communications physics

A Nature Portfolio journal

Article

https://doi.org/10.1038/s42005-025-02255-2

Genetically programmable optical random neural networks

Check for updates

Bora Çarpınlıoğlu & Uğur Teğin

Today, machine learning tools, particularly artificial neural networks, have become crucial for diverse applications. However, current digital computing tools to train and deploy artificial neural networks often struggle with massive data sizes and high power consumptions. Optical computing provides inherent parallelism accommodating high-resolution input data and performs fundamental operations with passive optical components. However, most of the optical computing platforms suffer from relatively low accuracies for machine learning tasks due to fixed connections while avoiding complex and sensitive techniques. Here, we demonstrate a genetically programmable yet simple optical neural network to achieve high performances with optical random projection. By programming the orientation of the scattering medium which acts as a random projection kernel and only using 1% of the search space, our technique finds an optimum kernel and improves initial test accuracies by 8–41% for various machine learning tasks. Through numerical simulations and experiments on a number of datasets, we validate the programmability and high-resolution sample processing capabilities of our design. Our optical computing method presents a promising approach to achieve high performance in optical neural networks with a simple and scalable design.

Nowadays, artificial neural networks are heavily used for learning from data to perform intricate tasks of high complexity[1]. In the last decade, machine learning applications have started to cover diverse fields, from autonomous driving to healthcare diagnosis, and demonstrate outstanding success in real-life applications[2]. With the availability of high volumes of data, the trend in machine learning is to develop computationally complex and data-hungry giant models. This increasing complexity in machine learning models adds computational and environmental challenges during training and inference steps[3]. Moreover, the currently used digital computing platforms, which are based on Turing–von Neumann architecture, are notoriously limited by the memory access bottleneck and cannot keep up with the demands of large models efficiently. The current transistor-based computing methods are reaching the physical and technological limits, indicating the end of Moore's Law.

Since all the information in the macroscopic world is natively analog, analog computing is an intuitive alternative hardware solution for creating and deploying machine learning models. Unlike digital computing, which processes information in discrete values (0s and 1s), analog computing operates on continuous signals; thus, it is compatible with complex, high-dimensional data by virtue of the complex, nonlinear, and high-dimensional physical systems that encode and evaluate the problem of interest. However, analog computing is intrinsically noisy and not as precise as digital computing. Neural networks' relative robustness against noise and device imperfections makes them ideal applications for analog computing solutions. In recent decades, analog computing tools such as brain-inspired (neuromorphic) computing methods, including approaches like reservoir computing[4] and extreme learning machines[5,6], have emerged. With fixed connections in the reservoir or hidden layers, these methods provide decent accuracy with less training effort.

Optical computing, a branch of analog computing, serves as a prominent candidate to accelerate machine learning efforts by offering several key operations like the Fourier transform, convolution operations, and matrix multiplications intrinsically[7]. This rich framework has led to the development of optical computers capable of pattern recognition[8] and the rapid optical implementation of neural network models[9]. However, initial efforts faced challenges in controlling/training optical connections and implementing nonlinear connections. With the introduction of neuromorphic computing methods in the 2000s, the training problem of optical computing has been partially circumvented. However, training of physical neural networks, in particular optical neural networks, is still a major concern as more and more works strive for full hardware implementations[10]. Recently, interest in optical neural networks has grown with free space, fiber, and integrated designs[11–15]. In its simple form, implementing artificial neural networks efficiently and rapidly in optics can be achieved by utilizing optical diffraction, metasurfaces, wave propagation, and passive optical elements like lenses and/or scattering media[16,17]. In particular, using a disordered

Department of Electrical and Electronics Engineering, Koç University, Istanbul, Türkiye. e-mail: utegin@ku.edu.tr

medium comprising millions of fully connected photonic nodes serves as a static random reservoir[18] with random matrix multiplication and offers random mapping of data for dimensionality reduction[19]. By incorporating digital feedback loops into the fixed random mapping of a disordered medium, optical neural networks are demonstrated for chaotic system prediction[20,21]. Most recently, the rapidly growing field of controlling methods of light propagation in disordered media has produced advancements with nonlinear photonic neural networks[22,23], new schemes for optical recurrent reservoir computing[24], and random optical encoders in the integrated domain[25]. Overall, these studies show the potential of controlling and programming light propagation in disordered media for highly performant optical computing architectures.

In order to control light propagation in disordered media, the transmission matrix (TM) formalism is sometimes adopted. However, due to the unavailability of phase information, a high number of modes, and all the angles and polarizations, such a characterization is difficult in practice[19,26], and computational approaches have been developed[27,28] to ease the process of manipulating the TM.

Here, we present the first genetically programmable optical random neural network without the computational overload of determining the full TM and improve the performance of scattering-based optical computing platforms for machine learning applications. Our novel method is based on finding an optimum random projection kernel to map information optically for designated machine learning applications by altering the disordered medium (diffuser) orientation inside the optical neural network. We employ an evolutionary search algorithm (genetic algorithm) and significantly increase accuracy for various datasets by using only 1% of the search space to find optimum random projection kernels. Our simple and genetically programmable (resulting from the use of a genetic algorithm to program random projection kernels) optical neural network has been tested in numerical simulations and experiments using Breast MNIST[29], Covid-19 X-Ray[30], and Fashion MNIST[31] datasets, which have varying resolutions. Test accuracies are improved from 74% to 82%, 78% to 93%, and 47% to 88%, respectively. Our optical computing scheme thus allows for significant improvements without having to implement full tunability, which is a major hurdle in physical neural network designs.

