nature portfolio

Peer Review File

Hundred-layer photonic deep learning

Corresponding Author: Prof Lu Fang

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
This work introduces perturbations on-chip to decouple computational correlations to eliminate the redundancies and develop deep photonic learning with a single-layer photonic computing (SLiM) chip. The SLiM chip overcomes the depth limitations of optical neural networks, which allows error rates constrained across over 200 layers, and extends spatial depth from millimeter to hundred-meter scale, which enables the 3-D chip clusters. The authors experimentally realized a neural network with 100 layers for image classification, a 0.345 billion-parameter LLM model with 384 layers for text generation, and a 0.192 billion-parameter LLM model with 640 layers for image generation.

The perturbation-introduction approach is novel and attractive, and the experimental results are also impressive. However, the reviewers have some questions and requests for explanation.

1.The structure of the SLiM chip is not illustrative. A structure figure is recommended.
2.Why the perturbations incorporated within the photonic chip can eliminate the redundancies? What do the redundancies refer to? Where do the redundancies come from?
3.How does the single-layer photonic chip minimize the error propagation? Does the “single-layer chip” mean that the photonic chip can process one layer of a LLM? Then, to build a 100-layer network, do the authors utilize 100 single-layer photonic chips? In one layer of LLM, there are many multiplications between vectors and matrices. The SLiM chip appears to complete only one MVM at one computing cycle. How does one SLiM chip correspond to the operations in the LLM?
4.The authors claim that W_i = |G_i *Phi_i|^2 in SI-2.1. When no perturbation, G_i~=G_0, Phi_i~=Phi_0, then W_i=W_0. If the perturbation is introduced, Phi_i != Phi_0, W_i are different. At the same time, the authors claim that the phase distribution appears random. Therefore, the W are random. How can a random W process the information effectively?
5.How is the propagation perturbation injected? Since the propagation perturbation can decorrelate the weights, does it mean that propagation perturbation can be trained and programmed? Is the training completed online or offline? If online, how is it realized? If offline, will the offline training be complicated and cost too much offline computational resources?
6.Besides the SLiM chip configuration and perturbation injection, many operations are processed in the electronic domain, like the nonlinear activation functions and the data transfer among the SLiM chips. During the experiments, how do the authors control the total computing cluster to process billions of parameters of the LLM and complete the LLM computing? How is the coordination of multiple SLiM chips managed for processing billion-parameter LLMs?
7.Regarding the computing power, the 327.68 TOPS is clear, but where does the 335.54 Peta OPS come from?
8.The computing energy 1510.4Peta OPS/W appears to only include the energy cost of the phase shifters. However, the authors claim that the SLiM achieves ExaOPS/W (10^6 TOPS/W) efficiency, which is unacceptable. The authors should assess the energy efficiency more accurately. The energy cost should include energy cost from the phase shifters, weight matrix programming, and detectors, as well as the energy cost to complete the nonlinear activation computing. Otherwise, the energy comparison between the SLiM chips and GPUs which include complete MVM computing is unfair.

In summary, the work is impressive and introduces novel methods to overcome the depth limitations. However, the details of the SLiM chip and the experiments should be described.

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)
- What are the noteworthy results?
The results are very noteworthy, the scale of the transmitter and SLiM chips are very impressive with respect to complexity and scale. Additionally, the packaging, control, and extension to multiple chiplets is state-of-the-art. The demonstrated architecture is novel. Congratulations to the authors for the sucessful results.

-Will the work be of significance to the field and related fields? How does it compare to the established literature? If the work is not original, please provide relevant references.
This work will be of significance within the field of neuromorphic photonics. The performance with respect to depth and scale is equal or greater than other approaches within the field.

-Does the work support the conclusions and claims, or is additional evidence needed?
The claims are very significant and as impressive as this experiment is. I think additional evidence would be useful to back up the claim. Additionally, the flow of the writing and logical conclusions are at time difficult to follow. I think it would be helpful to have more clearly presented performance metrics with respect to RF noise figure and loss per layer. With the addition of noise figure analysis, one could understand the required signal-to-noise of the input signal as well as the scalability of this system without error correction. If the photonic loss and digital loss are 3.04 and 2.96, respectively then only a few layers would be possible. The claim of deep learning within neuromophic photonic approaches is the key limiting factor and claiming to overcome this goal is a significant claim.

-Are there any flaws in the data analysis, interpretation and conclusions? Do these prohibit publication or require revision?
Not identified but also I am not confident with respect to the Machine Learning Data Analysis. It would be helpful to see more optical / electronic analysis with respect to noise and loss.

-Is the methodology sound? Does the work meet the expected standards in your field?
The methodology is sound and meet standards.

-Is there enough detail provided in the methods for the work to be reproduced?
I believe so, but more information regarding how the nonlinearity is achieved and the transfer function would be useful.

Additional Comments:
The experiment presented within this paper is tremendously impressive but the claims made are very significant. I think more information regarding loss per layer, nonlinearity, and noise performance is required. Some of the sections are difficult to comprehend and the writing should be improved before publication.

I believe the introduction should be expanded to focus a bit on the history of photonic neural networks and the need for this approach. There are very condensed statements made by the authors about noise, physical analog computing, but there is no discussion of other research within the field which has focused on tackling these limitations to scaling and analysis of scaling photonic neural networks. I would recommend the authors look into literature on silicon photonic neural networks for analog machine learning approaches, e.g. RF fingerprinting or automatic modulation classification

(Remarks on code availability)

Reviewer #3

(Remarks to the Author)
The authors present a single-layer photonic computing chip that by injecting purposely designed perturbations in both the linear transfer matrix and the detection non-linearity suppresses the classical error-accumulation bottleneck of analog optical neural networks.

The paper is well written, well structured and well documented.
I have some minor concerns:
- Authors state that energy measures rely on resonant phase shifters and excludes laser, ADC/DAC, CMOS drivers and control FPGA. Supplementary Note 4 states these are “additional overhead” yet does not quantify them.
- Authors state to reach 85.2 % accuracy on ImageNet for 100-layer SLiM, far below 91% reached by state-of-the-art digital models. Why this difference? Is the digital sota model not implementable with the SLiM?
- The theoretical model introduced to justify linear error accumulation across layers (rather than exponential) is good but lacks direct empirical validation. It would be useful to show variance or error growth as a function of depth from actual hardware measurements

(Remarks on code availability)
The paper does not appear to be accompanied by publicly available code, nor does it reference any repository containing implementation details, model weights, or control software used in the experiments. As such, I was not able to assess the usability or reproducibility of the code.

Without access to this material, the work remains difficult to replicate and reuse by the broader community, particularly for researchers interested in benchmarking or building upon the SLiM architecture. Making this codebase available would

significantly increase the impact and credibility of the results.

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
The reviewer appreciates the authors' explanation and response.

The experimental implementation based on TDM technology and chip reusability is well presented. Although the engineering workload is substantial, the approach appears reasonable for demonstrating SLiM's effectiveness.

However, several critical issues remain:
1. The working principle of the SLiM chip requires further clarification, particularly regarding:
1.1 The physical implementation of Y=WX in the chip architecture
1.2 The encoding mechanism for input X
1.3 The programming mechanism for the weight matrix W
1.4 The process of obtaining output Y
1.5 The error tolerance mechanism and whether it relies on O-E-O and A-D-A conversions between layers
It is strongly recommended to incorporate SI Figure 14 (chip structure diagram) and Supplementary Note 2 into the main manuscript, supported by clear explanations and relevant equations. Some descriptions create confusion. For instance, Line 125 states, "as shown in Figure 1b...the inputs x_i are assigned to specific wavelengths lambda_i." While this suggests N different wavelengths are individually modulated to encode x_i, SI Figure 14 shows only one input port. The mechanism enabling individual modulation of different wavelengths needs clarification.

2. The energy consumption claims appear optimistic. The current calculations exclude essential peripheral components (drivers, TIAs) and the energy overhead from necessary O-E-O conversions between layers, as the SLiM chip only performs matrix-vector multiplications.

(Remarks on code availability)

Reviewer #3

(Remarks to the Author)
All my concerns have been addressed.

Please, carefully proofread the paper once again.

(Remarks on code availability)

Version 2:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
All my concerns have been addressed.

(Remarks on code availability)
We have no more comment.

Response to referees:

We sincerely thank all the reviewers for their constructive and insightful comments. In this response letter, we address each remark point by point. Reviewer comments are highlighted in **blue**, and corresponding changes to the manuscript are marked in red. References specific to this response are denoted as Ref. [rX].

Summary of the main revisions:

1. We have included the analysis of the system efficiency of the SLiM chip. The revisions can be found in the [Main text], [Supplementary Note 4. Computing capabilities of single-layer photonic chip], and [Supplementary Table S1. Comparisons of deep AI experiments].

2. We have provided a detailed circuit diagram of the SLiM chip. Revisions are reflected in [Supplementary Fig. S14] and [Methods Section. Implementation of single-layer chips.].

3. We have provided additional explanations on the noise and errors, and have supplemented the validation of the error model. The corresponding revisions appear in [Supplementary Note 1. The error model of ONN.] and [Supplementary Figure S15].

4. We have discussed the implementation of parallel processing with the SLiM cluster. This discussion is included in [Supplementary Figure S16].

