thickness of SiO2 is 376.1 nm.

Refractive index of the substrate

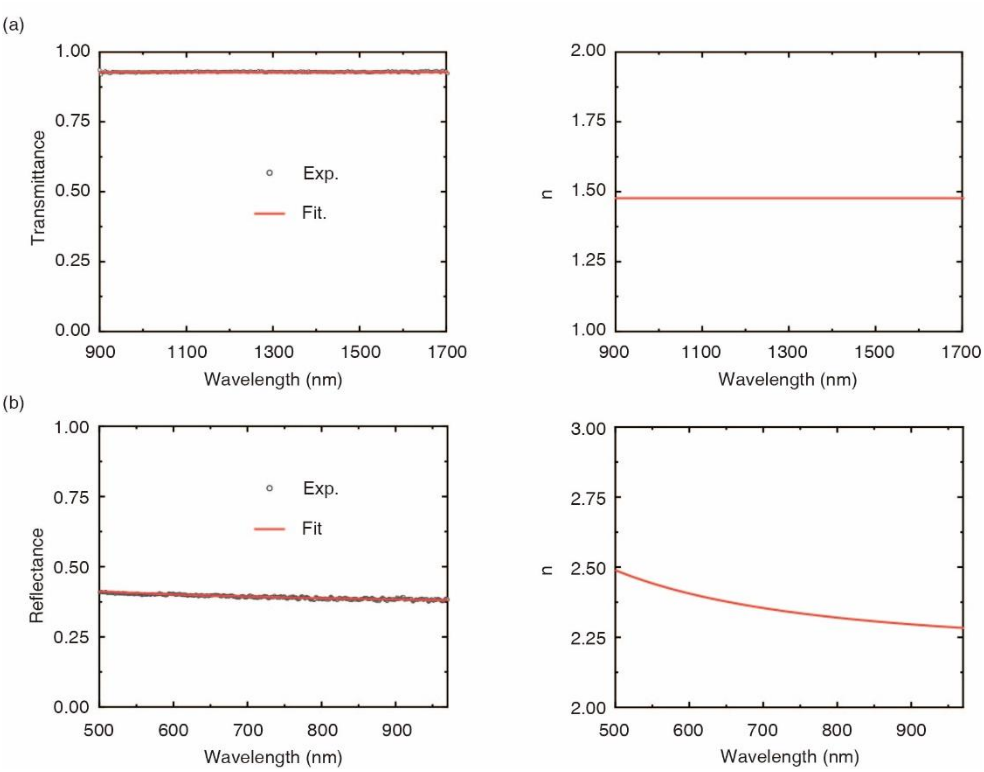

Fig. S11. **Obtaining the refractive index of the substrate.** (a) The refractive index of glass substrate obtained by measuring the transmittance at 0 degree. (b) The refractive index of STO substrate obtained by measuring the reflectance at 60 degrees.

Optimizing thicknesses and refractive indices of SOI wafer

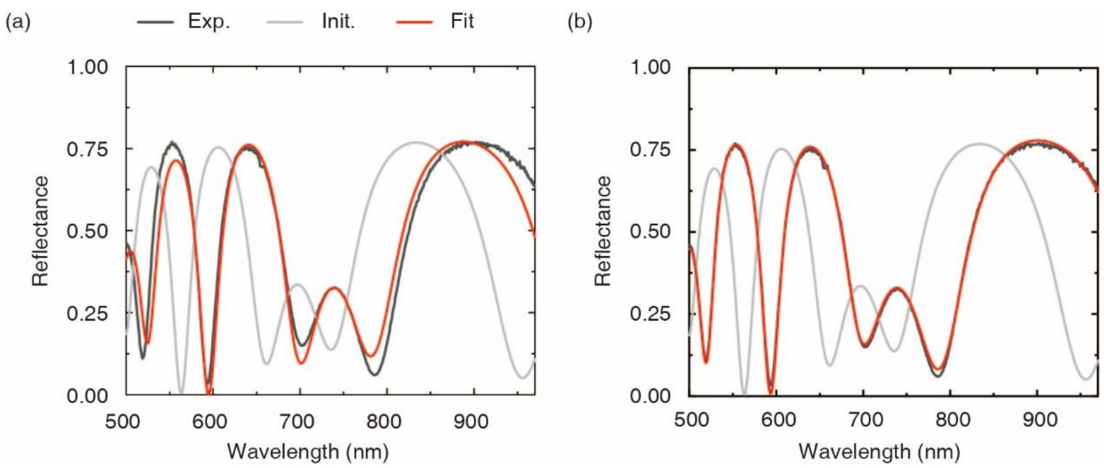

Fig. S12. **Optimizing thicknesses and refractive indices of SOI wafer.** (a) Only the thicknesses of SOI wafer are optimized. (b) Both the thicknesses and refractive indices of SOI wafer are optimized.