Results

Numerical studies

Before constructing the experimental setup, we make use of an accurate beam propagation method (BPM) simulation to validate our main idea of controlling optical random projection. The details of the simulation are given in Supplementary Note 1. Here, our design leverages Johnson-Lindenstrauss Lemma[32], which states that random projection can be used for dimensionality reduction purposes[33], which is at the core of learning from high-dimensional data. We benchmark our idea on a binary classification task using the Breast MNIST dataset containing 780 samples, and with an 80/20 train-test split, we achieve the GA accuracy evolution plot presented in Fig. 1c. In GA accuracy plots, the fitness of each candidate angular orientation is denoted with a dot. Boxes extend from the first quartile (Q1) to the third quartile (Q3) for a given generation, and the whiskers extend to the farthest data point lying within 1.5× the inter-quartile range (Q3–Q1). Flier points are those past the end of the whiskers. The median value in a generation is denoted with a horizontal line. As can be seen from the figure, simulation results show improvements over the baseline ridge classification accuracy of 66.67%, which corresponds to unprocessed samples. After the first iteration in GA, a ridge classification accuracy of 67.31% is reported, which is already above the baseline performance. Based on the classification accuracies in the following iterations, GA proposes an optimal point where the ridge classification accuracy reached its maximum, 77.56%. We present the improvement in classification accuracy by plotting confusion matrices at the beginning (Fig. 1a) and the end (Fig. 1b) of the programming process. The confusion matrices are normalized such that entries on the same row

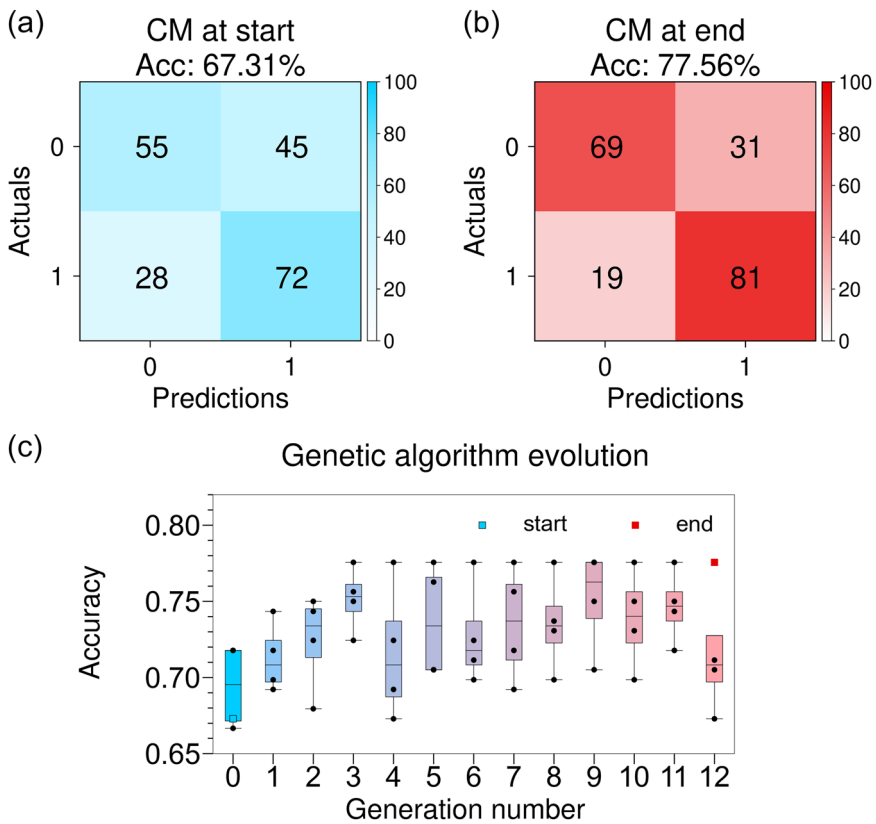


Fig. 1 | Simulated learning results for the Breast MNIST dataset. Row-wise normalized confusion matrices (CMs) corresponding to the first (**a**) and the last (**b**) genetic algorithm (GA) iteration. As classification accuracy (Acc) increases, the confusion matrices become more diagonal. **c** Evolution of the accuracy during GA. The starting and the best accuracy metrics are reported with blue- and red-filled squares, respectively. The fitness metric of each gene is denoted with a data point, and the distribution of fitness metrics for a given generation is visualized with a box–whisker plot. Box edges range from the first quartile (Q1) to the third quartile (Q3), and the whiskers (error bars) extend to the farthest data point lying within 1.5× the inter-quartile range (Q3–Q1). The median value in a generation is illustrated with a horizontal line.

sum up to 100, barring rounding errors. These confusion matrices for the simulated Breast MNIST dataset demonstrate that classification performance can be significantly improved by programming the optical computing platform and searching for an optimal random projection kernel heuristically. After validating the programmability of optical random projection, we move on to the experimental setup and results in the next subsection.

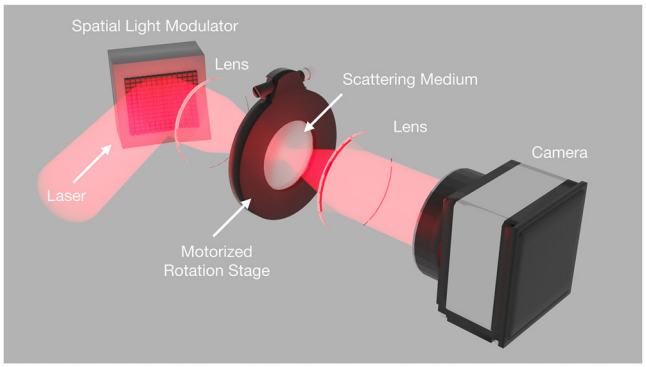


Fig. 2 | Schematic of random projection-based programmable optical computing setup. The experimental setup consists of a spatial light modulator for encoding information onto laser light, a scattering medium on an off-axis rotation stage, and an imaging system to relay light and a camera to decode information. The scattering medium performs random matrix multiplication in the complex Fourier domain, corresponding to convolution in the space domain. The off-axis alignment of the center of rotation of the disk produces the intermediate variation of an approximate translation of the random pattern along with a minor rotary effect at the detection plane. Therefore, at each angular orientation, pixel values at the camera are modified either partially or completely.