5. We have added further details regarding the SLiM matrix design and realization, as well as the training procedure for the SLiM network. The entire manuscript has also been thoroughly proofread and revised for clarity and precision.

6. We have open-sourced the SLiM model and data for the community.

The detailed, point-by-point responses are provided below.

****

Reviewer #1 (Remarks to the Author):

This work introduces perturbations on-chip to decouple computational correlations to eliminate the redundancies and develop deep photonic learning with a single-layer photonic computing (SLiM) chip. The SLiM chip overcomes the depth limitations of optical neural networks, which allows error rates constrained across over 200 layers, and extends spatial depth from millimeter to hundred-meter scale, which enables the 3-D chip clusters. The authors experimentally realized a neural network with 100 layers for image classification, a 0.345 billion-parameter LLM model with 384 layers for text generation, and a 0.192 billion-parameter LLM model with 640 layers for image generation.

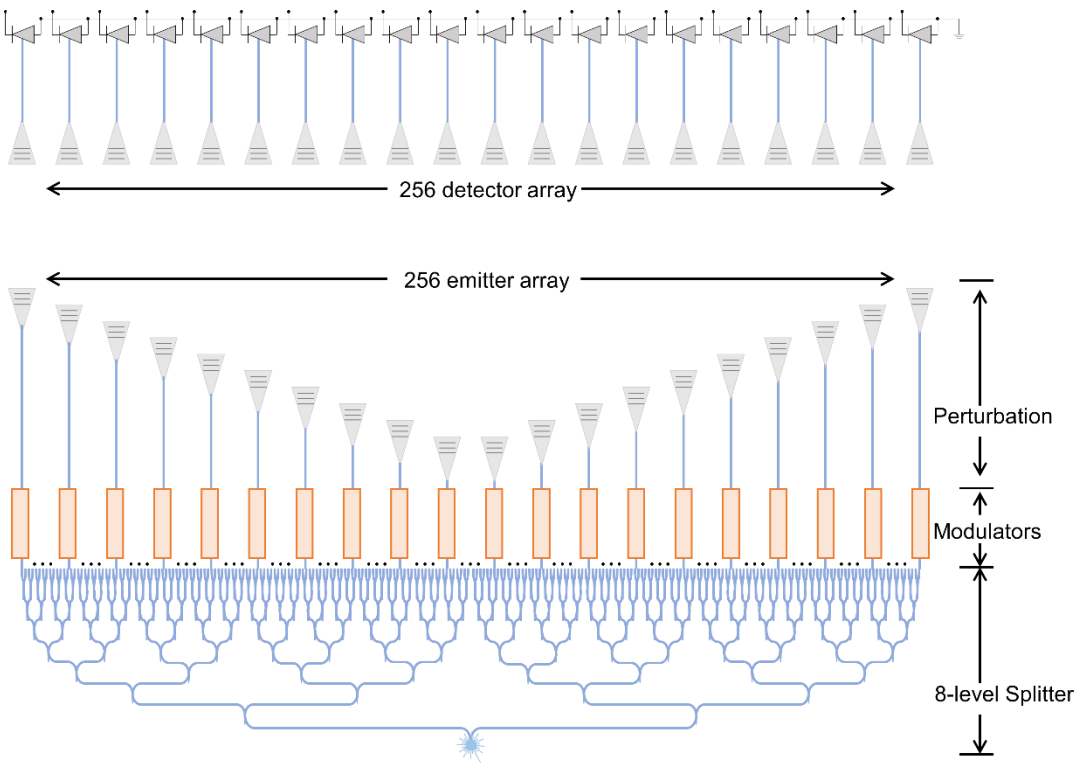
The perturbation-introduction approach is novel and attractive, and the experimental results are also impressive. However, the reviewers have some questions and requests for explanation.

Response: We sincerely thank the reviewer for recognizing our innovations. Please find our point-by-point responses to the comments below.

1.The structure of the SLiM chip is not illustrative. A structure figure is recommended.

Response: We thank the reviewer for the suggestions. We have provided a more detailed structure illustration in Fig. S14. As we described in the Methods section, the input is coupled onto the chip via a single-mode fiber, which is then uniformly distributed through an 8-level splitter tree. Each of the resulting 256 channels is independently modulated, followed by a perturbation region where the optical path length is varied by 20 μm. On the detection path, light can be coupled to the detector either through a 256-element grating array or sensed directly via free-space propagation.

Revision: We have added the detailed circuit diagram of the SLiM chip in Fig. S14 as below. We have also modified the description of the SLiM chip implementation in the Methods section as …The circuit diagram of the SLiM chip is displayed in Fig. S14.



Supplementary Figure S14. Circuit diagram of the SLiM circuit.

2.Why the perturbations incorporated within the photonic chip can eliminate the redundancies? What do the redundancies refer to? Where do the redundancies come from?

Response: We thank the reviewer for the comments. Please find our detailed responses below.

Q: What do the redundancies refer to?

Response: The redundancies refer to the unnecessary layers on photonic integrated circuits in realizing matrix operations. In conventional photonic computing architectures, the layer number is in direct proportion to the size of the target matrix. For example, to realize an M×N matrix, an L-layer photonic circuit is required, where L~O(N). Here we discovered that the (L-1) layers from layer-2 to layer-N are actually redundant: matrix could be realized with single layer propagation. These extra (L-1) layers amplify the processing errors.

Q: Where do the redundancies come from?

Response: The redundancies come from the correlation of the on-chip wavelength and spatial channels, because of which it is hard to realize arbitrary matrices with single layer circuits through existing methods. Specifically, to realize the computing matrix W, the transmission channels are modulated on chip by exerting wavefront Φ, derived from the equation $W = G\Phi$. However, because of the spectral proximity, the wavefront Φ and the transmission matrix G both exhibit high correlation, resulting in linear dependence on the right-hand side of the equation that does not guarantee a solution.

Q: Why the perturbations incorporated within the photonic chip can eliminate the redundancies?

Response: By incorporating the perturbations on chip, diverse wavelength channels are decorrelated, enabling arbitrary matrices to be realized with minimal error propagations. We introduce on-chip perturbative noise to break the correlation, mathematically represented as $\widehat{\Phi}_k = \Phi_k \cdot e^{jn2\pi\Delta k/\lambda_i}$, where k symbolizes the waveguide coordinate with corresponding noise Δk. The linearly correlated equations can be reformulated as $W = G\widehat{\Phi}$, which is independent set and guarantees a solution for matrix W with single-layer propagation.

3.How does the single-layer photonic chip minimize the error propagation? Does the "single-layer chip" mean that the photonic chip can process one layer of a LLM? Then, to build a 100-layer network, do the authors utilize 100 single-layer photonic chips? In one layer of LLM, there are many multiplications between vectors and matrices. The SLiM chip appears to complete only one MVM at one computing cycle. How does one SLiM chip correspond to the operations in the LLM?

Q: How does the single-layer photonic chip minimize the error propagation?

Response: We thank the reviewer for the comments. By reducing the layers from L layer to single layer, the error accumulation is minimized. As the output error is in proportion to the depth of layers $Var(\Delta Y/Y) \sim \sum_{i=1}^{L}(Var(\Delta W_i/W_i))$, The single layer chip reduces the errors by L times, through decreasing the layer number from L to 1. To illustrate this point, we employ a simulation validation: we randomly draw 10 samples of 10×10 matrices, with each entry drawn from gaussian distribution. During the validation, each layer is distorted with gaussian noise with 0.1 standard deviation. The mean squared error are calculated and averaged among 10,000 trials, which is visualized below, which shows the error accumulation is linear, and the single layer propagation has the least error.

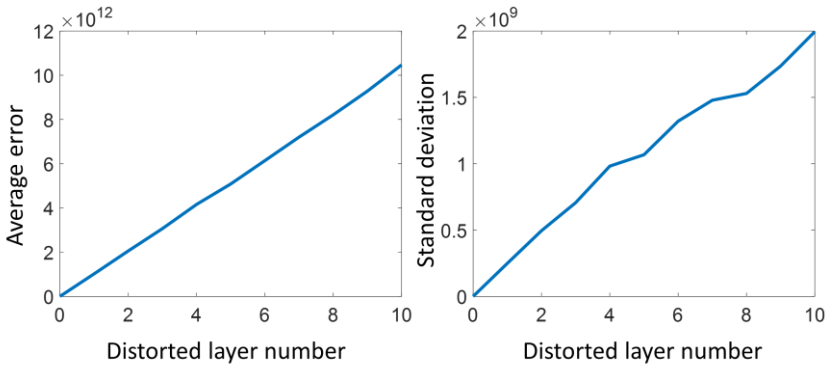
Q: Does the "single-layer chip" mean that the photonic chip can process one layer of a LLM? Then, to build a 100-layer network, do the authors utilize 100 single-layer photonic chips? In one layer of LLM, there are many multiplications between vectors and matrices. The SLiM chip appears to complete only one MVM at one computing cycle. How does one SLiM chip correspond to the operations in the LLM?

