# PhyFF: Physical forward forward algorithm for in-hardware training and inference

**Ali Momeni*** EPFL
**Babak Rahmani** Microsoft Research
**Matthieu Malléjac** EPFL
**Philipp del Hougne** University of Rennes, CNRS
**Romain Fleury** EPFL

## Abstract

Training of digital deep learning models primarily relies on backpropagation, which poses challenges for physical implementation due to its dependency on precise knowledge of computations performed in the forward pass of the neural network. To address this issue, we propose a physical forward forward training algorithm (phyFF) that is inspired by the original forward forward algorithm [1]. This novel approach facilitates direct training of deep physical neural networks comprising layers of diverse physical nonlinear systems, without the need for the complete knowledge of the underlying physics. We demonstrate the superiority of this method over current hardware-aware training techniques. The proposed method achieves faster training speeds, reduces digital computational requirements, and lowers training's power consumption in physical systems.

## 1 Introduction

To date, the training of physical neural networks (PNNs) has predominantly relied on backpropagation (BP), a method that has proven highly effective for digital neural networks [2]. However, BP faces several challenges when applied to PNNs, notably due to the complexity and lack of scalability in hardware implementations [3, 4]. Many proposals for PNNs resort to in-silico training, performing BP calculations on an external computer using a digital twin of the physical system. However, this approach often sacrifices speed and increases energy consumption during training. Additionally, the model may not accurately represent the real physical system, leading to a simulation-reality gap and inaccurate inference time prediction [5].

Recent efforts have made progress in addressing these issues. The physics-aware training method based on BP (PA-BP) [5] is the current state-of-the-art framework that mitigates some problems associated with in-silico methods. However, PA-BP still relies on a differentiable digital model for the backward pass, which limits its applicability in terms of training speed and power consumption, as well as requiring extra memory for the backward model. Moreover, PA-BP-trained PNNs may struggle when the physical system experiences strong perturbations, necessitating retraining from scratch. Another limitation of BP is the need for the complete knowledge of the forward pass computations to compute derivatives accurately [1, 6, 7, 8]. When a black box is introduced in the forward pass, BP becomes impractical. Consequently, researchers have been exploring alternative training methods for PNNs. One such approach is the augmented Direct Feedback Alignment (DFA) method [6], designed to eliminate the need for a differentiable digital model. However, DFA is only compatible with certain physical networks where it is possible to separate nonlinear and linear layers.

*ali.momeni@epfl.ch

37th First Workshop on Machine Learning with New Compute Paradigms at NeurIPS 2023(MLNCP 2023).