Experimental studies

Experimental realization of our genetically programmable optical random neural network is illustrated in Fig. 2. It consists of a continuous-wave (CW) laser source, a spatial light modulator (SLM), a pair of lenses to perform Fourier transforms, a scattering medium placed on a disc located on the Fourier plane to provide optical random projection and a camera to collect optically processed information. The optical information encoding process is carried out by superimposing each sample of the dataset, an 8-bit image, as a phase pattern on the SLM. After the encoding step, the information-carrying optical field is processed linearly all the way up until the recording plane with no power consumption. Optical random projection/mapping is realized with a diffuser.

When coherent light passes through the diffuser, different optical free paths are formed at different points on the scattering media, and at the camera plane, we obtain a speckle pattern. A small change in the orientation of the diffuser changes the result of the element-wise matrix multiplication. Since the element-wise matrix multiplication is performed in the Fourier domain, a convolution is performed in the space domain. In other words, the convolution kernel is dependent on the orientation of the scattering medium. Therefore, we propose searching for and finding an optimum kernel (diffuser surface) with better feature extraction capabilities. Since the information-carrying laser beam occupies a small region on the diffuser, introducing a new region to it by rotating the diffuser changes the random projection kernel. For this purpose, we constructed a disk covered with adhesive tape, which resulted in a search space where there is only one degree of freedom (angular position), allowing us to search for better kernels easily. The center of rotation of the disk is positioned off-axis, meaning that rotating the disk corresponds to the intermediate variation of an approximate translation of the random pattern along with a minor rotary effect at the detection plane. If the center of rotation of the disk were aligned on-axis, a mere rotation of the random pattern would have been obtained at the detection plane, leaving the pixel values unaffected. In such a scenario, since

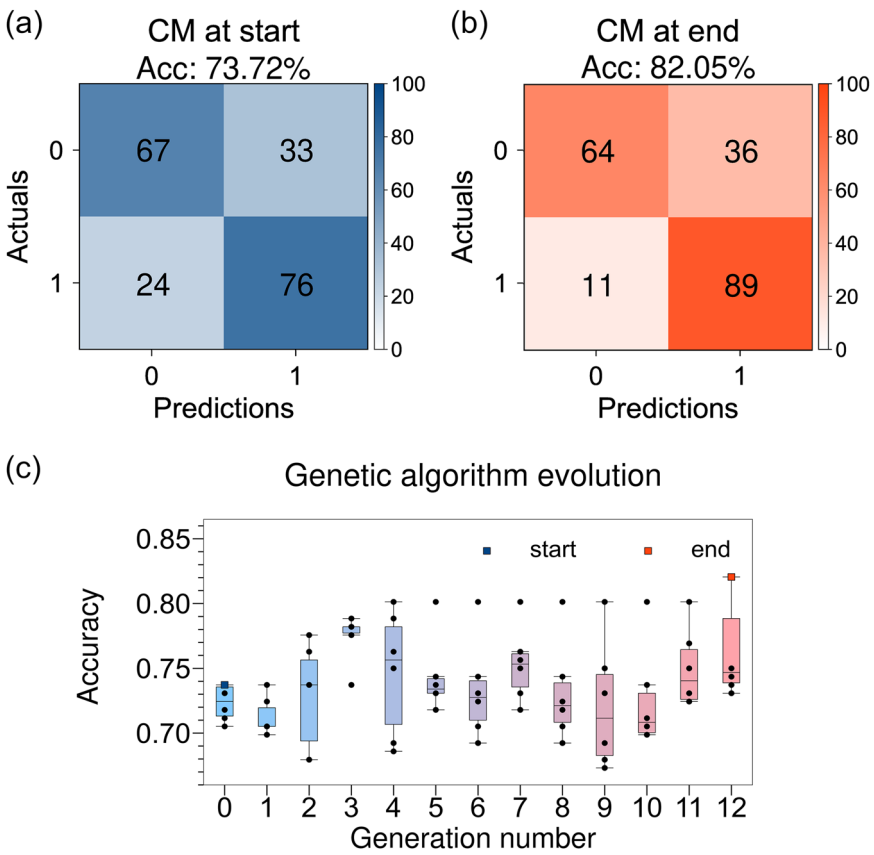


Fig. 3 | Experimental learning results for the Breast MNIST dataset. Row-wise normalized confusion matrices (CMs) corresponding to the first (**a**) and the last (**b**) genetic algorithm (GA) iteration. **c** Evolution of the accuracy (Acc) during GA illustrated with a box–whisker plot. As opposed to other experimental results, a population size of $p = 6$ is employed. The starting and the best accuracy metrics are reported with blue- and red-filled squares, respectively. The fitness metric of each gene is denoted with a data point, and the distribution of fitness metrics for a given generation is visualized with a box–whisker plot. Box edges range from the first quartile (Q1) to the third quartile (Q3), and the whiskers (error bars) extend to the farthest data point lying within 1.5× the inter-quartile range (Q3–Q1). The median value in a generation is illustrated with a horizontal line.