Response: Although a deep neural network may contain numerous layers, its computations can be decomposed into a series of basic matrix operations. These operations can be flexibly mapped onto SLiM chips without requiring a one-to-one correspondence between layers and chips. Each SLiM chip is capable of performing a matrix-vector multiplication. Therefore, even a model with 100 layers does not necessarily require 100 individual SLiM chips for implementation. As described in the Methods section [Deep neural network with SLiM chips], our large language model implementation partitions the overall computation into submatrices, with each submatrix handled by a single SLiM chip. In our experiments, we demonstrate this by reusing a single packaged SLiM chip to sequentially implement the entire model, validating the chip's capability to support deep architectures through time-multiplexed execution.

Revision:

We have updated the related discussion and figure analysis in the supplementary note 1, as,

We numerically validated the error trend by employing 10 samples of 10×10 matrices, with each entry randomly sampled from gaussian distribution. During the validation, a 10×1 random vector is sequentially multiplied with these 10 matrices, the computation is gradually distorted with the noise. For example, to analyze the errors with k layers, the first k layers are distorted with the noise. Finally, the errors are calculated for these 10 consecutive matrix multiplications. This procedure was repeated 10,000 times, and the mean and standard deviation of the errors were computed. As shown in Fig. S15, the mean error increases linearly with the number of noisy layers, corroborating our theoretical analysis.



Supplementary Figure S15. Analysis of the error accumulation with respect to layers. Left panel: average error, right panel: standard deviation.

We have discussed the parallelization method in Fig. S16.

4.The authors claim that W_i = |G_i *Phi_i|^2 in SI-2.1. When no perturbation, G_i~=G_0, Phi_i~=Phi_0, then W_i=W_0. If the perturbation is introduced, Phi_i != Phi_0, W_i are different. At the same time, the authors claim that the phase distribution appears random. Therefore, the W are random. How can a random W process the information effectively?

Response：We thank the reviewer for the comments. Even through the phase distribution is random, the W is not random because (1) the random perturbation remains fixed after chip fabrication as a random initialization from the view of neural network training; (2) the phase term Φ_{ik} can then be designed for the realization of target matrices. As we showed in the Fig. 2d (we replicate below), by designing the values of Φ_{ik}, matrices of diverse dimensions can be realized.

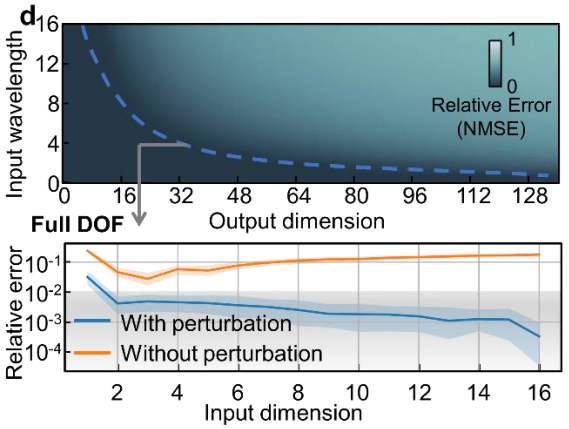


Fig. 2d. Matrices with arbitrary dimension and full degree-of-freedom (DOF). The relative error is quantitatively evaluated by sweeping the input dimensions as well as the output dimensions, where the error with full DOF as delineated with dashed lines, were plotted against the method without redundancy elimination. With redundancy elimination, the relative error dropped from ~10% to ~0.1%.

5.How is the propagation perturbation injected? Since the propagation perturbation can decorrelate the weights, does it mean that propagation perturbation can be trained and programmed? Is the training completed online or offline? If online, how is it realized? If offline, will the offline training be complicated and cost too much offline computational resources?

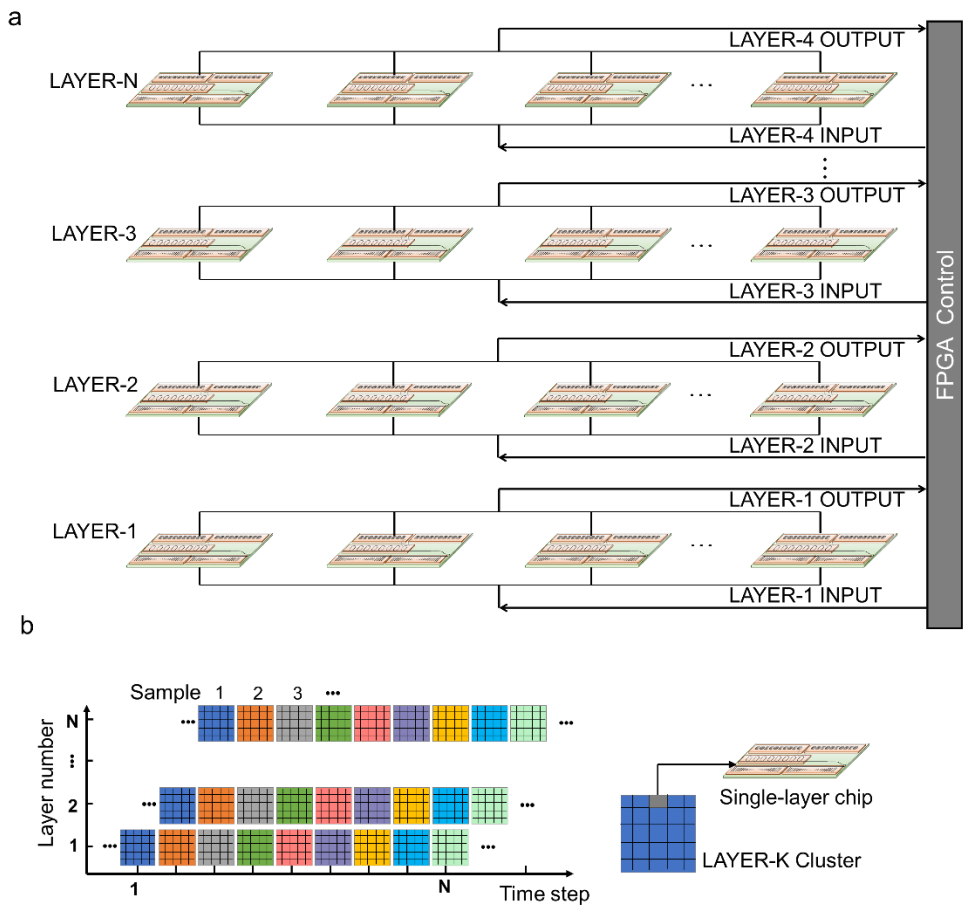
Response: We thank the reviewer for the questions. The perturbation is injected by tuning the waveguide lengths in the chip fabrication process. As we described in the main text [Error tolerance with single-layer photonic computing] and [Supplementary Note 2. Formulations of the single-layer photonic computing], the propagation lengths of the k-th on-chip emitter are selected as $k\Delta d$ and Δd is optimized to be 20 μm.

In the experiments, the phase modulation of the channels is trained offline with experimentally calibrated characteristics as shown in Fig. S13, which combined with the perturbation, helps realizing arbitrary matrices. In the experiment, each matrix design takes 0.167s on 3090ti GPU.

Revision: We have provided the training time of matrix on the GPU in the [Methods section Deep neural network with SLiM chips.]: The phase terms for realizing each matrix are realized by trained offline, taking 0.167 second on one 3090 Ti GPU.

6.Besides the SLiM chip configuration and perturbation injection, many operations are processed in the electronic domain, like the nonlinear activation functions and the data transfer among the SLiM chips. During the experiments, how do the authors control the total computing cluster to process billions of parameters of the LLM and complete the LLM computing? How is the coordination of multiple SLiM chips managed for processing billion-parameter LLMs?

Response: We thank the reviewer for the questions. In the experiment, we reuse one packaged SLiM chip for the implementation of the matrix vector multiplication, and the output is directed to the dedicated control board for the activation. Before the experiments, the computations are first decomposed into matrix multiplications with an offline computer. To parallelize the process and further enhanced the speed, computations could be amortized into the SLiM chip cluster. Computations in one layer of the SLiM network can be distributed across a chip cluster and processed concurrently, while N cluster facilitates the N layer network. The N cluster can process N inputs in a streamline mode. We supplement the discussion for parallelism in supplementary Fig. S16 as presented below.



Supplementary Figure S16. Parallel processing with SLiM chip cluster. a, The N layer neural network could be realized by N SLiM cluster. Each SLiM chip in one cluster parallelly implements the computations of one layer. The chips are scheduled by a controller that allocates the input data and collect the outputs. b, The streamline of the cluster processing, the N cluster can process N samples concurrently.

7.Regarding the computing power, the 327.68 TOPS is clear, but where does the 335.54 Peta OPS come from?

Response: We thank the reviewer for the questions. For 335.54 Peta OPS metrics are results of the 16 chip ensemble as displayed in Fig. 3 (we duplicate below). As described in [Supplementary Note 4. Computing capabilities of single-layer photonic chip], the total number of operations sums to $2K_mK_d$. On the 16-chip ensemble, $K_m = K_d = 256 \times 16 = 4096$, which can reach 33,554,432 operations and 335.54 Peta OPS at 10 GHz speed.