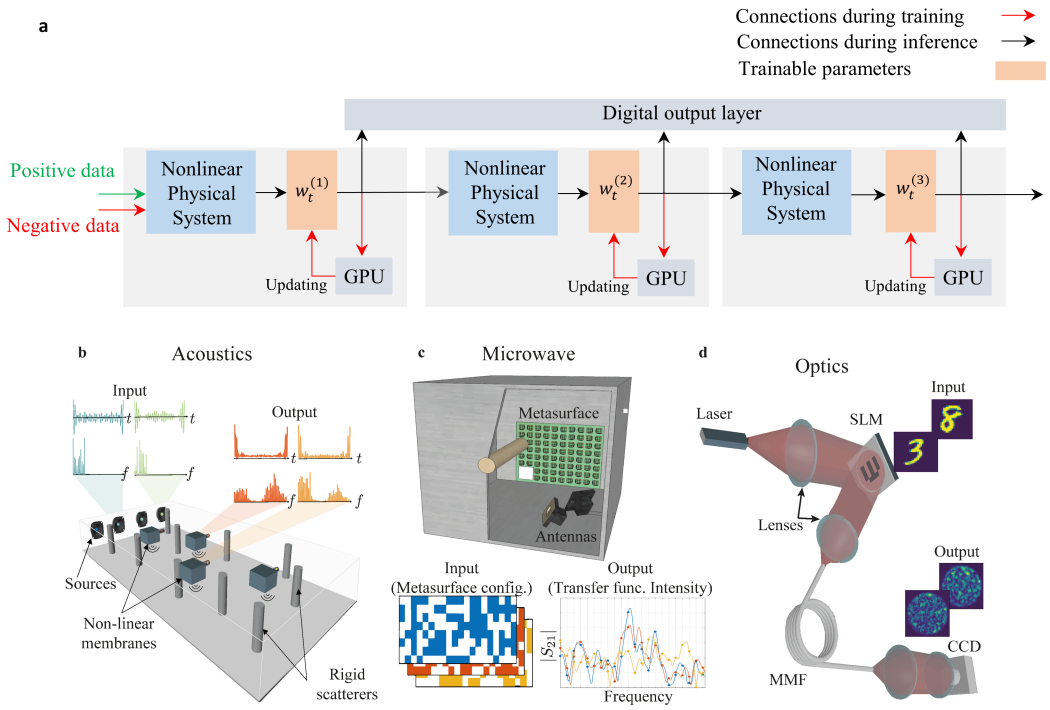


Figure 1 – **Deep physical neural networks.** **a**, A physics-compatible deep neural network that employs a sequence of nonlinear physical data transformers augmented by trainable matrix multiplications, trained by the PhyFF algorithm. We consider three different physical systems in terms of underlying wave phenomenon and type of non-linearity. **b**, In acoustics, input data is encoded into the intensity of sound waves at different frequencies injected on the left side of the cavity. Sound waves propagate through a chaotic cavity that comprises multiple rigid cylindrical diffusers and nonlinear membranes. The transformed waveforms are received by multiple microphones. **c**, In the chaotic microwave cavity, input data is encoded into the programmable metasurface configuration inside the metallic disordered cavity. The outputs are obtained from the waves' spectra (transfer function). **d**, In optical setup, input data is encoded onto the spatial light modulator (SLM), and after passing through the multimode fiber (MMF), the resulting optical intensity is measured on the CCD camera.

In addition, determining the nonlinearity form for PNNs through optimization remains an ongoing challenge.

In this context, we propose a simple and physics-compatible architecture for PNNs, enhanced by a biologically plausible learning algorithm known as the physical forward forward (PhyFF) training. Our proposed method facilitates both supervised and unsupervised training of arbitrary PNNs locally, without any requirement on the knowledge of the nonlinear physical layers nor the need to train a digital twin model. In the PhyFF method, we replace the conventional backward pass, typically executed by a digital computer, with a single forward pass through the physical system. This substitution offers substantial improvements in training speed compared to other hardware-aware training frameworks, while also reducing digital computations, memory usage, and power consumption during the training of wave-based PNNs. To illustrate the versatility of our approach, we perform experimental vowel and image classification using three wave-based systems, each characterized by distinct underlying wave phenomena and types of nonlinearities. The first example features a chaotic acoustic cavity implemented with nonlinear scatterers. The second example involves a chaotic microwave cavity with a transfer function extensively parameterized by a programmable metasurface with structural non-linearity. Our third example showcases a modeled optical multimodal fiber with readout non-linearity (data for these physical systems have been adapted from earlier research, as documented in [9]). We evaluate the performance of PhyFF across various datasets under both supervised and contrastive learning schemes, employing an end-to-end model of these systems for benchmarking purposes.

## 2 Method

Figure 1a depicts a physics-compatible deep PNN consisting of three nonlinear physical data transformers augmented by trainable linear multiplications. Each nonlinear physical data transformer executes a non-linear mapping between the input and output. This is subsequently followed by a trainable linear multiplication for classifying distinct classes through the PhyFF training algorithm. The output of each layer is subsequently passed to the next layer, which then performs the same hierarchical process on the output of its predecessor.