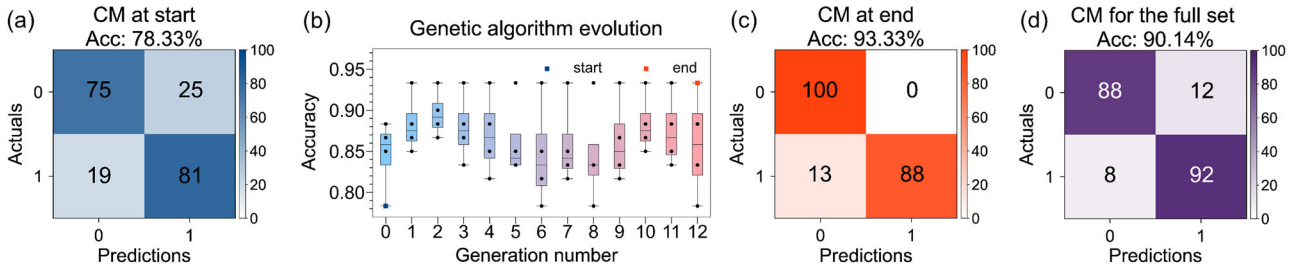


Fig. 4 | Experimental learning results for the COVID-19 X-Ray dataset when a subset of the full dataset is used for the genetic algorithm (GA). a Confusion matrix (CM) corresponding to the first GA iteration. **b** Evolution of the accuracy (Acc) during GA illustrated with a box–whisker plot. **c** CM was obtained at the end of GA. **d** CM corresponding to the full dataset when all the samples are passed through the optimal angular position yielded by GA. The starting and the best accuracy metrics are reported with blue- and red-filled squares, respectively. The fitness metric of each gene is denoted with a data point, and the distribution of fitness metrics for a given generation is visualized with a box–whisker plot. Box edges range from the first quartile (Q1) to the third quartile (Q3), and the whiskers (error bars) extend to the farthest data point lying within 1.5× the inter-quartile range (Q3–Q1). The median value in a generation is illustrated with a horizontal line.

rotation is a linear operation, the readout layer would have produced identical accuracy metrics through iterations. The difference between aligning the center of rotation of the disk on- and off-axis is further illustrated in Fig. S3 of Supplementary Note 5.

A stepper motor is employed to control the orientation of the diffuser with the half-stepping method, where one full rotation of the disk is divided into 4096 steps. Without benefiting from optimization algorithms and trying each possible kernel individually, evaluating the resulting accuracy would take a long time experimentally. Thus, we utilize a heuristic search algorithm, genetic algorithm (GA)[34], which is also employed in the training and design of optical neural networks[35–37], for optimizing for the maximum accuracy, which is a function of the angular position, θ. Such a method requires several trials to meet the designated task and decreases optimization time in experiments. In the broader spectrum, GA lies within model-free (or blackbox) optimization algorithms[38]. In contrast to model-based approaches, it does not require full knowledge of the system, and the classification accuracy can be effectively parametrized with a single parameter, which is the angular orientation of the disk. Although model-based approaches allow the system to be modeled with a collection of random matrices, memory issues arise with 4096 different complex, random matrices. Therefore, the choice of GA simplifies the optimization process to a large degree. The deployment of a heuristic search algorithm like GA can also be attributed to the uneven structure of the complex media, where there is no a priori relationship between the complex random matrices obtained from distinct points significantly far from each other. However, we want to emphasize that since the rotation steps are small, similarities in output images corresponding to matrices originating from a neighborhood of consecutive angular positions are observed in our experiments (see Supplementary Note 7 for quantitative metrics). GA parameters used in this study and the backend processing steps are explained in the "Methods" section.

In experiments, we initially employ GA on the Breast MNIST dataset and obtain the GA evolution plot shown in Fig. 3c. With the same 80/20 train-test split as in simulations, the ridge classification accuracy is improved to 82.05% starting from 73.72%. The confusion matrices corresponding to the first and the best GA iteration are given in Fig. 3a, b, respectively. To further ensure the claimed improvement is not due to noise, we perform 5-fold cross-validation on randomly projected samples corresponding to the best GA iteration. As a result, a mean classification accuracy of 73.08% with a standard deviation of 7.95% was obtained. For comparison, the first GA iteration produces 5-fold cross-validation metrics with a mean classification accuracy of 71.28% with a standard deviation of 2.88%. We additionally performed a paired-sample Student's t-test, and the returned value of $h = 1$ indicates that the t-test does reject the null hypothesis at the 1% significance level.

After experimentally demonstrating and validating the tunability of optical random projection on a small-scale dataset with samples having 28×28 pixels resolution, we note that we can accommodate high-resolution samples in our experimental setup and fully leverage the parallelism provided by our optical computing platform, where we are only limited by the SLM resolution, which is 600×800 pixels. For this reason, we decided to scale up and tackle a more complex classification task using the COVID-19 X-Ray[24] dataset with higher resolution (400×400 pixels) samples. However, the SLM used in this study operates at a 60 Hz repetition rate since liquid crystal technology is limited in terms of speed. To decrease the time spent on programming the random projection kernel and make our optical computing method useful for larger datasets, we propose to form a smaller subset of the dataset to be considered a proxy for the full set for the programming step. For this purpose, out of the full dataset (2481 samples), a subset of size 300 containing randomly selected 150 positive and 150 negative samples is created. As previously, an 80/20 train-test split is used. This way, our goal is to reduce the programming time while maintaining the improvements in classification accuracy. As demonstrated in Fig. 4b, we observe a significant increase in the ridge classification accuracy from 78.33% to 93.33% while programming the optical computing platform with GA.