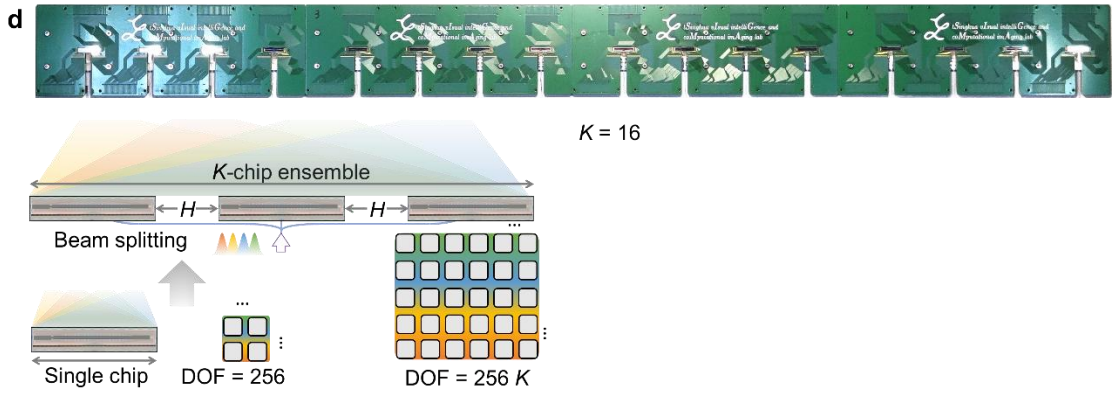


Fig. 3 d, The DOF can be expanded from 256 up to 256K by utilizing spatial ensemble techniques. The top panel displays the ensemble configuration of K=16.

8.The computing energy 1510.4Peta OPS/W appears to only include the energy cost of the phase shifters. However, the authors claim that the SLiM achieves ExaOPS/W (10^6 TOPS/W) efficiency, which is unacceptable. The authors should assess the energy efficiency more accurately. The energy cost should include energy cost from the phase shifters, weight matrix programming, and detectors, as well as the energy cost to complete the nonlinear activation computing. Otherwise, the energy comparison between the SLiM chips and GPUs which include complete MVM computing is unfair.

Response: We acknowledge the insightful question from the reviewer.

We have provided the system energy efficiency calculations to include the other components necessary for the implementation, such as the Digital-to-Analog Converters (DACs), Analog-to-Digital Converters (ADCs), optical power sources, and detectors. The SLiM computing employs 1-bit data conversions in high-speed experiments, which simplifies the circuit design and reduces energy consumption compared to higher-bit converters. With current technologies, the power dissipation of 1-bit DAC driving corresponds to the capacitance charge of the modulators, which consumes 6.3 fJ/bit [r1]. The ADC process could be simplified to a low-power comparator, which draws 27.3fJ/bit [r2]. Accordingly, the average power consumptions per channel for each component are as follows: P_{SLiM} = 1.7 μW, P_{DAC} = 63 μW, P_{opt} = 200 μW, P_{DET} = 250 μW, and P_{ADC} = 273 μW.

Denoting the input channel number, output channel number, and working frequency as N, M, and F (10-GHz), the overall system efficiency is calculated as $\eta = 2MNF/\left(N\left(P_{DAC} + P_{opt}\right) + M\left(P_{ADC} + P_{DET}\right) + MNP_{SLiM}\right)$. Our findings indicate that the system efficiency reaches 3.25×10^3 TOPS/W (as in the high-speed experiments), while considering the considering the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs).

[r1] Yuan, Y. *et al.* A 5× 200 Gbps microring modulator silicon chip empowered by two-segment Z-shape junctions. *Nature Communications* **15**, 918 (2024).

[r2] Wang, J., Ye, F. & Ren, J. in *2021 IEEE 14th International Conference on ASIC (ASICON).* 1-4 (IEEE).

Revision:

We have included the revised content in the main and supplementary information.

Supplementary Note 4:

…We also included the other components of the system including the Digital-to-Analog Converters (DACs), Analog-to-Digital Converters (ADCs), optical power sources, detectors, and calculate system energy efficiency. The SLiM chip employs 1-bit converters in high-speed experiments, which simplifies the circuit design and reduces energy consumption compared to higher-bit converters. With current technologies, the power dissipation of 1-bit DAC driving corresponds to the capacitance charge of the modulators, which consumes 6.3 fJ/bit [7]. The ADC process could be simplified to a low-power comparator, which draws 27.3fJ/bit [8]. The specific power consumptions per channel for each component are as follows: P_{SLiM} = 1.7 μW, P_{DAC} = 63 μW, P_{opt} = 200 μW, P_{DET} = 250 μW, and P_{ADC} = 273 μW. Denoting the input channel number, output channel number, and working frequency as N, M, and F (10-GHz), the overall system efficiency is calculated as $\eta = 2MNF/\left(N\left(P_{DAC}+P_{opt}\right)+M\left(P_{ADC}+P_{DET}\right)+MNP_{SLiM}\right)$. We have plotted the energy efficiency against the degrees of freedom (DOFs) in Fig. S14. Our findings indicate that in the high-speed experiments, the system efficiency reaches 3.25×10^3 TOPS/W, while considering the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs)….

7 Yuan, Y. *et al.* A 5× 200 Gbps microring modulator silicon chip empowered by two-segment Z-shape junctions. *Nature Communications* **15**, 918 (2024).

8 Wang, J., Ye, F. & Ren, J. in *2021 IEEE 14th International Conference on ASIC (ASICON).* 1-4 (IEEE).

We have also included the discussion of system energy efficiency in the main text and have updated the supplementary table 1.

Methods	Task	Depth	Cycle time	Energy efficiency
SLiM chip (Optoelectronics)	Image Classification Acc. 85.4% (ImageNet 1000 Classes)	**100**	**0.1 ns @10 GHz**	**3.25×10^3 TOPS/W[a]**
	Text generation Loss function value 3.0397 (Wikipedia text)	**384 (96×4)**		
	Image Generation Loss function value 7.321 (ImageNet dataset)	**640 (64×10)**		

a. Calculation includes the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs).

In summary, the work is impressive and introduces novel methods to overcome the depth limitations. However, the details of the SLiM chip and the experiments should be described.

Response: We sincerely thank the reviewer for the constructive comments. We hope that with point-by-point response, the details of the work could be better understood.

****

Reviewer #2 (Remarks to the Author):

- What are the noteworthy results?
The results are very noteworthy, the scale of the transmitter and SLiM chips are very impressive with respect to complexity and scale. Additionally, the packaging, control, and extension to multiple chiplets is state-of-the-art. The demonstrated architecture is novel. Congratulations to the authors for the sucessful results.

-Will the work be of significance to the field and related fields? How does it compare to the established literature? If the work is not original, please provide relevant references.
This work will be of significance within the field of neuromorphic photonics. The performance with respect to depth and scale is equal or greater than other approaches within the field.

-Does the work support the conclusions and claims, or is additional evidence needed?
The claims are very significant and as impressive as this experiment is. I think additional evidence would be useful to back up the claim. Additionally, the flow of the writing and logical conclusions are at time difficult to follow.

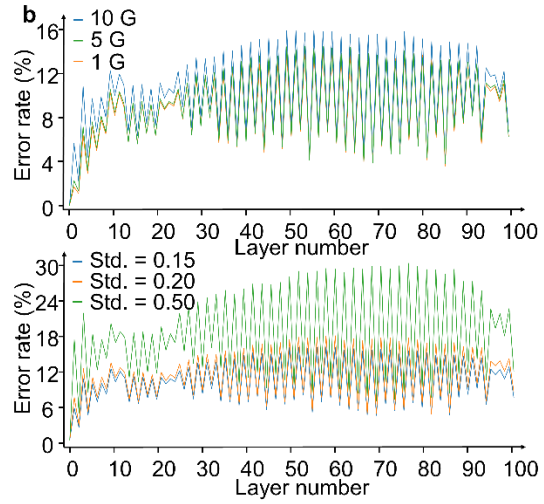
Response: we sincerely acknowledge the reviewer for the recognition of our innovations. We have thoroughly proofread the manuscripts to improve the readability. The comments are replied point-by-point below.

I think it would be helpful to have more clearly presented performance metrics with respect to RF noise figure and loss per layer. With the addition of noise figure analysis, one could understand the required signal-to-noise of the input signal as well as the scalability of this system without error correction. If the photonic loss and digital loss are 3.04 and 2.96, respectively then only a few layers would be possible. The claim of deep learning within neuromophic photonic approaches is the key limiting factor and claiming to overcome this goal is a significant claim.

Response: We sincerely thank the reviewer for the comments. We apologize for the ambiguity caused here. The figures 3.04 and 2.96 are the loss function values realized in the SLiM network, which is a quantification of the difference between the system outputs and the target in the dataset, rather than the noise in the system.

In response to the concerns on the noise, we calibrated the system noise in Fig. S3b (we duplicated below), which shows the error rates associated to the experiments carried out using the

100-layer error-tolerant model. The upper section showcases results from tests conducted at frequencies of 1 G, 5 G, and 10 G, yielding error rates of 9.04%, 9.22%, and 10.34%, respectively. The lower section details the outcomes under conditions of Gaussian noise, where standard deviations of 0.15, 0.20, and 0.50 corresponded to error rates of 10.48%, 11.56%, and 18.78%. Accordingly, the noise figure of the system is equivalent to the 0.15 standard deviation noise under the 10 GHz experiment scenario.



b, The error rates associated to the experiments carried out using the 100-layer error-tolerant model. The upper section showcases results from tests conducted at frequencies of 1 G, 5 G, and 10 G, yielding error rates of 9.04%, 9.22%, and 10.34%, respectively. The lower section details the outcomes under conditions of Gaussian noise, where standard deviations of 0.15, 0.20, and 0.50 corresponded to error rates of 10.48%, 11.56%, and 18.78%.

