Identification of spatial confinement in nanostructures using machine learning methods

Igor Boyko[*,†] and Marharyta Prachuk[†]

Ternopil Ivan Puluj National Technical University, Ruska Street 56, 46001 Ternopil, Ukraine

Abstract

Low-dimensional semiconductor systems used in modern nanodevices for generating and detecting electromagnetic radiation require an accurate and efficient approach to determining their geometric and functional parameters. This paper proposes a methodology that allows the analysis of large datasets obtained from experimental and theoretical studies of nanostructures in order to identify precise parameters of their spatial confinement. Automation of data processing and nanosystem parameter identification is implemented using machine learning methods and a convolutional neural network, enabling efficient and reliable characterization of nanosystem parameters of arbitrary symmetry. The developed software tool will significantly streamline the work of specialists in nanotechnology and nanostructured material synthesis.

Keywords

Nanostructures class, spatial confinement, machine learning, convolutional neural network, nanotechnology.

1. Introduction

Devices for detecting [1] and generating electromagnetic radiation [2] of different frequencies and intensities play a key role in the development of modern optical fiber electronics, computer technologies, and military equipment. The functional elements of such nanodevices are multilayer resonant tunneling nanostructures [3]. The development, research, and fabrication of these nanosystems require significant investment in automation, as well as the use of specialized software at various stages of fabrication and modeling.

A key problem in working with such nanostructures is the lack of a clear correlation between the results of the initial modeling of the parameters of these nanosystems and the results obtained as a result of their direct growth [4]. Typically, nanostructures whose properties were previously modeled on the basis of the same model, immediately after their growth, have diverse geometric confinement, deviating from the desired parameters, which affects their immediate spectral characteristics. Since nanodevices such as quantum cascade lasers or detectors [1, 2, 5–8] use hundreds of equivalent cascades, it is practically impossible to directly determine in practice what the deviations of the confinement of real nanostructures from their modeled calculated parameters are.

The experimental data obtained from such nanodevices are datasets containing tunneling current or electron conductivity values. These data can be interpreted directly using specialized software. In direct measurements, deviations of the geometric confinement of nanostructures from their model values are manifested in the ambiguity of determining the values of current or conductivity and their oscillations, which are determined by the spectral characteristics of electrons in such low-dimensional systems. Due to the multilayer nature of real resonant tunneling structures used in devices, experimenters face a significant problem, as it is impossible to unambiguously establish which geometric design of nanostructures corresponds to the obtained

[*] *AIT&AIS'2025: International Scientific Workshop on Applied Information Technologies and Artificial Intelligence Systems, December 18–19 2025, Chernivtsi, Ukraine*

[1,*] Corresponding author.

[†] These authors contributed equally.

boyko.i.v.theory@gmail.com (I. Boyko); margo1702prachuk@gmail.com (M. Prachuk)

0000-0003-2787-1845 (I. Boyko); 0009-0009-8099-9865 (M. Prachuk)

deviations of measured quantities from their model values. The main approaches to determining the geometric parameters of nanostructures from experimentally measured current or conductivity values are inverse problems of quantum scattering theory. However, although many different numerical methods of this kind exist even for single quantum wells, they usually do not yield unambiguous results [9]. It is clear that the mentioned approaches cannot be applied even in the case of complex low-change structures, which are the subject of modern research.

The relevance of this problem lies in the fact that an approach that allows one to establish the geometric confinement of nanostructures based on experimental measurements would allow one to significantly optimize the operation of existing nanodevices. In this paper, we aim to solve this problem by applying machine learning methods to the analysis of large data arrays obtained from experimental studies and measurements of typical values of electron current and conductivity in nanosystems. Using these data and generalizing them with a developed mathematical model describing the electronic states in nanostructures, a convolutional neural network will be trained. When working with new experimental data, the developed neural network will unambiguously and reliably determine the geometric parameters, thus solving the general problem.

2. Related work

Currently, there is relatively little specialized software designed for modeling the properties of nanostructures. However, the following software systems, which can be classified by their level of abstraction, should be highlighted. In particular, software systems that use quantum-mechanical methods to describe interactions at the atomic level are worth mentioning: DFT, VASP, and Quantum ESPRESSO [10]. Another group includes software systems that employ density functional theory or molecular dynamics methods: NanoTCAD ViDES and NEMO [11]. Yet another group of software systems is based on mesoscopic methods: they use simplified models that allow for qualitative modeling of the electronic, optical, or transport properties of nanostructures, e.g., nextnano [12]. There are also systems based on continuum approaches to the Poisson equation and the Boltzmann kinetic equations. Among numerical methods, the most commonly used are the finite difference method, the finite element method, and the Monte Carlo method. Fig. 1 shows an example of modeling the properties of a nanostructure using the nextnano software system. Fig. 1 shows an example of modeling the properties of a nanostructure using the nextnano software system.

