Electronic Supplementary Material (ESI) for Digital Discovery.

Supplementary Information

S1. *thicknessML* Hyperparameters

Table S1. *thicknessML* hyperparameters

thicknessML Hyperparameters	Setting
Convolutional layers	4
Filter size	[8, 5, 3, 3]
No. of filters	[512, 128, 64, 32]
Max pooling layers	4
Spatial extent of pooling layers	[3,3,2,2]
Number of units for fully connected layers for *d*	[2048, 1024, 512, 1]
Number of units for fully connected layers for *n*	[2048, 1024, 651]
Number of units for fully connected layers for *k*	[2048, 1024, 651]
Dropout rate of fully connected layers	0.3
Epochs	2000
Batch size	128
Optimization algorithm	AdaGrad
Learning rate	0.001 (initial and restored every 50 epochs since 150)
The *d* loss scaler w_d	0.01
The *n* loss scaler w_n	10
The *k* loss scaler w_k	8

S2. Visualization of *thicknessML* Activation Maps of an Example *R*, *T* Spectra

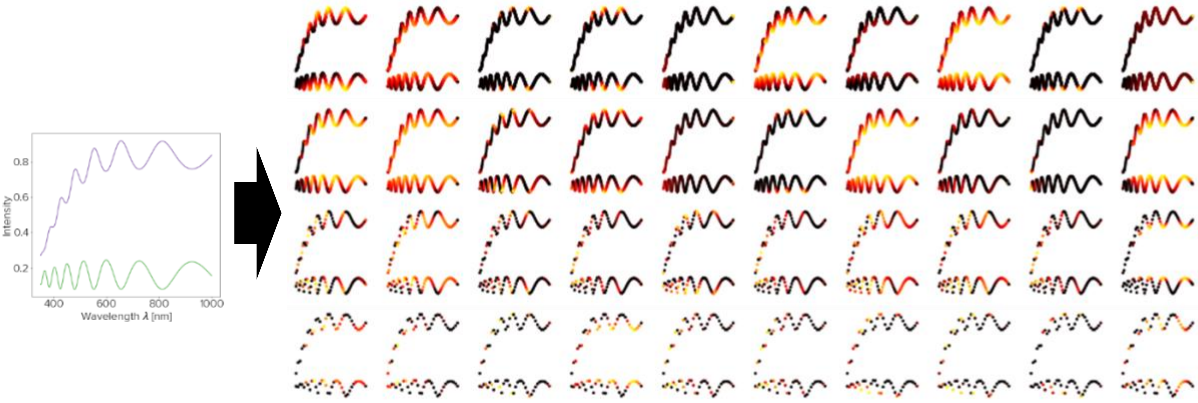


Figure S1. Visualization of activation maps of ten random filters of the 4 convolutional layers (top to bottom). First to fourth convolutional layer) from a pre-trained thicknessML. Lighter color indicates higher activation of the spectra.

Figure S1 plots the activation maps of an example *R*, *T* spectra. An activation map is the output of a given filter (weights of convolutional layers) applied to the previous layer. Activation maps give some intuition into which part of the spectra each filter focuses on (gets more activated). For instance, we can see some filters focus on extracting the hill tops and valley bottoms of the spectra, some filters get activated when encountering certain gradients like uphill or downhills portions, and some filters recognize more holistically where the whole hills and valleys are. This tallies well with the physical picture, where positions of the hills and valleys are the most prominent feature in the *R*, *T* spectra when thickness *d* is varied. More oscillations (hills and valleys) are being seen with a larger *d*.

S3. Selection of 1,116 *n*, *k* Spectra in the Generic Source Dataset

In our preliminary running by randomly sampling from all the simulated *n*, *k* spectra (to pair with randomly selected thicknesses), we find that the prediction of *thicknessML* is worse for simulated *n*, *k* spectra with more noticeable “bumps”. We thus perform a K-means clustering on all the simulated *n*, *k* spectra from the parameter grids: A, 10 to 200 with 11 grid nodes; C, 0.5 to 10 with 10 grid nodes; E_0, 1 to 10 with 10 grid nodes, and E_g, 1 to 5 with 10 grid nodes. The K-means (with K=3) centers are shown in Figure S2. (The *n*, *k* spectra are normalized to between 0 and 1.) We notice that simulated *n*, *k* spectra with noticeable “bumps” (first cluster center) consist of a noticeably smaller population than the other two. Thus, in our final generic source dataset, we take all 372 *n*, *k* spectra in the first cluster and the corresponding numbers randomly from the other two clusters, yielding 372×3=1116 spectra.