Revision:

To avoid misunderstanding, we have updated the expression of the loss, to loss function value, in supplementary table S1, when ambiguity may arise.

-Are there any flaws in the data analysis, interpretation and conclusions? Do these prohibit publication or require revision?
Not identified but also I am not confident with respect to the Machine Learning Data Analysis. It would be helpful to see more optical / electronic analysis with respect to noise and loss.

Response: We thank the reviewer for the suggestion. We will illustrate the noise and error rates presented throughout the manuscript.

The 100-layer image classification:

As described in the response 2.2, the experimental noise is benchmarked with the 100-layer network, and calibrated to be equivalent to gaussian noise of 0.15 standard deviation, with which the error rate is controlled to below 10.34%. We also calibrate the maximal noise tolerated of the architecture in Fig. 4c (duplicated below) as ~20%-std gaussian noise, when the performance deteriorates obviously beyond this error regime.

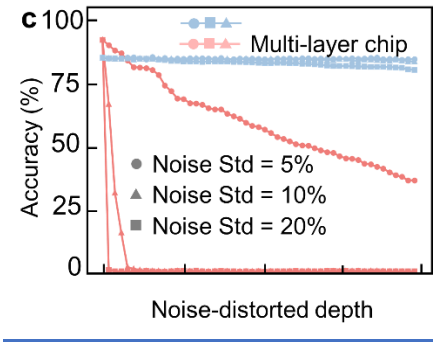


Fig. 4. c, Analysis was conducted for the deep ONNs involved in classifying the ImageNet-1000 dataset. The networks were tested against various levels of Gaussian noise featuring different standard deviations (Stds), focusing on both the ONNs with single-layer and the multi-layer chip implementation as noise gradually distorted the layers.

The Billion-parameter ONN:

We present in Fig. 5d, the error rate controlled throughout the 640 layers. Under the experiment noise, the error rates maintained within 37.2%/38.6%/40.2%/38.8%/42.1%/39.1% for six-category image generation.

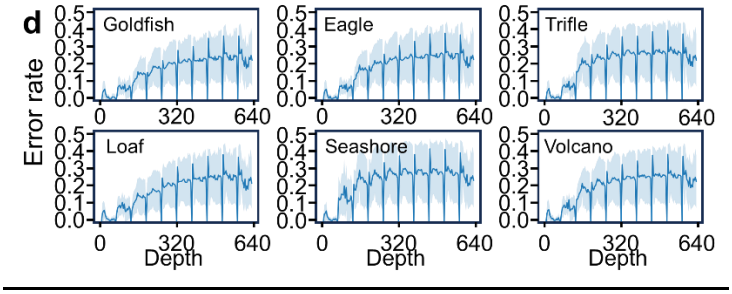


Fig. 5. d, The bounded error rate across 640 layers during a 10-step image generation process.

Theoretically proof:

The error model of ONN is presented in [Supplementary Note 1. The error model of ONN.], where we show that the error increase at least linearly. To corroborate the theoretical analysis, we supplement the validation of the error trend by employing 10 samples of 10×10 matrices, with each entry randomly sampled from gaussian distribution. During the validation, a 10×1 random vector is sequentially multiplied with these 10 matrices, the computation is gradually distorted with the noise. For example, to analyze the errors with k layers, the first k layers are distorted with the noise. Finally,

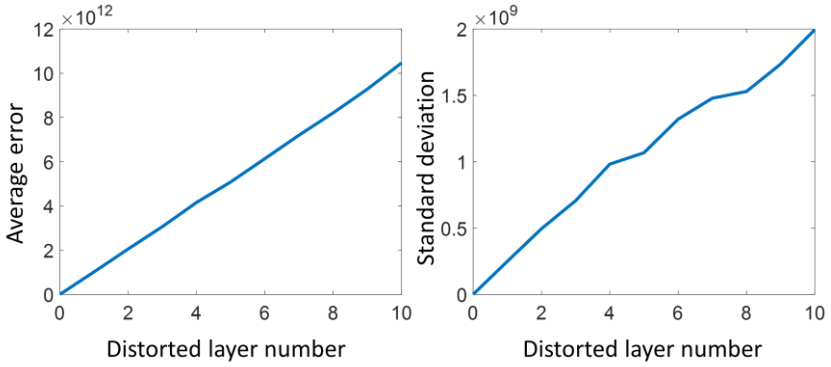
the errors are calculated for these 10 trials. The above process is repeated for 10000 times and averages, the mean and standard deviation of the errors are plotted as Fig. S15, which shows a linear scale of the mean errors.

The error bounding is presented in Fig. S15 as [Supplementary Note 3. Error-bounding with single-layer photonic computing.], where we prove that the error rate can be controlled below 50%, which is validated by the experiment results of 100-layer image classification and The Billion-parameter ONN.

Revision:

We have updated the related discussion and figure analysis in the supplementary note 1, as,

We numerically validated the error trend by employing 10 samples of 10×10 matrices, with each entry randomly sampled from gaussian distribution. During the validation, a 10×1 random vector is sequentially multiplied with these 10 matrices, the computation is gradually distorted with the noise. For example, to analyze the errors with k layers, the first k layers are distorted with the noise. Finally, the errors are calculated for these 10 consecutive matrix multiplications. This procedure was repeated 10,000 times, and the mean and standard deviation of the errors were computed. As shown in Fig. S15, the mean error increases linearly with the number of noisy layers, corroborating our theoretical analysis.



Supplementary Figure S15. Analysis of the error accumulation with respect to layers. Left panel: average error, right panel: standard deviation.

-Is the methodology sound? Does the work meet the expected standards in your field?
The methodology is sound and meet standards.

-Is there enough detail provided in the methods for the work to be reproduced?
I believe so, but more information regarding how the nonlinearity is achieved and the transfer function would be useful.

Response: We thank the reviewer for the suggestion. As is described in the section [Error tolerance with single-layer photonic computing], the nonlinear activation function has the form of $f = \int \delta(x - \Delta f)$, where Δf represents the perturbation position in the nonlinear function. This activation function can be implemented at the detection with the analog-to-digital conversion $(\cdot)\_D$,

by assigning the error-distorted values to the nearest signal levels. In this implementation of the high-speed experiments (Fig. 4 and Fig. 5), the outputs are captured and activated by feeding the result to the dedicated control board, and the distribution of the perturbation position Δf are visualized in the Fig. 4b, Fig. S4, and Fig. S5.

Revision:

We have updated the implementation of the nonlinearity in the Methods section as:

…The outputs are captured and activated by feeding the result to the dedicated control board.. …

Additional Comments:

The experiment presented within this paper is tremendously impressive but the claims made are very significant. I think more information regarding loss per layer, nonlinearity, and noise performance is required. Some of the sections are difficult to comprehend and the writing should be improved before publication.

Response: We thank the reviewer for the suggestion. We have presented the loss and noise analysis in the response 2.2&2.3. We have described the nonlinear realization in response 2.4. We have thoroughly proofread the manuscripts to improve the readability.

I believe the introduction should be expanded to focus a bit on the history of photonic neural networks and the need for this approach. There are very condensed statements made by the authors about noise, physical analog computing, but there is no discussion of other research within the field which has focused on tackling these limitations to scaling and analysis of scaling photonic neural networks. I would recommend the authors look into literature on silicon photonic neural networks for analog machine learning approaches, e.g. RF fingerprinting or automatic modulation classification

Response: We thank the reviewer for the suggestion. We provided more discussion on silicon photonic neural networks in the introduction.

Revision:

We have modified the introduction part as:

…Notably, with the ongoing evolution of photonic integration, the silicon photonic platform has emerged as the most mature and scalable solution for mapping neural networks onto hardware for intelligent signal processing29,34,39-41. Its large bandwidth and compatibility with CMOS processes make it particularly well-suited as an analog machine learning module for high-speed signal processing tasks, especially in the radio frequency (RF) domain42,43….

29 Tait, A. N. et al. Neuromorphic photonic networks using silicon photonic weight banks. Scientific reports 7, 7430 (2017).

34 Shen, Y. et al. Deep learning with coherent nanophotonic circuits. Nature Photonics 11, 441 (2017).

39 Bogaerts, W. et al. Programmable photonic circuits. Nature 586, 207-216 (2020).

40 Jalali, B. & Fathpour, S. Silicon photonics. Journal of lightwave technology 24, 4600-4615 (2006).

41 Reed, G. T. Silicon photonics: the state of the art. (2008).

42 Deng, H. et al. Single-chip silicon photonic engine for analog optical and microwave signals processing. Nature Communications 16, 5087 (2025).

43 Pérez-López, D. et al. General-purpose programmable photonic processor for advanced radiofrequency applications. Nature Communications 15, 1563 (2024).