Using the aforementioned software, and as further illustrated in Fig. 1, it is possible to model the fundamental properties of nanostructures with a small number of layers. This allows a qualitative assessment of the spectral parameters of electrons in nanostructures and the calculation of the potential diagram of such a nanosystem. However, it should be concluded that none of the indicated software systems even approximately possesses the functionality required to solve inverse problems of identifying the geometric confinement of nanostructures, despite their fairly rich functionality for manipulating the input parameters of nanostructures and the materials from which they are formed. In light of the existing challenges associated with this topic, two relevant studies [2, 13] are noteworthy in which machine learning methods were directly applied to the development of nanostructures and the modeling of their properties. This allowed us to improve the input parameters of the nanostructures and optimize the performance of the final nanodevices, although these studies did not resolve the underlying issues. Nevertheless, they demonstrate progress in the application of artificial intelligence tools in this subject area. Therefore, given the objectives set in this paper, we will continue to develop this direction, using these studies as a bridge for further research.

3. Methods

We are beginning to address the stated problem of establishing a general approach, according to which we are working with input data networks that characterize the electronic spectrum E_n, n =

1, 2, .. in a nanostructure. We cannot directly apply datasets obtained from experimental spectral measurements, as such datasets do not directly correspond to the geometric structure x_i, $i = L/M$ (L – is the characteristic size of the nanostructure) of nanostructures. Therefore, we link spectral datasets to basic mathematical models that most accurately describe the spectral characteristics of electrons in such systems.