Optical inverse problem of 3D NAND

The detection of the erroneous layer in 3D NAND is presented based on simulation as a potential application of TFNNs. The multilayer structure of 3D NAND with 200 layers is shown in Fig. S13(a). Here, two samples of 3D NAND are discussed. One is a normal sample, and another is an outlier sample with an erroneous layer in it. Our main aim is to distinguish the outlier sample from the normal sample and identify the position of the erroneous layer in the outlier sample. For the normal sample, the thicknesses of SiO2 and Si3N4 layers are around 30 nm and 20 nm, respectively. For the outlier sample, the thickness of 40-th layer is intentionally set to be 5 nm thicker than the thickness of normal Si3N4 layers, while the rest layers remain unchanged. We add random noise on the thickness of each layer by considering the fluctuation of thickness in practical fabrication. The standard deviation of the random noise on thickness is 0.3 nm. After several iterations in the training of TFNNs, the predicted thicknesses and actual thicknesses of the normal and outlier 3D NAND are shown in Fig. S13(b). For the normal sample, the obtained thicknesses of all layers are restricted around 20 nm and 30 nm with the standard deviation of 0.3 nm. For the outlier sample, the obtained thickness of 40-th layer, marked as red circle as shown in Fig. S13(b), has a large deviation from other layers because of the large gradient of this layer in the training process of TFNNs, while the standard deviation of the rest layers is 0.3 nm. Therefore, TFNNs could successfully distinguish the outlier sample from the normal sample and detect the erroneous layer in 3D NAND.

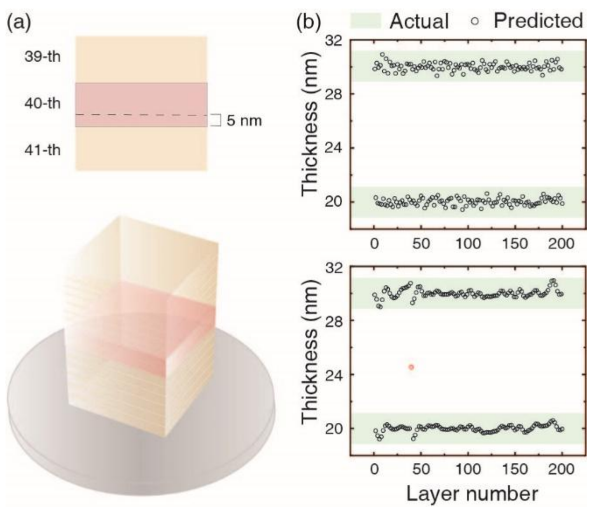

Fig. S13. **TFNNs for 3D NAND detection.** (a) Schematic view of the multilayer stacks of 3D NAND. (b) The actual thicknesses and predicted thicknesses by TFNNs of the normal sample (top) and outlier sample (bottom).

Extended to other nanophotonic structures

For other nanophotonic structures (taking photonic crystals as an example), we can also find such a structural similarity between photonic crystals and neural networks. Because of the structural similarity, we can build backpropagation process similar to neural networks in photonic crystals.

At the interface of photonic crystals, Eq. (1a) and Eq. (1b) in our manuscript, the structural similarity between the weight connection of neural networks and the interface of nanophotonic structures, still hold true. The differences in photonic crystals are: a_{i-1} and b_{i-1} of photonic crystals are vectors with N elements, where N is the number of the reciprocal lattice vector used in calculation, while a_{i-1} and b_{i-1} of thin films are numbers. The interface matrix $\boldsymbol{T_i}$ of photonics crystals is $\boldsymbol{T_i} = \boldsymbol{M_{i-1}^{-1}M_i}$. The details expression and the meaning of matrix $\boldsymbol{M_i}$ refer to Eq. (4.6) and Eq. (4.7) in [***D.M. Whittaker, I.S. Culshaw, Scattering-matrix treatment of patterned multilayer photonic structures, Phys. Rev. B 60 (1999) 2610-2618***]. While the interface matrix of thin films is $\boldsymbol{T_i = D_{i-1}^{-1}D_i}$. The details expression and the meaning of matrix $\boldsymbol{D_i}$ refer to Eq. (1) in [***Katsidis, C. C. & Siapkas, D. I. General transfer-matrix method for optical multilayer systems with coherent, partially coherent, and incoherent interference. Applied Optics 41, 3978–87 (2002)***].