****

Reviewer #3 (Remarks to the Author):

The authors present a single-layer photonic computing chip that by injecting purposely designed perturbations in both the linear transfer matrix and the detection non-linearity suppresses the classical error-accumulation bottleneck of analog optical neural networks.

The paper is well written, well structured and well documented.

Response: We sincerely thank the reviewer for recognizing our innovations. Please find our point-by-point responses to the comments below.

I have some minor concerns:

- Authors state that energy measures rely on resonant phase shifters and excludes laser, ADC/DAC, CMOS drivers and control FPGA. Supplementary Note 4 states these are "additional overhead" yet does not quantify them.

Response: We sincerely thank the reviewer for the comments.

We have provided the system energy efficiency calculations to include the other components necessary for the implementation, such as the Digital-to-Analog Converters (DACs), Analog-to-Digital Converters (ADCs), optical power sources, and detectors. The SLiM computing employs 1-bit data conversions in high-speed experiments, which simplifies the circuit design and reduces energy consumption compared to higher-bit converters. With current technologies, the power dissipation of 1-bit DAC driving corresponds to the capacitance charge of the modulators, which consumes 6.3 fJ/bit [r1]. The ADC process could be simplified to a low-power comparator, which draws 27.3fJ/bit [r2]. Accordingly, the average power consumptions per channel for each component are as follows: P_{SLiM} = 1.7 μW, P_{DAC} = 63 μW, P_{opt} = 200 μW, P_{DET} = 250 μW, and P_{ADC} = 273 μW.

Denoting the input channel number, output channel number, and working frequency as N, M, and F (10-GHz), the overall system efficiency is calculated as $\eta = 2MNF/\left(N\left(P_{DAC}+P_{opt}\right)+M\left(P_{ADC}+P_{DET}\right)+MNP_{SLiM}\right)$. Our findings indicate that the system efficiency reaches 3.25×10^3 TOPS/W (as in the high-speed experiments), while considering the considering the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs).

[r1] Yuan, Y. *et al.* A 5× 200 Gbps microring modulator silicon chip empowered by two-segment Z-shape junctions. *Nature Communications* **15**, 918 (2024).

[r2] Wang, J., Ye, F. & Ren, J. in *2021 IEEE 14th International Conference on ASIC (ASICON).* 1-4 (IEEE).

Revision:

We have included the revised content in the main and supplementary information.

Supplementary Note 4:

…We also included the other components of the system including the Digital-to-Analog

Converters (DACs), Analog-to-Digital Converters (ADCs), optical power sources, detectors, and calculate system energy efficiency. The SLiM chip employs 1-bit converters in high-speed experiments, which simplifies the circuit design and reduces energy consumption compared to higher-bit converters. With current technologies, the power dissipation of 1-bit DAC driving corresponds to the capacitance charge of the modulators, which consumes 6.3 fJ/bit [7]. The ADC process could be simplified to a low-power comparator, which draws 27.3fJ/bit [8]. The specific power consumptions per channel for each component are as follows: P_{SLiM} = 1.7 μW, P_{DAC} = 63 μW, P_{opt} = 200 μW, P_{DET} = 250 μW, and P_{ADC} = 273 μW. Denoting the input channel number, output channel number, and working frequency as N, M, and F (10-GHz), the overall system efficiency is calculated as $\eta = 2MNF/\left(N\left(P_{DAC}+P_{opt}\right)+M\left(P_{ADC}+P_{DET}\right)+MNP_{SLiM}\right)$. We have plotted the energy efficiency against the degrees of freedom (DOFs) in Fig. S14. Our findings indicate that in the high-speed experiments, the system efficiency reaches 3.25×10^3 TOPS/W, while considering the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs)….

7 Yuan, Y. *et al.* A 5× 200 Gbps microring modulator silicon chip empowered by two-segment Z-shape junctions. *Nature Communications* **15**, 918 (2024).

8 Wang, J., Ye, F. & Ren, J. in *2021 IEEE 14th International Conference on ASIC (ASICON).* 1-4 (IEEE).

We have also included the discussion of system energy efficiency in the main text and have updated the supplementary table 1.

Methods	Task	Depth	Cycle time	Energy efficiency
SLiM chip (Optoelectronics)	Image Classification Acc. 85.4% (ImageNet 1000 Classes)	**100**	**0.1 ns @10 GHz**	**3.25×10^3 TOPS/W[a]**
	Text generation Loss function value 3.0397 (Wikipedia text)	**384 (96×4)**		
	Image Generation Loss function value 7.321 (ImageNet dataset)	**640 (64×10)**		

a. Calculation includes the power consumption of modulators, detectors, optical power sources, Digital-to-Analog Converters (DACs), and Analog-to-Digital Converters (ADCs).

- Authors state to reach 85.2 % accuracy on ImageNet for 100-layer SLiM, far below 91% reached by state-of-the-art digital models. Why this difference? Is the digital sota model not implementable with the SLiM?

Response: We thank the reviewer for the comments. In the experiments, the SLiM network for imagenet-1000 classification tasks reached 85.4 %, while the purely digital simulation reached 93.0 % (as shown in section [Reaching ONNs beyond the depth limit for advanced artificial intelligence)). This is because the SLiM network employs 1-bit weights contains 11.475 MB parameter and thus has smaller model size than the digital model with 32-bit floating point weights of 220.32 MB

parameter. For fair comparison, we decrease the size parameter of digital model to 11.48MB by uniformly reducing the number of channels to (14, 28, 56, 115) in the blocks while maintaining the depth of network. The network accuracy converged to 79.19% accuracy after 90 epochs of training, which is below the 85.4% accuracy reached by SLiM network.

Revision:

We have updated the SLiM network training in the Methods section [Network architecture and training methods.] as:

…We also evaluated digital models of the same size to the 100-layer SLiM network. For both networks with 11.475 MegaByte parameter, the SLiM network reached 85.4% accuracy, while the network of same size with 32-bit floating point weights reached 79.19% accuracy…

- The theoretical model introduced to justify linear error accumulation across layers (rather than exponential) is good but lacks direct empirical validation. It would be useful to show variance or error growth as a function of depth from actual hardware measurements

Response: The theoretical model shows that $Var\left(\frac{\Delta A}{A}\right) \geq \sum_{i=1}^{L} Var\left(\frac{\Delta W_i}{W_i}\right)$, indicating that the accumulation of error in a multi-layer optical neural network scales at least linearly with the number of layers. Since our fabricated chip features a single-layer design, this architecture intrinsically minimizes cumulative error. To empirically validate this theoretical result, we conducted a simulation experiment in which 10 samples of 10×10 matrices were generated, with each element drawn from a standard Gaussian distribution. In each simulation trial, Gaussian noise with a standard deviation of 0.1 was injected into each matrix layer to simulate system errors. The resulting mean squared error (MSE) was computed after 10 consecutive matrix multiplications, and the results were averaged across all 10,000 trials. The simulation results, visualized below, confirm a clear linear scaling trend in error accumulation with increasing layer depth.

Revision:

We have updated the related discussion and figure analysis in the supplementary note 1, as,

We numerically validated the error trend by employing 10 samples of 10×10 matrices, with each entry randomly sampled from gaussian distribution. During the validation, a 10×1 random vector is sequentially multiplied with these 10 matrices, the computation is gradually distorted with the noise. For example, to analyze the errors with k layers, the first k layers are distorted with the noise. Finally, the errors are calculated for these 10 consecutive matrix multiplications. The above process is repeated for 10000 times and averages, the mean and standard deviation of the errors are plotted as Fig. S15, which shows a linear scale of the mean errors.

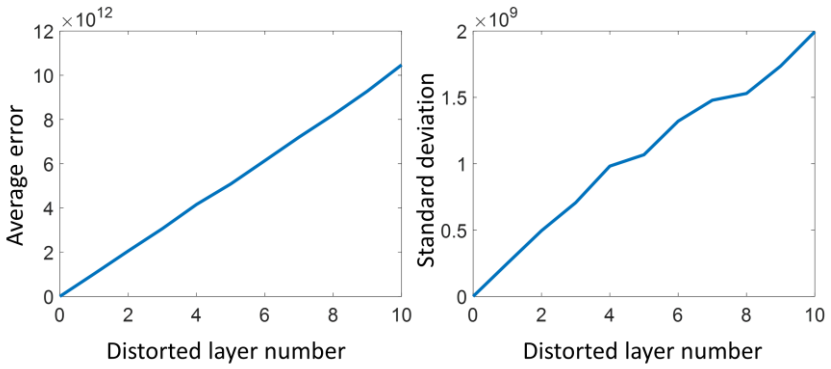


Fig. S15. Analysis of the error accumulation with respect to layers. Left panel: average error, right panel: standard deviation.

Reviewer #3 (Remarks on code availability):

The paper does not appear to be accompanied by publicly available code, nor does it reference any repository containing implementation details, model weights, or control software used in the experiments. As such, I was not able to assess the usability or reproducibility of the code.

Without access to this material, the work remains difficult to replicate and reuse by the broader community, particularly for researchers interested in benchmarking or building upon the SLiM architecture. Making this codebase available would significantly increase the impact and credibility of the results.