To evaluate the performance of the subset method, we optically process all the samples of the COVID-19 X-Ray dataset with the optimized random projection kernel proposed by GA, employing an 80/20 train-test split on the whole dataset. An accuracy of 90.14% is achieved over the baseline accuracy level of the dataset, which is 74.85%. These results demonstrate that the genetically programmed optical computing platform provides 15% higher accuracy. In contrast, when we use GA for the full dataset instead of 300 samples, we observed a maximum accuracy level of 91.15%. It shows that by utilizing a randomly selected subset and decreasing the programming time significantly, we sacrifice a 1% performance difference in classification, and even a small subset gives rise to improvements in classification accuracy. We can observe the evolution of accuracy via the confusion matrices presented in Fig. 4a, c. The confusion matrix corresponding to the entire dataset is also given in Fig. 4d. With the COVID-19 X-Ray dataset results, we validate the high-resolution sample processing capability of optical random projection on top of programmability demonstrated earlier. We would like to note that such high resolutions are atypical in conventional machine learning frameworks.

After experimentally validating the programmability and high-resolution sample processing capabilities of our scheme, we explored the popular Fashion MNIST dataset with 70,000 samples[31] to benchmark the performance of our programmable optical random projection scheme and to evaluate our method's performance for multilabel classification. Similar to the COVID-19 X-Ray dataset, we utilize a subset to decrease the programming time of our optical neural network. Thus, a subset of randomly selected 3000 samples with an 80/20 train-test split is created from the original dataset. To set a baseline for our method, we apply the ridge classification over the selected subset and obtain 73.83% accuracy. At the beginning of the programming, after the first GA iteration, an accuracy of 71.33% is obtained. At the end of the programming, this classification

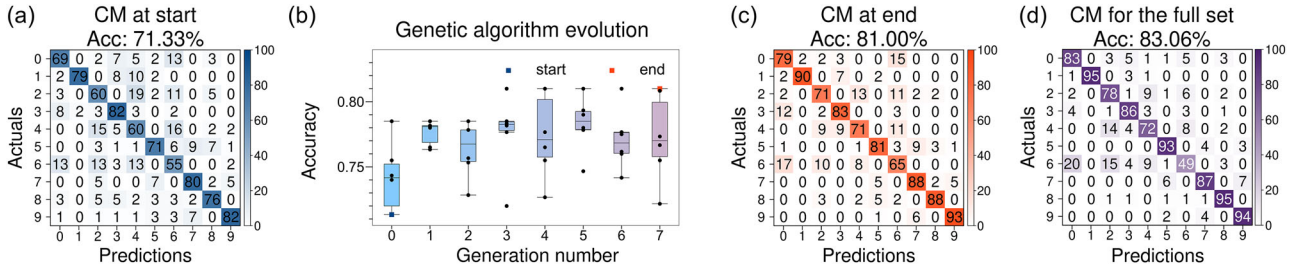


Fig. 5 | Experimental learning results for the Fashion MNIST dataset when a subset of the full dataset is used for the genetic algorithm (GA). a Confusion matrix (CM) corresponding to the first GA iteration. **b** Evolution of the accuracy (Acc) during GA illustrated with a box–whisker plot. **c** CM was obtained at the end of GA. **d** CM corresponding to the full dataset when all the samples are passed through the optimal angular position yielded by GA. As opposed to other experimental results, a population size of $p = 6$ and $n = 7$ generations are employed. The starting and the best accuracy metrics are reported with blue- and red-filled squares, respectively. The fitness metric of each gene is denoted with a data point, and the distribution of fitness metrics for a given generation is visualized with a box–whisker plot. Box edges range from the first quartile (Q1) to the third quartile (Q3), and the whiskers (error bars) extend to the farthest data point lying within 1.5× the inter-quartile range (Q3–Q1). The median value in a generation is illustrated with a horizontal line.

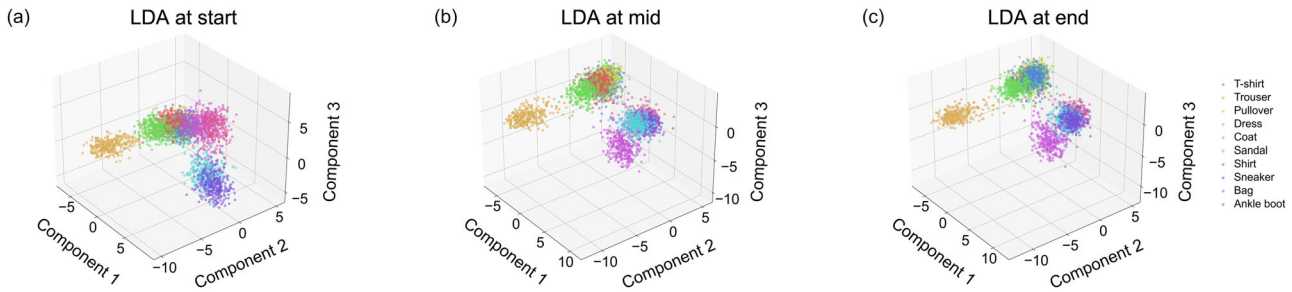


Fig. 6 | Three-dimensional linear discriminant analysis (LDA) plots obtained through genetic algorithm (GA) iterations performed on a subset of the Fashion MNIST dataset. a LDA results before genetic algorithm (GA) iterations. **b** LDA results for an arbitrary iteration of GA. **c** LDA results after GA has found the optimal random projection kernel. The separability between different classes, denoted by different colors, increases as GA progresses to increase classification accuracy. A two-dimensional LDA plot and three-dimensional LDA plots from different elevation and azimuthal angles are given in Supplementary Note 13.

accuracy is increased to 81.00% (Fig. S7b), resulting in an approximately 10% improvement. The entire Fashion MNIST dataset with 60,000 training and 10,000 test samples is optically processed for the programmed condition, and the classification accuracy of 83.06% is achieved. The row-wise normalized confusion matrices captured during the evolution of GA (Fig. 5a, c) and inference (Fig. 5d) further supported our claim that programming random neural networks effectively decreases classification errors. Additional results on high-resolution datasets can be found in Supplementary Notes 10 and 11.