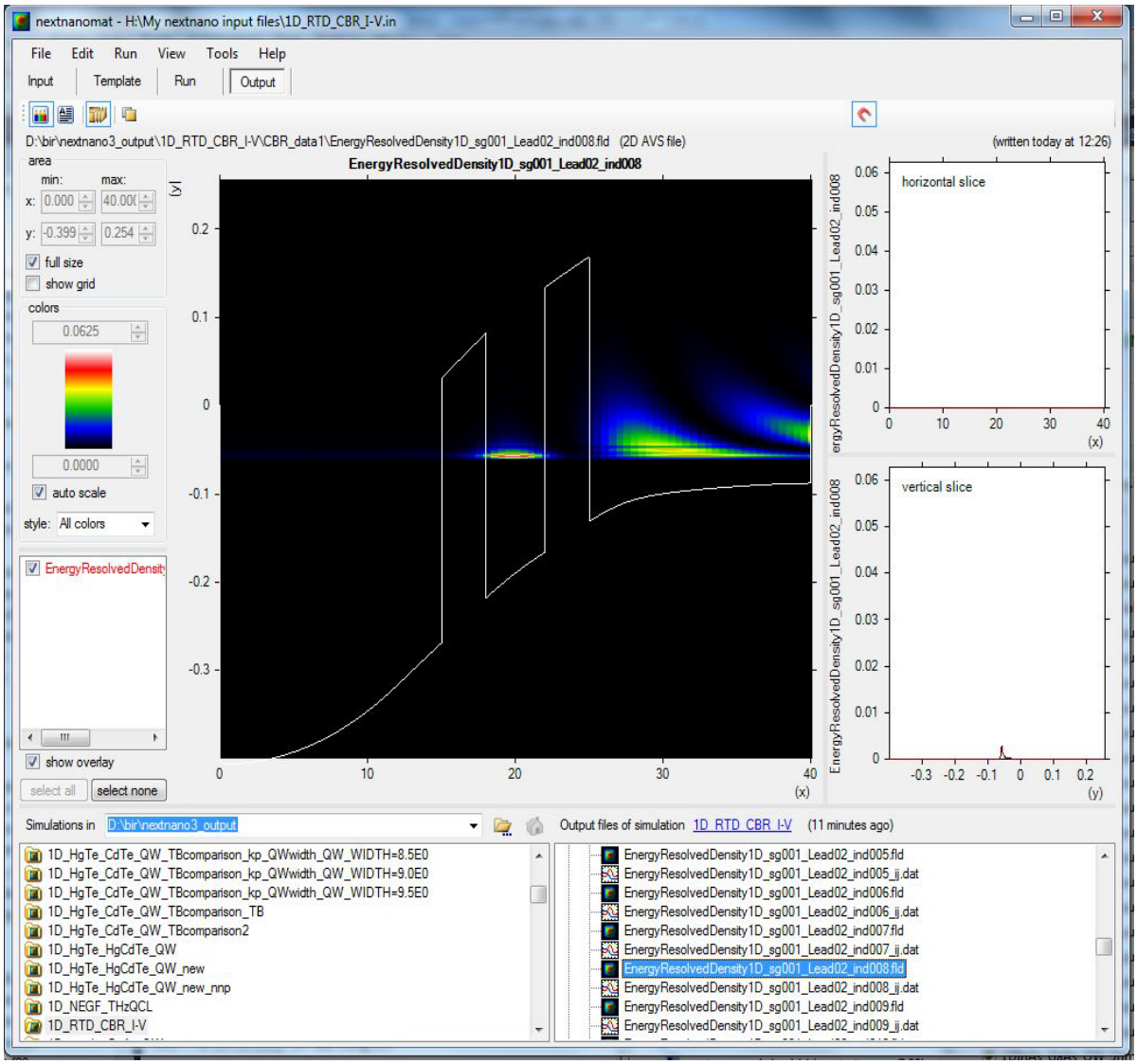

Figure 1: Modeling nanostructure properties using the nextnano software system (source: https://www.nextnano.com/ products/nextnanomat.php).

This basic mathematical model is based on the application of the stationary Schrödinger equation and the conditions of continuity of the wave function and probability flows for a nanosystem with any number of layers. This mathematical model can be represented as follows:

$$\left[-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}+U(x)-E\right]\Psi(x)=0;\quad \begin{cases} \Psi^{(p)}(x)\big|_{x=x_p}=\Psi^{(p+1)}(x)\big|_{x=x_p}, \\ \dfrac{\partial\Psi^{(p)}(x)}{\partial x}\bigg|_{x=x_p}=\dfrac{\partial\Psi^{(p+1)}(x)}{\partial x}\bigg|_{x=x_p}, \end{cases} \tag{1}$$

where $U(x)$ is the geometric confinement to be identified, E is the electronic spectrum, m is the effective mass of the electron in the given structure.

Next, we present the Schrödinger equations and boundary conditions in discretized form on a one-dimensional uniform grid:

$$-\alpha\left(\Psi_{i+1}-2\Psi_i+\Psi_{i-1}\right)+U_i\Psi_i-E\Psi_i=0,\quad \alpha=\frac{\hbar^2}{2mh^2},\quad i=1,\ldots,N-1;$$

$$\begin{cases}\Psi_0=\Psi_a,\Psi_N=\Psi_b,\\ \dfrac{\Psi_1-\Psi_0}{h}=g_0,\dfrac{\Psi_N-\Psi_{N-1}}{h}=g_N,\\ \Psi_0=\Psi_N,\Psi_{-1}=\Psi_{N-1}.\end{cases} \tag{2}$$

As a result, the spectral problem is reduced to a matrix equation describing the relationship between the energy spectrum and the geometric confinement of the nanostructure on a discrete grid:

$$H\Psi=E\Psi,\quad \Psi=\left(\Psi_1,\ldots,\Psi_{N-1}\right)^T,$$

$$H=\begin{pmatrix}2\alpha+U_1 & -\alpha & 0 & \vdots & 0\\ -\alpha & 2\alpha+U_2 & -\alpha & \ddots & \ldots\\ 0 & -\alpha & \ddots & \ddots & 0\\ \ldots & \ddots & \ddots & 2\alpha+U_{N-2} & -\alpha\\ 0 & \vdots & 0 & -\alpha & 2\alpha+U_{N-1}\end{pmatrix}. \tag{3}$$

We will use the resulting table dependencies $xi = xi(En)$ as input datasets for training the neural network. Another mathematical model we will use takes into account the kinetics of processes in nanostructures. It is based on the full Schrödinger equation and its limiting conditions. This mathematical model is represented as follows:

$$i\hbar\frac{\partial\Psi(x,t)}{\partial t}=\left[-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}+U(z)\right]\Psi(x,t);$$

$$\begin{cases}\Psi^{(p)}(x,t)\big|_{x=x_p}=\Psi^{(p+1)}(x,t)\big|_{x=x_p},\\ \dfrac{\partial\Psi^{(p)}(x,t)}{\partial x}\bigg|_{x=x_p}=\dfrac{\partial\Psi^{(p+1)}(x,t)}{\partial x}\bigg|_{x=x_p}.\end{cases} \tag{4}$$

Approximating this mathematical model on a two-dimensional discrete variable grid (x, t), we will have the following Crank-Nicholson type schemes for a multilayer system:

$$i\hbar\frac{\Psi_i^{n+1}-\Psi_i^n}{\Delta t}=-\frac{\alpha}{2}\left[\left(\Psi_{i+1}^{n+1}-2\Psi_i^{n+1}+\Psi_{i-1}^n\right)+\left(\Psi_{i+1}^n-2\Psi_i^n+\Psi_{i-1}^{n+1}\right)\right]+U_i\frac{\Psi_i^{n+1}+\Psi_i^n}{2},$$

$$\alpha=\frac{\hbar^2}{2mh^2},\ i=1,\ldots,N-1,\ n\geq 0;$$

$$\begin{cases}\Psi_0^n=\Psi_a^n,\Psi_N^n=\Psi_b^n,\\ \dfrac{\Psi_1^n-\Psi_0^n}{h}=g_0^n,\dfrac{\Psi_N^n-\Psi_{N-1}^n}{h}=g_N^n,\\ \Psi_0^n=\Psi_N^n,\Psi_{-1}^n=\Psi_{N-1}^n.\end{cases} \tag{5}$$