Response: We sincerely acknowledge the comments from the reviewer. We have uploaded the code and data to a publicly available repository: https://github.com/Yheechou/SLiM.

We sincerely acknowledge and follow the correction notes in the text associated file. Regarding the Low-power phase modulation, we have followed the suggestion of the reviewer, and the modulation efficiency is 0.753 π·cm corresponding to a modulation length of 43.5μm.

Revision:

We have updated the modulation efficiency as,

With a calibrated modulation efficiency of 1.7 μW/π **($V\pi L = 0.753$ V-cm,** as elaborated in [Methods Low-power phase modulation] and Fig. S6).

Response to referees:

We sincerely thank all the reviewers for their constructive and insightful comments. In this response letter, we address each remark point by point. Reviewer comments are highlighted in **blue**, and corresponding changes to the manuscript are marked in red. References specific to this response are denoted as Ref. [rX].

Summary of the main revisions:

1. We have provided a detailed working principle of the SLiM chip. Revisions are reflected in [Fig. 1 Single-layer photonic computing], and [Supplementary Fig. S14].

2. We have added further details regarding the SLiM matrix design and realization. Revisions are reflected in [Main text].

3. The entire manuscript has been thoroughly proofread and revised for clarity and precision.

4. We have provided the source data.

The detailed, point-by-point responses are provided below.

****

Reviewer #1 (Remarks to the Author):

The reviewer appreciates the authors' explanation and response. The experimental implementation based on TDM technology and chip reusability is well presented. Although the engineering workload is substantial, the approach appears reasonable for demonstrating SLiM's effectiveness. However, several critical issues remain.

Response: We sincerely thank the reviewer for recognizing our innovations. Please find our point-by-point responses to the comments below.

1. The working principle of the SLiM chip requires further clarification, particularly regarding:
1.1 The physical implementation of Y=WX in the chip architecture

Response: We thank the reviewer for the comments. The implementation of *Y=WX* is divided into the realization of *X* and the realization of *W*.

The implementation of *X*: as we illustrate in the main text, in single-layer photonic computing (SLiM) chip, the inputs x_i are assigned to specific wavelengths λ_i. In the experiment, the broadband light of multiple wavelengths is multiplexed into single channel and is fed into the input loading chip consisting of cascaded ring modulators (shown in Fig. 4a and detailed structures illustrated in Methods section) for individually configuring the input intensities to specific wavelength, after which the light encoding X is fed into the single-layer chip for processing. The entire pipeline is shown in Supplementary Fig. S1c.

The implementation of *W*: as we illustrated in the main text, to realize the computing matrix W, the transmission channels are modulated on chip by exerting wavefront Φ, derived from the equation $W = |G\Phi|^2$, with G the transmission matrix and Φ the wavefront profile. We introduce on-chip perturbative noise to break the correlation, mathematically represented as $\widehat{\Phi}_k = \Phi_k \cdot e^{jn2\pi\Delta k/\lambda_i}$, where k symbolizes the waveguide coordinate with corresponding noise Δk. The linearly correlated equations can be reformulated as $W = \left|G\widehat{\Phi}\right|^2$, which is independent set and guarantees a solution for matrix W with single-layer propagation. Accordingly, the single-layer chip performs the wavefront encoding $\widehat{\Phi}$ by exerting the on-chip modulation, while the propagation matrix G is realized with chip-to-chip propagation.

The implementation of *Y*: after chip-to-chip propagation, the detectors capture the output intensities, which is the computing results Y. The pipeline is shown in Fig. 1b.

Revision:

To improve the clearness of the implementation details of Y=WX. We have included the revised content in the main text.

Main manuscript:

…As illustrated in the Fig. 1b, in single-layer photonic computing (SLiM) chip, the inputs x_i

are assigned to specific wavelengths λ_i. Multi-wavelengths light is multiplexed into single fiber channel and coupled into an input loading chip for the configuration of inputs x_i onto corresponding light intensities. To realize the computing matrix W, the transmission channels are modulated on chip by exerting wavefront Φ, derived from the equation $W = |G\Phi|^2$, where G represents the chip-to-chip propagation.

1.2 The encoding mechanism for input X

Response: We thank the reviewer for the comments. Like we responded in 1.1, the input vector X is loaded onto the intensity of light, specifically, the broadband light of multiple wavelengths is multiplexed into single channel and is fed into the input loading chip consisting of cascaded ring modulators (shown in Fig. 4a and detailed structures illustrated in Methods section) for individually configuring the input intensities to specific wavelength, after which the light encoding X is fed into the single-layer chip for processing.

Revision:

We have included the revised content in the main text.

Main manuscript:

…As illustrated in the Fig. 1b, in single-layer photonic computing (SLiM) chip, the inputs x_i are assigned to specific wavelengths λ_i. Multi-wavelengths light is multiplexed into single fiber channel and coupled into an input loading chip for the configuration of inputs x_i onto corresponding light intensities.

1.3 The programming mechanism for the weight matrix W

Response: We thank the reviewer for the question. The weight matrix is formulated as $W = \left|G\widehat{\Phi}\right|^2$.

The programming includes three steps. First, the propagation matrix G is calibrated for specific propagation distances. Secondly, the modulation characteristics $\widehat{\Phi}$ is solved given the weight matrix W. Finally, the modulation coefficients is configured onto the single-layer chips.

Revision:

We have included the revised content in the main text.

Main manuscript:

…In realization of the target weight matrix W, the modulation wavefront Φ is first solved with calibrated propagation matrix G, after which the corresponding modulation coefficients are loaded onto the single-layer chips.

1.4 The process of obtaining output Y

Response： We thank the reviewer for the comments. Y is obtained with signal detection process after layer-to-layer propagation, performed using a germanium photodetector on the single-layer chip.

1.5 The error tolerance mechanism and whether it relies on O-E-O and A-D-A conversions between layers

Response: We thank the reviewer for the questions. The error tolerance is realized with the photodetection and signal conversion process, like we discussed in the main text: Note that this activation function can be implemented at the detection with the analog-to-digital conversion $(\cdot)_D$, by assigning the error-distorted values to the nearest signal levels.

It is strongly recommended to incorporate SI Figure 14 (chip structure diagram) and Supplementary Note 2 into the main manuscript, supported by clear explanations and relevant equations. Some descriptions create confusion. For instance, Line 125 states, "as shown in Figure 1b...the inputs x_i are assigned to specific wavelengths lambda_i." While this suggests N different wavelengths are individually modulated to encode x_i, SI Figure 14 shows only one input port. The mechanism enabling individual modulation of different wavelengths needs clarification.

Response: We thank the reviewer for the suggestion. For the better illustration of the implementation details. We have improved the SI Figure 14, and have incorporated SI Fig. S14 as well as the Supplementary Note 2 into the main text.

Regarding the multi-wavelength input, as illustrated in Fig. 1b, the multiwavelength light is first multiplexed into a single optical fiber before being coupled onto the single-layer chip. We have clarified the multiplexing in Fig. 1.

Revision:

We have included the revised content in the main text figure 1 and associated illustrations. We have also improved the Fig. S14 in the supplementary information.

Main manuscript:

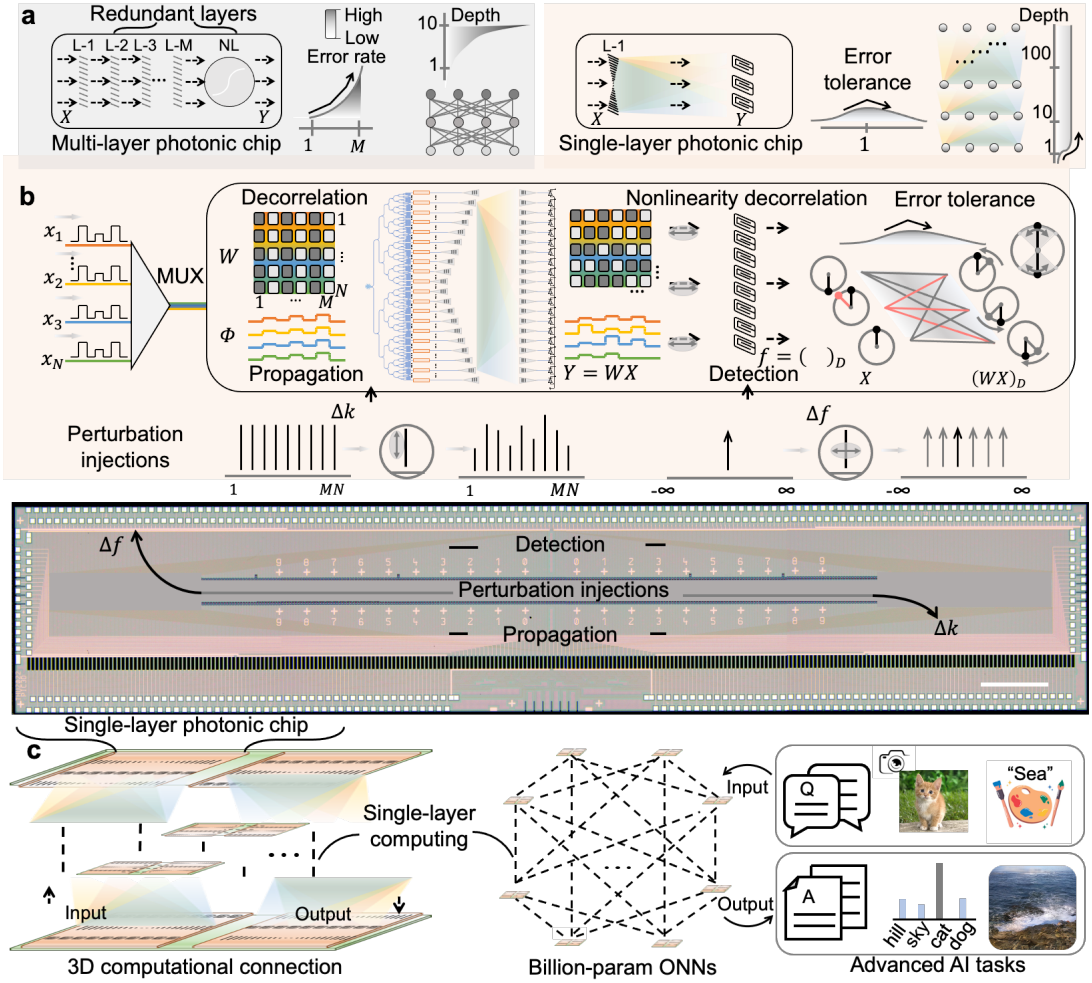


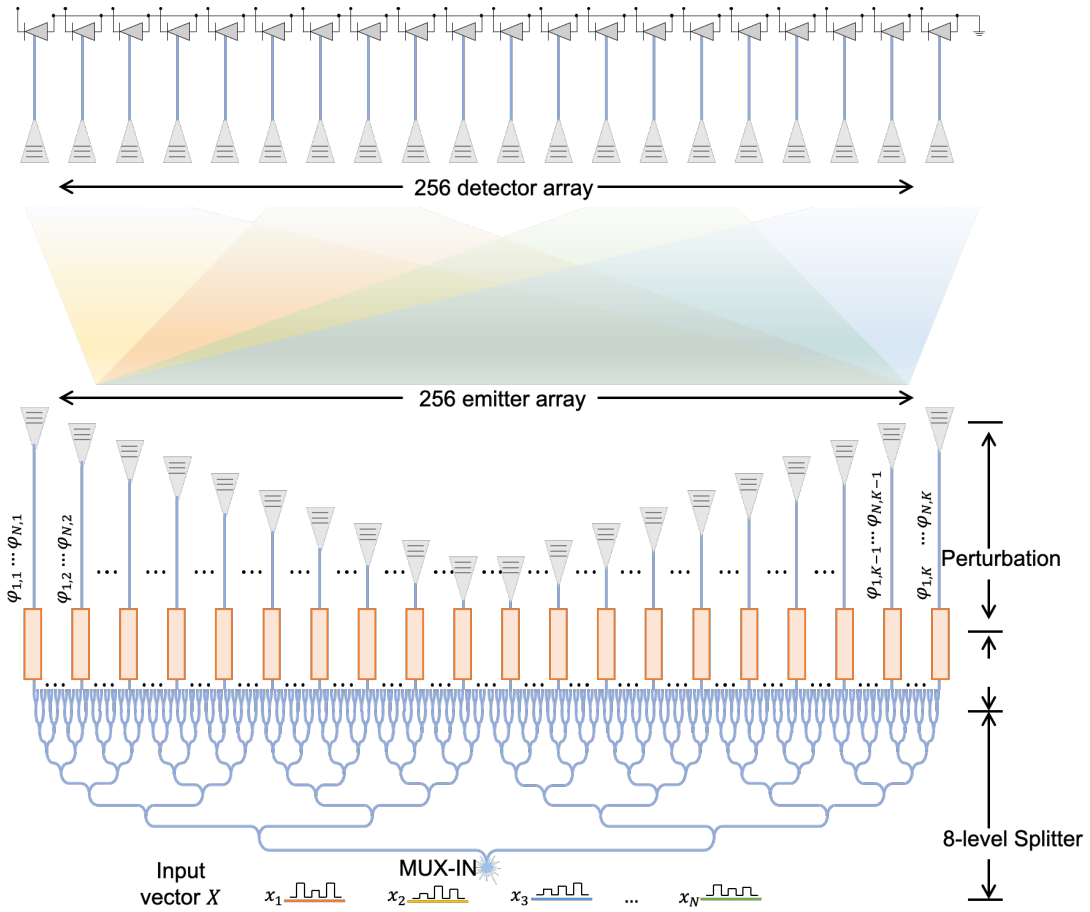
Fig. 1. Single-layer photonic computing. a, Due to the on-chip redundancies, error accumulation arises and precludes optical neural networks (ONNs) of large depth. We propose single-layer photonic chip (SLiM) removing these redundancies and realizing bounded error rate across over 100 layers. **b,** By injecting the propagation perturbation on chip, we decorrelate the weights for arbitrary matrices, thus eliminating the redundant propagations to minimize the error accumulation. Perturbation is further injected into the detection process and decorrelate the signals from the errors, alleviating the error-prone nonlinear layers. The single-layer propagation and detection lead to an error-tolerant photonic chip shown at the bottom panel. **c,** The single-layer computing connects chips in the three-dimensional (3D) hundred-meter space for the execution of billion-parameter deep neural networks, supporting advanced tasks like sentence completion, image recognition, image generation. NL, nonlinearity, MUX, wavelength multiplexing, Scalebar, 1 mm.

As illustrated in the Fig. 1b, in single-layer photonic computing (SLiM) chip, the inputs x_i are assigned to specific wavelengths λ_i. Multi-wavelength light is multiplexed into single fiber channel and coupled into an input loading chip for the configuration of inputs x_i onto corresponding light intensities. To realize the computing matrix W, the transmission channels are modulated on chip by exerting wavefront Φ, derived from the equation $W = |G\Phi|^2$, where G represents the chip-to-chip propagation. However, because of the spectral proximity, the wavefront Φ and the

transmission matrix G both exhibit high correlation, resulting in linear dependence on the right-hand side of the equation that does not guarantee a solution. We introduce on-chip perturbative noise to break the correlation, mathematically represented as $\widehat{\Phi}_k = \Phi_k \cdot e^{jn2\pi\Delta k/\lambda_i}$, where k symbolizes the waveguide coordinate with corresponding noise Δk. The linearly correlated equations are now reformulated as

$$W_i = \left|G_0\widehat{\Phi}_i\right|^2, 1 \le i \le N. \quad (1)$$

The set of equations includes N linearly independent equations, enabling the realization of arbitrary matrices. In realization of the target weight matrix W, the modulation wavefront Φ is first solved with calibrated propagation matrix G, after which the corresponding modulation coefficients are loaded onto the single-layer chips.



Supplementary Figure S14. Circuit diagram of the SLiM chip and schematics of computing.

2. The energy consumption claims appear optimistic. The current calculations exclude essential peripheral components (drivers, TIAs) and the energy overhead from necessary O-E-O conversions between layers, as the SLiM chip only performs matrix-vector multiplications.

Response: We thank the reviewer for the questions. The SLiM chip employs 1-bit encoding and simplifies the circuit design and reduces energy consumption compared to higher-bit converters. In the analysis of the energy efficiency, we have included the peripheral components. Specifically, the 1-bit DAC drives the modulators by charging the capacitor within. Also, for the O-E-O conversions, we have included the optical power, the photodetector and the DAC driving in the calculation. In the revision, we further included the TIAs into the analysis, which usually consumes 1-10mW per channel. The specific modifications are as follows.

Revision:

With current technologies, the power dissipation of 1-bit DAC driving corresponds to the capacitance charge of the modulators, which consumes 6.3 fJ/bit [7]. The ADC process could be simplified to a low-power comparator, which draws 27.3fJ/bit [8]. The transimpedance amplifier consumes around 2.2 mW [9]. The specific power consumptions per channel for each component are as follows: P_{SLiM} = 1.7 μW, P_{DAC} = 63 μW, P_{opt} = 200 μW, P_{DET} = 250 μW, P_{TIA} = 2.2 mW, and P_{ADC} = 273 μW. Denoting the input channel number, output channel number, and working frequency as N, M, and F (10-GHz), the overall system efficiency is calculated as $\eta = 2MNF/\left(N\left(P_{DAC}+P_{opt}\right)+M(P_{ADC}+P_{DET}+P_{TIA})+MNP_{SLiM}\right)$.

Summary: We sincerely thank the reviewer for the constructive comments and suggestions. The main contribution of this work is to validate the concept of hundred-layer photonic computing and to demonstrate the effectiveness of SLiM in supporting LLM-scale models. We believe that the revised version has substantially improved the clarity of the chip implementation and the transparency of the model details.

****

Reviewer #3 (Remarks to the Author):

All my concerns have been addressed.

Please, carefully proofread the paper once again.

Response: We sincerely thank the reviewer for recognizing our innovations. We have thoroughly proofread the manuscripts to improve the readability.

****

Reviewer #1 (Remarks to the Author):

All my concerns have been addressed.

Reviewer #1 (Remarks on code availability):

We have no more comment.

Response: We sincerely thank the reviewer for recognizing our innovations.