At the bulk of photonic crystals, Eq. (2a) and Eq. (2b) in our manuscript, the structural similarity between the neurons of neural networks and the bulk of nanophotonic structures, still hold true. The differences in photonic crystals are: The propagation matrix $\boldsymbol{P_i}$ of photonic crystal is

$$\boldsymbol{P_i} = \begin{bmatrix} \hat{\mathbf{f}}(\mathbf{z}) & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{f}}(-\mathbf{z}) \end{bmatrix} \qquad \text{(S9)}$$

where the detail expression and the meaning of $\hat{\mathbf{f}}(\mathbf{z})$ and $\hat{\mathbf{f}}(-\mathbf{z})$, diagonal matrices, refer to Eq. (4.1) and Eq. (4.2) in [***D.M. Whittaker, I.S. Culshaw, Scattering-matrix treatment of patterned multilayer photonic structures, Phys. Rev. B 60 (1999) 2610-2618***]. While the propagation matrix $\boldsymbol{P_i}$ of thin films is

$$\boldsymbol{P_i} = \begin{bmatrix} e^{j\varphi_i} & 0 \\ 0 & e^{-j\varphi_i} \end{bmatrix} \qquad \text{(S10)}$$

Through the above comparison, the backpropagation process could also be established in photonic crystals. And the method in our manuscript could be extended to other nanophotonic structures. What's more, a_{i-1} and b_{i-1} of photonic crystals are vectors with N elements, which means that the number of the neurons in photonic crystals eventually depends on the number of the optical modes propagating in photonic crystals

Reuse property

Here, we present the example of one training be used for multiple thin film structures. We first establish a multilayer thin film model with 10 layers, and the optimization result is shown in Fig. S14 (blue line). Then, we add 10 layers on previous model, and it could be directly used for the inverse design task for thin films with 20 layers, and better optimization result is obtained shown in Fig. S14 (green line). And there is no need for training a new model for the 20-layer inverse design task.

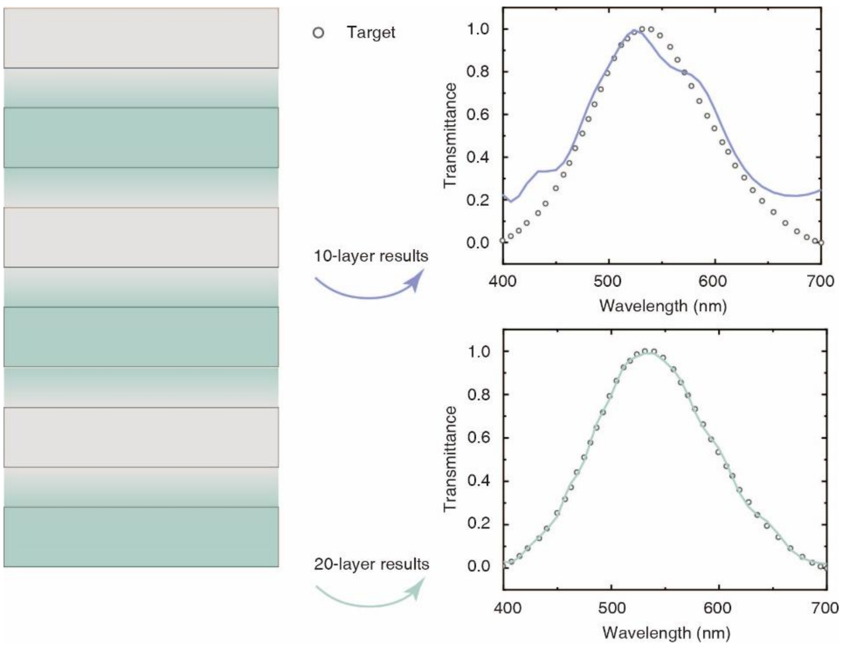

Fig. S14. **One training for multiple thin film structures.** the results of 20-layer thin films could be obtained by adding 10 layers on previous trained model of 10-layer thin films.

Further reducing difference/errors

We will discuss how to further reduce the difference/errors of the two aspects of optical inverse problems, optical metrology and optical inverse design.

For optical metrology task of monolayer thin films, since the initial structural parameter is near the global minimum point, the difference/errors mainly come from the noise during experimental measurement. We can reduce errors through some experimental methods (e.g. averaging the values of repeated measurements.)