(a)	Xn(X)	EnX(Y)	EnY(Y)	EnX1(Y)
2	0,4	23,61591	50,65779	95,18734
3	0,8	23,56073	51,30238	97,38847
4	1,2	23,5194	52,15331	99,71174
5	1,6	23,49279	53,06278	102,16471
6	2	23,48189	54,03461	104,75558
7	2,4	23,48774	55,07295	107,49321
8	2,8	23,5115	56,18233	110,38702
9	3,2	23,55442	57,36767	113,44698
10	3,6	23,61786	58,63432	116,68326
11	4	23,7033	59,9881	120,10531
12	4,4	23,81238	61,43537	123,71848
13	4,8	23,94688	62,98306	127,50672
14	5,2	24,10874	64,63873	131,2242
15	5,6	24,30012	66,41063	128,02879
16	6	24,52335	68,30772	119,52186
17	6,4	24,78105	70,33969	111,86323
18	6,8	25,07607	72,51686	105,08682
19	7,1	25,3237	74,2513	100,514
20	7,2	25,41159	74,84961	99,07745

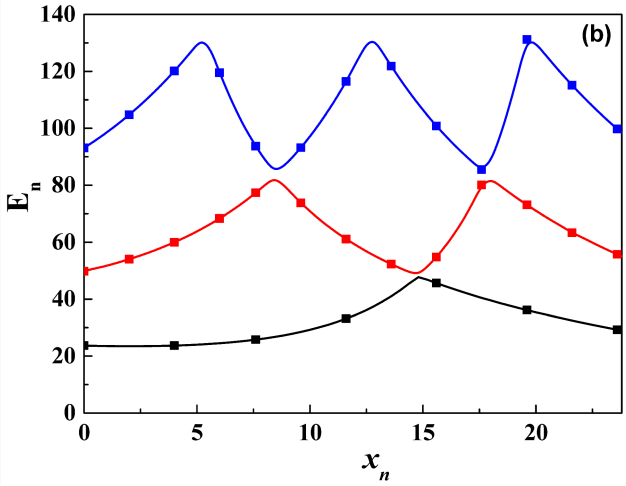

Figure 2: Example of tabular data used to train the neural network (a) and its graphical visualization (b).

Representing the result in a convenient matrix form, we will obtain:

$$A\Psi^{n+1}=B\Psi^{n},\quad \Psi^{n}=(\Psi_1^n,\dots,\Psi_{N-1}^n)^T,$$

$$A=\begin{pmatrix} 1+\lambda(2\alpha+U_1) & -\lambda\alpha & 0 & \vdots & 0 \\ -\lambda\alpha & 1+\lambda(2\alpha+U_2) & -\lambda\alpha & \ddots & \dots \\ 0 & -\lambda\alpha & \ddots & \ddots & 0 \\ \dots & \ddots & \ddots & 1+\lambda(2\alpha+U_{N-2}) & -\lambda\alpha \\ 0 & \vdots & 0 & -\lambda\alpha & 1+\lambda(2\alpha+U_{N-1}) \end{pmatrix},$$

$$B=\begin{pmatrix} 1-2\lambda(2\alpha+U_1) & 2\lambda\alpha & 0 & \vdots & 0 \\ 2\lambda\alpha & 1-2\lambda(2\alpha+U_2) & 2\lambda\alpha & \ddots & \dots \\ 0 & 2\lambda\alpha & \ddots & \ddots & 0 \\ \dots & \ddots & \ddots & 1-2\lambda(2\alpha+U_{N-2}) & -2\lambda\alpha \\ 0 & \vdots & 0 & 2\lambda\alpha & 1-2\lambda(2\alpha+U_{N-1}) \end{pmatrix}. \quad (6)$$

By now substituting into this difference scheme the data obtained from experimental measurements of the energy spectrum (E_n), we again obtain datasets (in form $x_i = x_i(E_n, t_i)$) convenient for use in training a neural network.

The data obtained by substituting the experimental results into models (5) and (6) underwent minimal preprocessing. The primary goal of the data processing was to organize the values into columns: the first contained the coordinates x_n of the difference scheme corresponding to the partition of the structure's localization region, while the remaining columns contained the spectral values E_n. An example of such tabular data and its visualization is presented in Figures 2a and 2b, respectively.