The training algorithm is inspired by the recently proposed forward-forward algorithm [1] which has been extended and adapted to the supervised and unsupervised model-free physical learning of neural networks. As shown in Fig. 1, each nonlinear physical system performs a nonlinear transformation on input data, which can be expressed as $h^{(l)} = f_{\mathrm{N}}^{(l)}(W_{\mathrm{p}}^{(l)} x^{(l)})$, where $x^{(l)}$, $W_{\mathrm{p}}^{(l)}$, and $f_{\mathrm{N}}^{(l)}$ correspond to the physical inputs (e.g., optical intensity, electric voltage, vibration), physical interconnections (e.g., optical, electrical, or mechanical coupling) in the physical system, and physical nonlinearity (e.g., nonlinear optical, magnetic, or mechanical effects) in layer $l$, respectively. Here, $W_{\mathrm{p}}^{(l)}$ and $f_{\mathrm{N}}^{(l)}$ denote the mixing operation and non-linear kernel of the $l$–th physical systems, respectively. Afterward, the output of layer $l$ can be expressed as the multiplication of $h^{(l)}$ by the augmented trainable weight matrix $W_{\mathrm{t}}^{(l)}$, i.e. $y^{(l)} = W_{\mathrm{t}}^{(l)} h^{(l)}$. Such trainable matrix multiplications can be performed either digitally or via physical systems, for instance using Mach-Zehnder Interferometer (MZI) integrated photonics [10] or Spatial Light Modulators (SLMs) in optics[11, 12]. The goal here is to train $\mathrm{W}_{\mathrm{t}}^{(\mathrm{l})}$ locally without the need to know the nonlinear physical layers ($\mathrm{f}_{\mathrm{N}}^{(\mathrm{l})}$ and $\mathrm{W}_{\mathrm{p}}^{(\mathrm{l})}$). Instead of a forward and backward passes, we use here two physical forward passes through the physical system: a positive and a negative forward path, each running on different physical inputs. The positive physical pass, $y_{\mathrm{pos}}^{(l)} = W_{\mathrm{t}}^{(l)} f_{\mathrm{N}}^{(l)}(W_{\mathrm{p}}^{(l)} x_{\mathrm{pos}}^{(l)})$, uses positive inputs that include the input dataset and the correct labels, while the negative physical pass, $y_{\mathrm{neg}}^{(l)} = W_{\mathrm{t}}^{(l)} f_{\mathrm{N}}^{(l)}(W_{\mathrm{p}}^{(l)} x_{\mathrm{neg}}^{(l)})$, uses negative inputs that include the input dataset and the incorrect labels. Refer to [1] to see how labels are added to the inputs to generate both positive and negative data. In unsupervised or contrastive learning, various techniques exist for generating positive and negative pairs. The real data serves as positive data, while negative data can be produced by either shuffling real data in batches or masking positive samples. In each layer, we calculate the so-called "goodness" function, defined as the cosine similarity between the positive and negative activities. Eventually, for each layer l, $\mathrm{W}_{\mathrm{t}}^{(\mathrm{l})}$ is trained by minimizing the following loss function:

$$\mathrm{L}^{(\mathrm{l})} = \log\left(1 + \exp\left(\theta\left(\cos_{\mathrm{sim}}(\mathrm{y}_{\mathrm{pos}}, \mathrm{y}_{\mathrm{neg}})\right)\right)\right) \tag{1}$$

In supervised learning, the goodness function is defined as the cosine similarity between the activities of the layer and a random vector drawn from normal distribution both for the positive and negative physical passes. In this case, the loss function reads as:

$$\mathrm{L}^{(\mathrm{l})} = \log\left(1 + \exp\left(-\theta\left(\cos_{\mathrm{sim}}(\mathrm{y}_{\mathrm{pos}}, \xi^{(\mathrm{l})}) - \cos_{\mathrm{sim}}(\mathrm{y}_{\mathrm{neg}}, \xi^{(\mathrm{l})})\right)\right)\right) \tag{2}$$

In the equations above, $\cos_{\mathrm{sim}}$ is the cosine similarity defined as the cosine of the angle between the two arguments, $\theta$ is a scale factor and $\xi^{\mathrm{l}}$ is the random vector for the layer l and of the same dimension as the output of the layer. The original forward forward algorithm uses only the difference of the positive and negative squared activities, hence necessitating layer normalization to be applied to the data before proceeding to subsequent layers [1]. Conversely, our algorithm avoids incorporating layer normalization into the architecture; this is advantageous because, to this date, there is no efficient hardware implementation for the layer normalization operation. Using cosine similarity in the goodness function allows us to conveniently normalize the outputs without using any extra normalization layers.