Discussion

To understand the mechanism behind programming random projection with a genetic algorithm, we perform linear discriminant analysis (LDA), a dimensionality reduction technique, on one of the widely-used datasets for benchmarking different machine learning architectures, which is the Fashion MNIST dataset[31]. We feed randomly projected samples to LDA, which helps us to illustrate the inner workings of our optical computing platform. In Fig. 6, 960 features obtained from the randomly mapped subset (3000 samples) of the Fashion MNIST dataset are represented by three components. The distribution of color-coded classes shows that the clustering performance of optical random projection improves as GA progresses. The clustering of similar samples eases the task of the readout layer, thus increasing the performance of the optical computing method.

Without sacrificing the simplicity of the demonstrated optical computing platform, a nonlinear activation function such as the inverse rectified linear unit (ReLU) function can be implemented by partially saturating the measured beam profiles via either increasing the laser power or tuning the camera parameters. In the literature, the effect of such an additional nonlinearity has been reported[15]. It can increase the accuracy of the machine learning tasks when incorporated with the absolute square measurement introduced by the intensity detection of the camera at the readout layer.

In our study, the preferred camera can record 8-bit grayscale images. However, we emulate a camera capable of representing images using fewer bits to test our approach in data-scarce settings by mapping the range 0-255 to a smaller range. We observe that with such a device, faster data rates would be enabled without sacrificing the classification accuracy (see Fig. S16 and Table S1 in Supplementary Note 14). We conclude that with suitable hardware tailored for modern machine vision tasks requiring real-time performance, our programmable computing platform can continue to perform machine learning tasks with high accuracy.

The presented programmable optical computing platform can be further improved with alternative programming algorithms. For example, one can use surrogate optimization tools[39] instead of GA, choose a different/learnable weighting parameter α during ridge classification, or change the pool size used in local averaging, which would affect the number of features pertaining to each captured image. Especially, we note that the α parameter during ridge classification can be nontrivial, and an optimal α should be searched for a certain task for a higher classification accuracy. We have kept α fixed to better assess the performance of the optical configuration. We believe formulating alternative fitness metrics based on continuous loss functions such as cross-entropy loss and mean-squared error instead of the raw classification accuracy will be beneficial over the long term, since classification accuracy is a coarse signal, shown in Fig. S12 of Supplementary Note 12. Imbalanced datasets can also be handled by optimizing the F-score[40] during GA iterations, which would balance the precision and recall properties of the optical model. Although we have yet to opt to make these changes for the sake of simplicity and not losing the essence of our endeavor, they can potentially increase the ridge classification accuracy of the readout

layer. In addition, our approach can be augmented by using the recurrence relations of the reservoir.

In all experiments, phase objects were processed and classified. However, amplitude objects and, more generally, incoherent data can be processed using our approach as well. In this case, the speckle response will be different than the coherent case because constructive/destructive interference will not occur under broadband illumination. Since the irradiances will only be superposed without the contribution from cross-terms, the complexity of the system will be lower, and the feature extraction capabilities of the incoherent setup will be affected.

Broader applicability of our method can be realized by imposing gratings in x- and y-axes on input samples, where this time, instead of angular position, the grating x and y values are to be optimized. Therefore, programmable optical random projection in our optical computing platform without any moving parts can be constructed in principle.

Energy consumption in our setup is only due to the power requirements of components such as the laser source, SLM, and camera. All processing, including the random matrix multiplication, is done passively, and our computing platform relies on linear optics. It indicates that the consumed energy is independent of the sample resolution. Energy consumption in our method and conventional electronics is further compared in Supplementary Note 16. Our work opens up an avenue for energy-efficient, resolution-agnostic, and programmable optical processing schemes for success in modern machine learning practice.

Methods

Physical model of a genetically programmable optical random neural network

Here $A(x, y)$ and $\phi(x, y)$ represent the amplitude and the phase of the incoming laser beam and $S(x, y)$ denotes a sample of a dataset. With the help of a spatial light modulator (SLM), they create the following expression,

$$O(x, y) = A(x, y)\exp[i\phi(x, y)]\exp[iS(x, y)] \tag{1}$$

for the laser beam with the encoded information, $O(x, y)$. The mathematical expression that indicates the operation of the optical random neural network can be shown as

$$R(x, y) = F^{-1}\{F\{O(x, y)\}H(f_x, f_y; \theta)\} \tag{2}$$

$$I(x, y) = |R(x, y)|^2 \tag{3}$$

where $H(f_x, f_y; \theta)$ is the complex, random matrix representing the diffuser (its transfer matrix) as parameterized by the angular position of the disk, θ, $R(x, y)$ is the received optical field at the camera plane, and $I(x, y)$ its intensity is recorded by the camera. Here, x and y denote spatial coordinates, while f_x and f_y denote spatial frequency coordinates. F and F^{-1} denote the Fourier and inverse Fourier transforms, respectively.