4. Experiment

We begin working with the input data directly based on model (5), (6). We will accept the input data as vectors, which leads to the following matrix equation:

$$H(U)=E\Psi, \quad (7)$$

where $H(U)$ is a tridiagonal Hamiltonian that depends on the vector of potentials $U = (U_1, U_2, ...,U_{N-1})$. In our case, the experimental spectral data is a set of energy levels E_n. As a result, we need to construct a mapping as follows:

$$f_{NN}:\{E_n, t_i\} \rightarrow \{x_i\}, \tag{8}$$

or is it the same as a dependency relationship:

$$\{U_i\}_{i=1}^{N-1} = U_i(\{x_i\}). \tag{9}$$

To preprocess and normalize the data, we used the Wolfram Mathematica 13.2 environment, which implements this as follows:

```
data = Import["spectra_table.csv"];

xGrid = data[[2 ;; , 1]];
Edata = data[[2 ;; , 2 ;;]];
scaledE = Standardize /@ Transpose[Edata];
scaledE = Transpose[scaledE];

scaledX = (xGrid - Mean[xGrid])/StandardDeviation[xGrid];
dataset = MapThread[Association["Input" -> #1, "Target" -> #2] &, {scaledE, scaledX}];
{train, val} = TakeDrop[dataset, Round[0.8 Length[dataset]]];
```

where in our case scaledXList is a list of corresponding labels for training the neural network: coordinates and U-vectors for each dadaset. The neural network architecture was designed to use the vector E_n itself to reconstruct the vectors x_n and U_n.

For this, we use a multi-layer perceptron (MLP) (NetChain / NetGraph). In Wolfram Mathematica, we implemented this as follows:

```
inputDim = Length[scaledE[[1]]];
outputDim = Length[scaledXList[[1]]];

net = NetChain[{
        LinearLayer[256],
        ElementwiseLayer[Ramp], (* ReLU *)
        DropoutLayer[0.2],
        LinearLayer[128],
        ElementwiseLayer[Tanh],
        LinearLayer[outputDim]
},
"Input" -> inputDim
];
```

When training the neural network, the process was controlled by a loss function in the form:

$$L_{MSE} = \frac{1}{M}\sum_{j=1}^{M}\left|x_j^{(true)} - x_j^{(pred)}\right|^2. \tag{10}$$

In Wolfram Mathematica, this parameter was taken into account automatically using the NetTrain directive:

```
trainedNet = NetTrain[net, train,
ValidationSet -> val,
TrainingProgressReporting -> "Panel",
BatchSize -> 32,
MaxTrainingRounds -> 200,
TargetDevice -> "GPU",
LossFunction -> MeanSquaredLossLayer[]
];
```

After the initial good approximation was made, the predicted values of x_n and U_n were updated by minimizing the discrepancy between the experimental energy data E^{exp} and the eigenvalues $\lambda_j(U)$ obtained in the model (5), (6), where the objective function is as follows:

$$J(U)=\sum_{j=1}^{K}\left[\lambda_j H(U)-E_j^{\exp}\right]^2+\gamma R(U), \tag{11}$$

where γ is the weighting coefficient and the regularizer has the form:

$$R(U)=\sum_{j=1}^{K}(U_{j+1}+U_j)^2. \tag{12}$$

Now the implementation of the construction of a tridiagonal matrix, the formation of the objective function and minimization using a combined scheme looks as follows:

```
makeH[U_List, alpha_] := Module[{n = Length[U]},
SparseArray[{
        Band[{1, 1}] -> 2 alpha + U,
        Band[{1, 2}] -> -alpha,
        Band[{2, 1}] -> -alpha
}, {n, n}]
];

lossFunc[Uvec_?VectorQ] := Module[{H, eig},
H = makeH[Uvec, alphaVal];
eig = Sort[Eigenvalues[H]];
Total[(eig[[1 ;; K]] - Eexp)^2] + gamma Total[(Differences[Uvec])^2]
];

Ustart = NetPredict[trainedNet, singleEInput];

res = FindMinimum[
lossFunc[U],
{U, Ustart, StepMonitor :> Null},
MaxIterations -> 200
];

Uopt = U /. res[[2]];
```

Figure 3 schematically presents the principle of training a neural network using the vector representation of the spectrum E_n as a function of the discrete grid coordinates x_n.