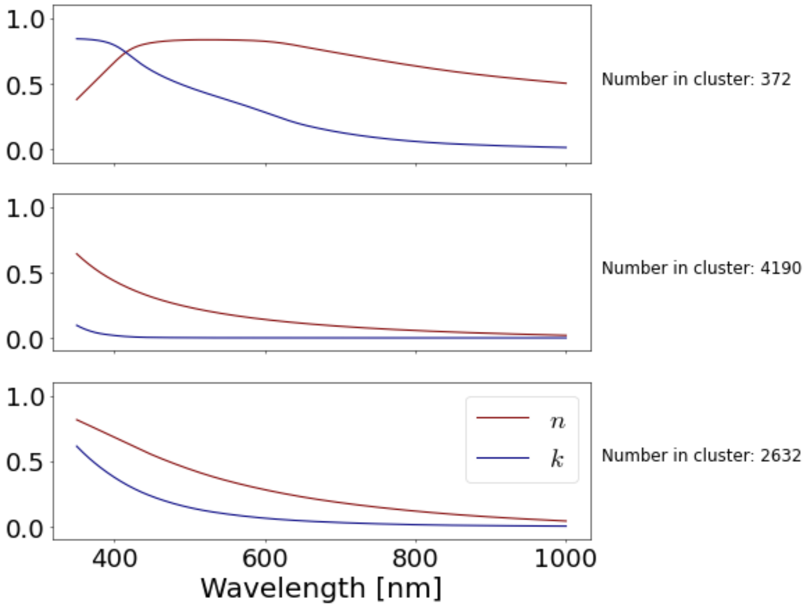
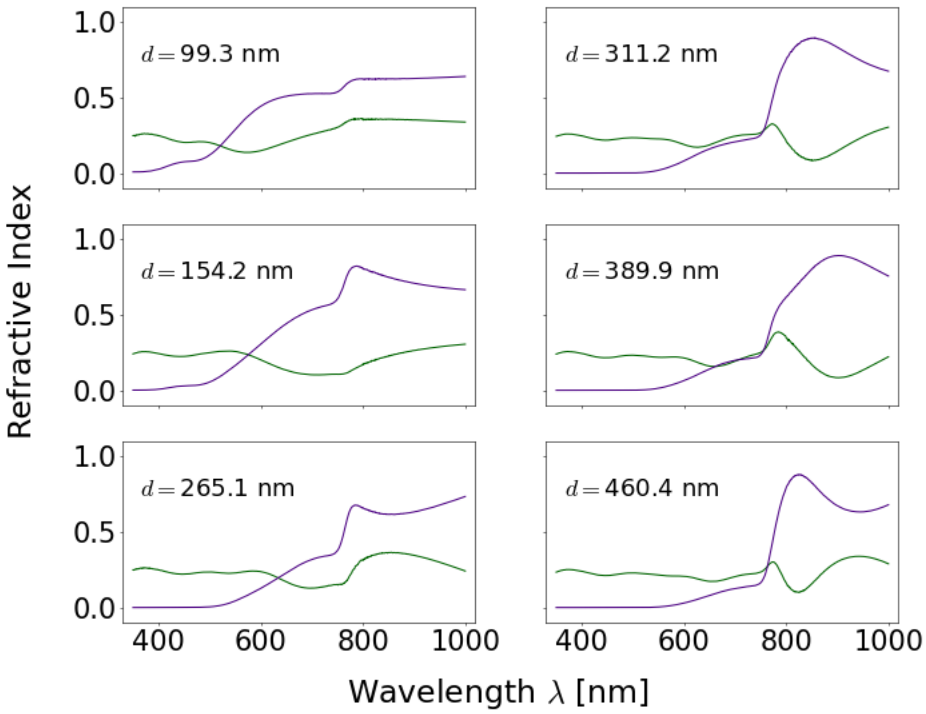


Figure S2. K-means (K=3) centers in clustering all the simulated *n*, *k* spectra from the parameter grids: A, 10 to 200 with 11 grid nodes; C, 0.5 to 10 with 10 grid nodes; E_0, 1 to 10 with 10 grid nodes, and E_g, 1 to 5 with 10 grid nodes. The *n*, *k* spectra are normalized to between 0 and 1.

S4. Raw Measurements of the Experimentally Deposited $MAPbI_3$ Films



1	99.29 nm
2	154.17 nm
3	265.07 nm
4	311.15 nm
5	389.89 nm
6	460.35 nm

Figure S3. Raw measurements of the experimentally deposited $MAPbI_3$ films: UV-Vis measurements shown on the left and profilometry measurements (averaged value directly obtained from the profiler) shown on the right.

The raw UV-Vis and thickness measurements are directly accessible from Expdataset.h5 either in the data repository https://doi.org/10.6084/m9.figshare.23501715.v1 or in the data folder of the code repository https://github.com/PV-Lab/thicknessML.

S5. Distribution of *d* in the Generic Source Dataset (Training Set)

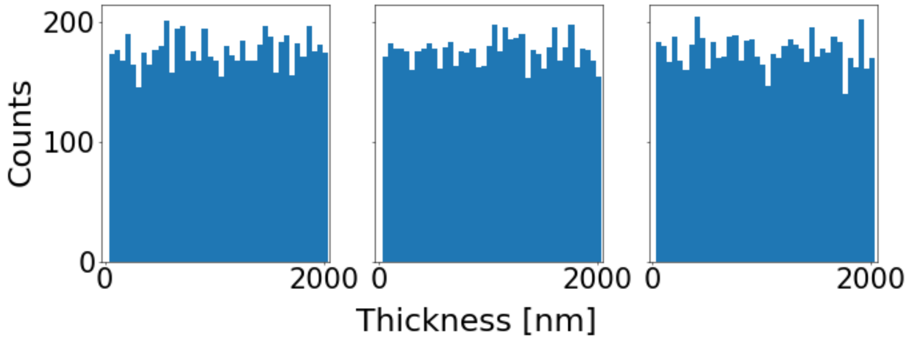


Figure S4. The distribution of randomly selected thicknesses *d* in the generic source dataset (training set). Three distributions correspond to three training-validation-test splits (for ensemble training).