The classification operation performed by the readout layer can be expressed as

$$w^* = \mathrm{argmin}_w |y_l - Xw|^2 + \alpha|w|^2 \tag{4}$$

where the optimum weights w^* are inferred from a training set with images X and corresponding labels y_l. In our experiments, the training set with images X corresponds to randomly mapped input samples for a given dataset. α governs the regularization strength, which is set to 1 throughout all analyses.

Experimental procedure

A continuous-wave laser source with a central wavelength of 1064 nm is used for illuminating a spatial light modulator (SLM, HAMAMATSU X10468) with a 800×600 pixels resolution. Since the samples we used originate from datasets developed for machine learning tasks, they typically have low resolution, e.g., 28×28 pixels. To be able to maximally use the effective area of our SLM, we performed image upsampling by an integer factor for low-resolution datasets. For datasets with samples having higher resolution than our SLM resolution, we performed image downsampling. The phase encoding process was then carried out as shown by the expressions above.

We placed an adhesive tape at the processing plane (the Fourier plane) as the scattering medium, providing random matrix multiplication. The scattering medium was rotated to different angles using a stepper motor (28BYJ-48) to yield parameterizations of the random matrix. Since the classification accuracies depend on the random matrix values, and no a priori relationship is known among different random matrix distributions corresponding to different angles, a heuristic search algorithm, a genetic algorithm (GA)[34], was utilized for optimizing for the maximum accuracy.

While programming the optical random neural network, at each GA iteration, all the dataset samples are optically processed with a particular candidate random projection kernel and collected with the camera. All camera parameters, including exposure, saturation, and gain, are kept constant throughout the experiments. Captured 2D RGB images are converted to grayscale images and downsampled by local averaging with a pool size of (20, 16) (see Supplementary Note 14 for the selection of the pool size). As a final step, downsampled 2D grayscale images are converted to 1D arrays (flattened), and ridge classification is performed. The fitness function in GA is set as the classification accuracy, and depending on the result, the stepper motor is rotated to the angle yielded by GA. Following this pipeline, GA searches and finds the kernel that provides the best accuracy level for designated datasets.

Genetic algorithm (GA) employs biology-inspired operations such as mutation, crossover, and selection. In order to utilize GA, we formulated a fitness function where higher accuracies were favored, the input parameter (gene) being angular position specified in terms of the steps of the motor in the interval [0,4095], represented as a bitstring for genetic operations, and the output being the ridge classification accuracy of the randomly mapped dataset. We specified 12 generations and a population size of 4 (unless noted otherwise), which is the number of genes in each generation, and explored different GA parameters in Supplementary Note 12. In the first generation, all genes were selected randomly, then genes with superior classification accuracies were chosen as mating parent genes (2 in our case), and a new generation was created by crossover (single-point crossover with 80% probability) and mutation (random mutation with 1% probability) operations. The crossover operation randomly selects an index in the bitstrings of the mating parent genes as the crossover point, and bits to the right of the crossover point are swapped between the parents, creating a new offspring. The mutation operator, on the other hand, randomly inverts the bits of a gene at certain locations. We obtained the optimal kernel like so. To ensure a nondecreasing nature for the accuracy, we kept the best gene in a given generation and passed it on to the next generation. So for each generation, 3 new genes were chosen by GA. In total, the dataset was passed through 40 locations. In general, given genetic algorithm parameters where the number of generations is shown with n and the population size with p, the total number of random projection kernels that are probed is given by the following formula:

$$n(p - 1) + p \tag{5}$$

To calculate the fitness function corresponding to each gene, we used a laptop with an Intel Core i7-7600U CPU and 8 GB RAM that sends commands via serial communication to a microcontroller board (Arduino Uno), which processes step sequences to rotate the stepper motor to the orientation specified by genes. Our experimental procedure, including the hardware flow control and software flow control, is schematically illustrated in Supplementary Note 6.

Data availability

Data underlying the figures are publicly available at Zenodo[41]. Any additional data generated during experiments may be obtained from the authors

upon reasonable request. Datasets used in the study are publicly available through their respective access links.

Code availability

Code related to the results in this work may be obtained from the authors upon reasonable request.