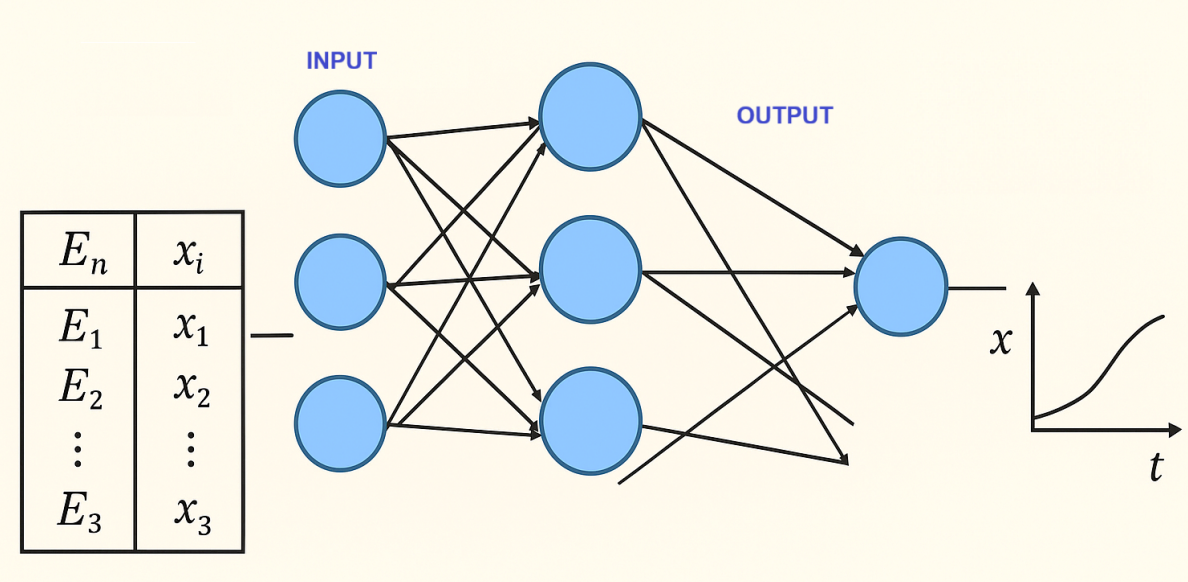

Figure 3: Scheme of training a neural network using tabular spectra $E_n = E_n(x_i)$.

5. Results

Let us first focus on the training process of the neural network used to identify the parameters of the nanosystems in all the samples studied.

Table 1
Results of neural network training for identifying precise nanostructure parameters (PINN approach)

Parameter	Symbol	Value
Number of layers	N_{layers}	5
Neurons per layer	N_{neurons}	64
Activation function	$\varphi(x)$	$\tanh(x)$
Learning rate	η	5×10^{-4}
Optimizer		Adam
Batch size	B	128
Training epochs	N_{epochs}	2000
Loss function type	$\mathscr{L}_{\text{PINN}} = \langle R^2 \rangle + w_{\text{BC}} \langle (\Psi - \Psi_{\text{BC}})^2 \rangle$	PINN residual + boundary
Boundary condition weight	w_{BC}	0.1
Mean residual error	ε_{res}	$2{,}3 \times 10^{-4}$
Potential identification error	$\Delta U(x)$	≤ 1.5%
Computation time	t_{train}	38 min
Hardware used		Intel Core i7-12700H, 16 GB RAM
Framework		Wolfram Mathematica 13.3 (NetTrain)

Table 1 presents the main parameters and initial training results for a neural network implemented using the general Physics-Informed Neural Network (PINN) approach. The neural

network was directly applied to the task of identifying precise nanostructure parameters from experimental data based on the stationary Schrödinger equation in difference form. This model was trained using experimentally derived data from published experimental papers [14–17] (170 datasets taken from 74 structures of different symmetry were used; the criterion for selecting the data was to use all experimentally measured values of the energy spectra related to the geometric configuration of the structure), taking into account only the boundary conditions and the general differential equation scheme. As shown by the obtained results, the use of a loss function that includes a physical (PINN) component and an additional penalty term for the boundary conditions ensures high accuracy in identifying nanostructure confinement with minimal mean square error. The training process was carried out on a personal computer with relatively modest computational resources, demonstrating the efficiency of the proposed implementation without the need for high-performance systems or computing clusters.

A key factor in selecting the datasets used to train the neural network was their consistency with experimental results obtained for various nanostructures possessing diverse physic-chemical parameters and geometric confinement. The only unifying feature among them is that all of these structures are two-dimensional in terms of geometry.

To test the performance of the neural network trained on experimental datasets, experimental measurement data were used that did not overlap with the input data employed for training. Key factors for evaluating the neural network's performance included its speed and its ability to accurately identify parameters compared to other available models. The reference model, currently the standard, is based on self-consistent solutions to the Schrödinger-Poisson system of equations (Schrödinger-Poisson solver). The theoretical foundation of this method was, for example, developed in our previous paper [18].

6. Discussion

To demonstrate the performance of the trained neural network in identifying nanostructure parameters, five samples composed of different materials were randomly selected. The experimental studies from which the verification data were obtained correspond to papers [1, 2, 5, 6, 14]. The computation speed results for the developed neural network, compared to the direct analytical method, are presented in Table 2. As shown in Table 2, the developed neural network identifies the geometric confinement parameters of nanostructures in all samples nearly an order of magnitude faster than the conventional direct calculation method. Moreover, the neural network's execution time is independent of the sample type, as the input and experimental data were previously normalized so that their weights were approximately equal.