For inverse design task and metrology task of multilayer thin films, it's easy to get stuck in the local minimum. An effective solution is to select multiple initial values in the global range, and then choose the result with the smallest error among these results. We conduct relevant analysis for 60-layer thin films. 200 initial points are selected and the following results are 10 cases among them, shown in **Table** S1. Therefore, Case 6 with the smallest errors among 200 initial points are chosen. By using this method, we could avoid plunging into the local minimum, and further reduce the difference/errors.

If the number of layers or free parameters of design tasks is not limited, the method in our manuscript can get smaller errors by adding more layers or free parameters into the design tasks, as shown in Fig. S13.

Table S1. Multiple initial values for 60-layer thin films.

Case	1	2	3	4	5	6	7	8	9	10
MSE ($\times 10^{-3}$)	1.38 18	1.39 56	1.21 76	1.59 68	1.54 24	1.19 71	1.20 38	1.43 11	1.31 57	1.43 75

Comparing with other neural network

Here, we list the advantages and disadvantages of other neural network methods, as well as the advantages and disadvantages of the methods in our manuscript.

Advantages of other neural network methods:

1. It’s easy to be applied to optical inverse design of other structures. By using the spectra of different structures to train the neural network model, the neural network methods can be easily applied to other nanophotonic structures.

2. Less simulation time. The time needed for the well-trained neural network model to complete a calculation from structure to spectrum is far less than that for a conventional electromagnetic simulation.

3. Analytical gradients. The analytic gradients can be obtained by using the backpropagation of neural network.

Disadvantages of other neural network methods:

1. Training neural network model requires large dataset, especially for complex inverse design problem (e.g. multilayer thin films with 232 layers)

2. Low accuracy. There is between the output results of the training model and the electromagnetic simulation results.

3. Some cases are difficult to train. Part of the training tasks for optical inverse problem has been proved difficult, which needs to be solved by reasonably design neural network model.

4. The neural network model trained for the inverse design task of thin films with 232 layers cannot be applied to the reverse design task of thin films with 231 layers. A new neural network model needs to be trained for the thin films with 231 layers.

Advantages of the method in our manuscript:

1. Without dataset for training. The backpropagation process is directly established based on the transfer matrix, and the thin films could be directly regarded as a neural network without a large number of datasets to train another neural network to approximate Maxwell‘s Equation.

2. The spectra calculated by this method are accurate, and the analytical gradient can also be obtained by back propagation.

3. One training model could be used for multiple thin film structures. Based on the neural network model for thin films with 232 layers, one layer can be reduced to make it suitable for thin films with 231 layers.

Disadvantages of the method in our manuscript:

1. Compared with other neural networks, it takes more time to complete one iteration, but it's still faster than differential methods and evolutionary algorithms.

2. If we need to extend this method to other nanophotonic structures, we need to exploit the structural similarity between nanophotonic structures and neural networks, and construct the backpropagation process in other nanophotonic structures.

Software architecture

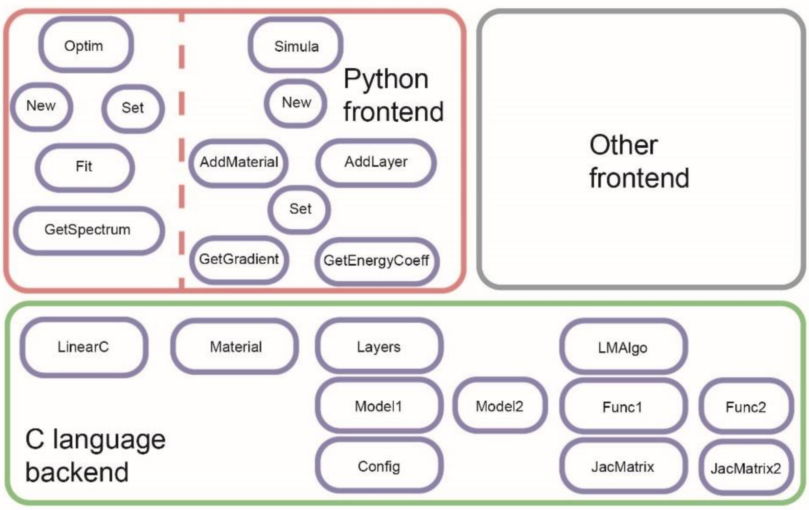

Fig. S15. Whole architecture of the implementation of TFNNs.