Received: 9 April 2025; Accepted: 31 July 2025;
Published online: 25 August 2025

References

1. Haykin, S. *Neural Networks and Learning Machines*, 3rd edn (Pearson Education India, 2009).
2. Esteva, A. et al. A guide to deep learning in healthcare. *Nat. Med.* **25**, 24–29 (2019).
3. Schmidhuber, J. Deep learning in neural networks: an overview. *Neural Netw.* **61**, 85–117 (2015).
4. Tanaka, G. et al. Recent advances in physical reservoir computing: a review. *Neural Netw.* **115**, 100–123 (2019).
5. Huang, G.-B., Zhu, Q.-Y. & Siew, C.-K. Extreme learning machine: theory and applications. *Neurocomputing* **70**, 489–501 (2006).
6. Kesgin, B. U. & Teğin, U. Implementing the analogous neural network using chaotic strange attractors. *Commun. Eng.* **3**, 99 (2024).
7. Goodman, J. W. *Introduction to Fourier Optics* (Roberts and Company Publishers, 2005).
8. Abu-Mostafa, Y. S. & Psaltis, D. Optical neural computers. *Sci. Am.* **256**, 88–95 (1987).
9. Farhat, N. H., Psaltis, D., Prata, A. & Paek, E. Optical implementation of the Hopfield model. *Appl. Opt.* **24**, 1469–1475 (1985).
10. Momeni, A. et al. Training of physical neural networks. Preprint at https://arxiv.org/abs/2406.03372 (2024).
11. Wetzstein, G. et al. Inference in artificial intelligence with deep optics and photonics. *Nature* **588**, 39–47 (2020).
12. Shen, Y. et al. Deep learning with coherent nanophotonic circuits. *Nat. Photonics* **11**, 441–446 (2017).
13. Teğin, U., Yıldırım, M., Oğuz, I., Moser, C. & Psaltis, D. Scalable optical learning operator. *Nat. Comput. Sci.* **1**, 542–549 (2021).
14. Bueno, J. et al. Reinforcement learning in a large-scale photonic recurrent neural network. *Optica* **5**, 756–760 (2018).
15. Pierangeli, D., Marcucci, G. & Conti, C. Photonic extreme learning machine by free-space optical propagation. *Photonics Res.* **9**, 1446–1454 (2021).
16. Lin, X. et al. All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004–1008 (2018).
17. Pierangeli, D. et al. Living optical random neural network with three dimensional tumor spheroids for cancer morphodynamics. *Commun. Phys.* **3**, 160 (2020).
18. Saade, A. et al. Random projections through multiple optical scattering: approximating kernels at the speed of light. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 6215–6219 (IEEE, 2016).
19. Gigan, S. Imaging and computing with disorder. *Nat. Phys.* **18**, 980–985 (2022).
20. Dong, J., Rafayelyan, M., Krzakala, F. & Gigan, S. Optical reservoir computing using multiple light scattering for chaotic systems prediction. *IEEE J. Sel. Top. Quantum Electron.* **26**, 1–12 (2019).
21. Rafayelyan, M., Dong, J., Tan, Y., Krzakala, F. & Gigan, S. Large-scale optical reservoir computing for spatiotemporal chaotic systems prediction. *Phys. Rev. X* **10**, 041037 (2020).
22. Wang, H. et al. Large-scale photonic computing with nonlinear disordered media. *Nat. Comput. Sci.* **4**, 429–439 (2024).
23. Oguz, I. et al. Programming nonlinear propagation for efficient optical learning machines. *Adv. Photonics* **6**, 016002–016002 (2024).
24. Wang, H. et al. Optical next generation reservoir computing. *Light Sci. Appl.* **14**, 245 (2025).
25. Wang, X. et al. Integrated photonic encoder for low power and high-speed image processing. *Nat. Commun.* **15**, 4510 (2024).
26. Popoff, S. M., Lerosey, G., Fink, M., Boccara, A. C. & Gigan, S. Controlling light through optical disordered media: transmission matrix approach. *N. J. Phys.* **13**, 123021 (2011).
27. Valzania, L. & Gigan, S. Online learning of the transmission matrix of dynamic scattering media. *Optica* **10**, 708–716 (2023).
28. Zhao, S. et al. Single-pixel transmission matrix recovery via two-photon fluorescence. *Sci. Adv.* **10**, eadi3442 (2024).
29. Yang, J. et al. MedMNIST v2-A large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Sci. Data* **10**, 41 (2023).
30. Soares, E., Angelov, P., Biaso, S., Froes, M. H. & Abe, D. K. SARS-CoV-2 CT-scan dataset: a large dataset of real patients CT scans for SARS-CoV-2 identification. Preprint at https://doi.org/10.1101/2020.04.24.20078584 (2020).
31. Xiao, H., Rasul, K. & Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017).
32. Johnson, W. B. & Lindenstrauss, J. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in Modern Analysis and Probability* 189–206 (Am. Math. Soc., 1984).
33. Bingham, E. & Mannila, H. Random projection in dimensionality reduction: applications to image and text data. In *Proc. of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 245–250 (ACM, 2001).
34. Gad, A. F. PyGAD: an intuitive genetic algorithm Python library. Preprint at http://arxiv.org/abs/2106.06158 (2021).
35. Zhang, H. et al. Efficient on-chip training of optical neural networks using genetic algorithm. *ACS Photonics* **8**, 1662–1672 (2021).
36. Shao, R., Zhang, G. & Gong, X. Generalized robust training scheme using genetic algorithm for optical neural networks with imprecise components. *Photonics Res.* **10**, 1868–1876 (2022).
37. Chai, J., Bi, M., Teng, X., Yang, G. & Hu, M. A mixed mutation strategy genetic algorithm for the effective training and design of optical neural networks. *Opt. Fiber Technol.* **82**, 103600 (2024).
38. Skalli, A. et al. Model-free front-to-end training of a large high performance laser neural network. Preprint at https://arxiv.org/abs/2503.16943 (2025).
39. Eriksson, D., Bindel, D. & Shoemaker, C. A. pySOT and POAP: an event-driven asynchronous framework for surrogate optimization. Preprint at https://arxiv.org/abs/1908.00420 (2019).
40. Goutte, C. & Gaussier, E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *European Conference on Information Retrieval* 345–359 (Springer, 2005).
41. Carpinlioglu, B. Genetically programmable optical random neural networks - figures source data. *Zenodo* https://doi.org/10.5281/zenodo.16413770 (2025).

Acknowledgements

We thank Aydoğan Özcan for fruitful discussions and Çağrı Şenel for providing the SLM and laser source used in experiments. This work is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under grant number 123E308.

Author contributions

B.Ç. constructed the experimental setup and conducted numerical simulations and experiments. U.T. conceived the idea behind the study, provided conceptual advice, and was in charge of supervision. Both authors participated in the analysis of results and the writing of the manuscript.

Competing interests

U.T. is an Editorial Board Member for *Communications Physics*, but was not involved in the editorial review of, or the decision to publish this article. All other authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42005-025-02255-2.

Correspondence and requests for materials should be addressed to Uğur Teğin.

Peer review information *Communications Physics* thanks Hao Wang, Anas Skalli, and the other anonymous reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

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

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

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025