Table 2
Comparison of computational speed of neural network versus direct calculation method

Sample No	Neural Network Time (min)	Direct Method Time (min)
1	23	210
2	16	187
3	14,2	114
4	17	142
5	27,1	217

Further, Figures 4a and 4b present the results of calculating the full spatial confinement for two different nanostructure samples using both the neural network and the direct computational method. Moreover, both samples were fabricated from entirely different types of semiconductors: arsenide (a) and nitride (b). As shown, both methods produce virtually identical results; however, the neural network-based approach, due to its speed, is more efficient and streamlines the

computational process. This represents a significant advantage for direct applications in nanoelectronics, as it considerably facilitates the automation of computational work with samples.

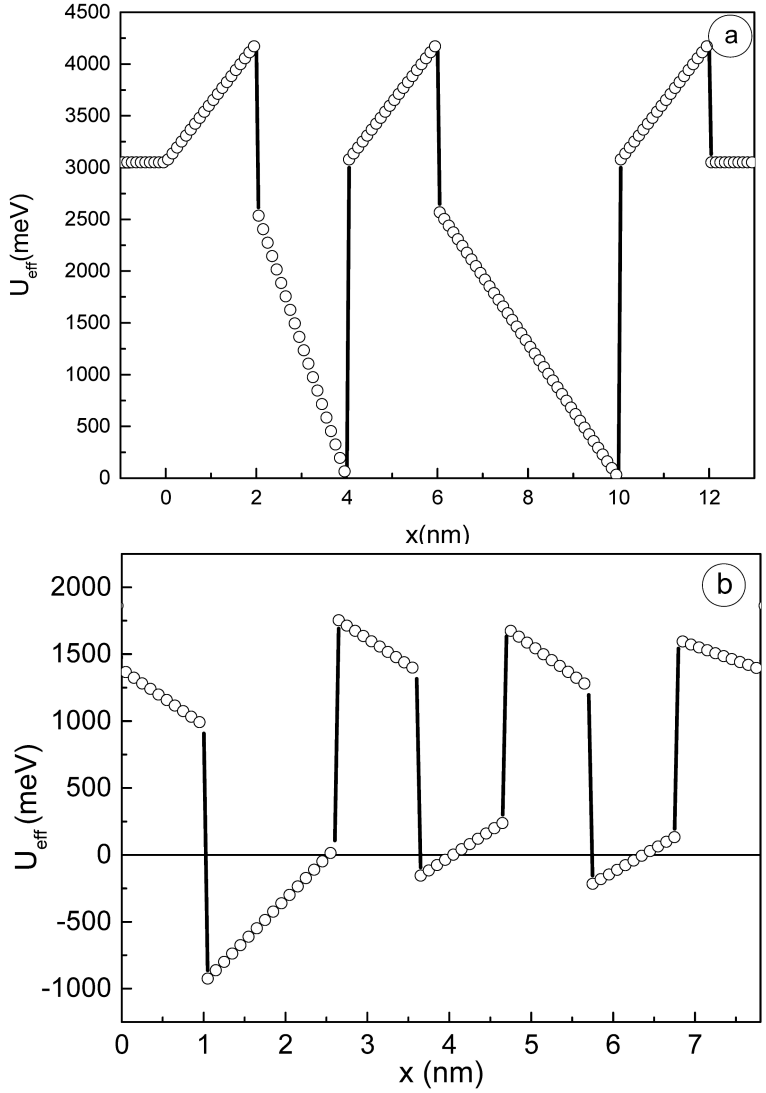

Figure 4: The potential profiles of two nanostructure samples were calculated using two methods: a trained neural network based on experimental data (solid line) and the direct method (circles).

7. Conclusions

An approach to the automated identification of geometric and spatial confinement in low-dimensional structures using machine learning methods is proposed.

The developed neural network, trained on input experimental data corresponding to measured electronic spectra and aligned with a difference scheme based on a mathematical model of the low-dimensional system, enables efficient identification of nanostructure shapes with arbitrary input physical parameters. Its performance is nearly one order of magnitude higher than that of the Schrödinger-Poisson model.

The obtained results demonstrate that the proposed methodology provides efficient processing of experimental samples and reliable identification of their parameters. These results show promise for the development of software with broad applications in specialized areas of nanotechnology and electronics.

Declaration on Generative AI

The author's have not employed any Generative AI tools.

References

[1] I. Heckelmann, M. Bertrand, A. Forrer, M. Shahmohammadi, M. Beck, J. Faist, Measurement of sub-poissonian shot noise in a quantum cascade detector, Appl. Phys. Lett. 124 (2024). doi:10.1063/5.0196803.

[2] Y. Hu, S. Suri, J. Kirch, B. Knipfer, S. Jacobs, S. K. Nair, Z. Zhou, Z. Yu, D. Botez, L. J. Mawst, Large-scale data generation for quantum cascade laser active-region design with automated wavefunction identification, Appl. Phys. Lett. 124 (2024). doi:10.1063/5.0209613.

[3] T.-S. Nguyen, C. Savant, A. Asteris, H. G. Xing, D. Jena, Transport properties of lattice-matched AlScN/GaN single- and multichannel heterostructures, Appl. Phys. Lett. 127 (2025). doi:10.1063/5.0281623.

[4] A. A. McAsule, M. M. Halim, Zinc oxide nanostructured random lasers: a review of their potential as light sources for bioimaging and biosensing applications, J. Appl. Phys. 138 (2025). doi:10.1063/5.0285614.

[5] B. Zhou, S. Zhang, Y. Ma, Z. Ma, H. Dong, R. Li, Y. Zhu, N. Zhuo, S. Zhai, J. Zhang, F. Liu, Q. Lu, On-chip wide tuning of high-power quantum cascade laser based on a vertical-integrated heater, APL Photonics 10 (2025). doi:10.1063/5.0284535.

[6] M. T. Vaughan, W. Michailow, R. Xia, L. Li, M. Salih, H. E. Beere, D. A. Ritchie, E. H. Linfield, A. G. Davies, J. R. Freeman, J. E. Cunningham, Free-space and polarization maintaining delivery of terahertz radiation from a quantum cascade laser in a dry dilution refrigerator, Rev. Sci. Instrum. 96 (2025). doi:10.1063/5.0250844.

[7] T. Bonazzi, H. Dely, P. Didier, D. Gacemi, B. Fix, M. Beck, J. Faist, A. Harouri, I. Sagnes, F. Grillot, A. Vasanelli, C. Sirtori, Metamaterial unipolar quantum optoelectronics for mid-infrared free-space optics, APL Photonics 9 (2024). doi:10.1063/5.0225920.

[8] V. Kumar, P. Mukherjee, F. Fauquet, K. Khare, S. Gigan, P. Mounaix, Computational terahertz phase imaging using a sequence of multi-plane intensity measurements, APL Photonics 10 (2025). doi:10.1063/5.0283596.

[9] O. S. K. S. Sastri, A. Sharma, A. Awasthi, Constructing inverse scattering potentials for charged particles using a reference potential approach, Phys. Rev. C 109 (2024). doi:10.1103/PhysRevC.109.064004.

[10] Z. Zhang, L. Zhang, X. Wang, Rational design of graphyne-based dual-atom site catalysts for CO oxidation, Nano Res. 16 (2023) 343–351. doi:10.1007/s12274-022-4823-3.

[11] D. Marian, E. G. Marin, M. Perucchini, Multi-scale simulations of two-dimensional material-based devices: the NanoTCAD ViDES suite, J. Comput. Electron. 22 (2023) 1327–1337. doi:10.1007/s10825-023-02048-2.

[12] T. K. Sodhi, P. Chrétien, Q. C. Bui, A. Chevillard, L. Travers, M. Morassi, M. Tchernycheva, F. Houzé, N. Gogneau, Surface charge: an advantage for the piezoelectric properties of GaN nanowires, Nanoenergy Advances 4 (2024) 133–146. doi:10.3390/nanoenergyadv4020008.

[13] Y. Hu, S. Suri, J. Kirch, B. Knipfer, S. Jacobs, Z. Yu, D. Botez, L. J. Mawst, Enhancing quantum cascade laser active region design through inverse neural networks: a machine learning approach to metric-based structure generation, AIP Adv. 14 (2024). doi:10.1063/5.0227270.

[14] G. Scalari, J. Faist, 30 years of the quantum cascade laser, Commun. Phys. 7 (2024). doi:10.1038/s42005-024-01888-z.

[15] A. Khalatpour, A. K. Paulsen, C. Deimert, Z. R. Wasilewski, Q. Hu, High-power portable terahertz laser systems, Nat. Photonics 15 (2021) 16–20. doi:10.1038/s41566-020-00707-5.

[16] D. Botez, M. A. Belkin (Eds.), Mid-infrared and terahertz quantum cascade lasers, Cambridge University Press, Cambridge, UK, 2023.

[17] L. A. Sterczewski, et al., Terahertz hyperspectral imaging with dual chip-scale combs, Optica 6 (2019) 766–771. doi:10.1364/OPTICA.6.000766.

[18] I. Boyko, Analytical method for calculation of the potential profiles of nitride-based resonance tunneling structures, Condens. Matter Phys. 21 (2018). doi:10.5488/CMP.21.43701.