During the inference phase, we input a particular label into the PNNs and accumulate the goodness values for all layers. This process is repeated for each label separately. The label with the highest accumulated goodness value is then selected as the output. In unsupervised learning or contrastive learning, a single linear layer is used to map representations from pre-trained hidden layers to labels, eliminating the need to perform the aforementioned process repeatedly. The proposed method is also capable of integrating non-differentiable physical systems or components between the layers.

| | Architectures | Datasets | Details | Algorithms | Test Accuracy (%) |
|---|---|---|---|---|---|
| **DNNs** | Fully-connected | D-MNIST | 2 layers ($676 \times 676$) | Proposed | 97.70 |
| | | | | Original FF [8] | 93.90 |
| | | | | Ideal BP | 98.10 |
| | | CIFAR 10 | 3 layers ($2000 \times 2000$) | Proposed | 56.16 |
| | | | | Original FF [8] | 50.68 |
| | | | | Ideal BP | 57.34 |
| | | F-MNIST | 6 layers ($1000 \times 1000$) | Proposed | 89.13 |
| | | | | Ideal BP | 90.21 |
| **PNNs** | Acoustics-PNN | Vowel | 2 layers ($40 \times 20$) | Proposed | 97.31 |
| | | | | Ideal BP | 97.31 |
| | Microwave-PNN | Vowel | 3 layers ($40 \times 20$) | Proposed | 97.31 |
| | | | | Ideal BP | 98.46 |
| | Optics-PNN | Vowel | 2 layers ($51 \times 51$) | Proposed | 97.14 |
| | | | | Ideal BP | 97.21 |
| | | D-MNIST | 2 layers ($676 \times 676$) | Proposed | 96.41 |
| | | | | Ideal BP | 97.95 |
| | | F-MNIST | 6 layers ($676 \times 676$) | Proposed | 87.80 |
| | | | | Ideal BP | 88.48 |

Table 1 – **Supervised classification results.** Comparison of test accuracy with ideal BP across various datasets and architectures.

## 3 Results

In Figures 1**b-d**, we present three deep PNNs for various standard datasets including vowel, digit, fashion Mnist, and CIFAR10, based on three distinct physical systems. The results for supervised and contrastive learning versions are summarized in Tables 1 and 2 for three different physical systems across various datasets. These results show the high competitiveness of the proposed training method compared to the ideal BP baseline.

## 4 Discussion

Training of ANNs has become substantially costly due to the increasingly growing size of neural networks. Specialized hardware such as PNNs have the potential to drastically decrease these costs by implementing the underlying transformation of data, i.e. the vector matrix multiplication followed by nonlinearities, in hardware. A few methods have been proposed for training PNNs that either entirely (in-silico) or partly (PA-BP [5]) rely on surrogate models for training of the the physical network and hence face issues such as a mismatch between the forward model and the physical system or sensitivity to perturbations. This is because these methods perform the entire backward pass through a digital computer during training, involving either a digital model in PA-BP or numerical

| | Architectures | Datasets | Details | Algorithms | Test Accuracy (%) |
|---|---|---|---|---|---|
| **DNNs** | Convolution | D-MNIST | **3 layers:** 2 convolution (16 channels and kernel size 5 by 5) layers appended with one linear layer ($2704 \times 784$). One decision linear layer ($784 \times 10$) | Proposed | 98.45 |
| | | | | Ideal BP | 98.60 |
| **PNNs** | Optics-PNN | D-MNIST | **3 layers:** 2 physical layers ($676 \times 676$) and one decision linear layer ($676 \times 10$) | Proposed | 96.51 |
| | | | | Ideal BP | 95.12 |

Table 2 – **Contrastive classification results.** Comparison of test accuracy with ideal BP across various datasets and architectures.

simulations in in-silico training, which can hinder their effectiveness in the training phase. PhyFF enables forward passes through physical systems, resulting in a significant speed-up during both inference and training phases while moving away with the simulation-reality gap that is prevalent in hardware-aware training schemes.

## References


[1] Geoffrey Hinton. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.

[2] Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.

[3] Sunil Pai, Zhanghao Sun, Tyler W Hughes, Taewon Park, Ben Bartlett, Ian AD Williamson, Momchil Minkov, Maziyar Milanizadeh, Nathnael Abebe, Francesco Morichetti, et al. Experimentally realized in situ backpropagation for deep learning in nanophotonic neural networks. *arXiv preprint arXiv:2205.08501*, 2022.

[4] Xianxin Guo, Thomas D Barrett, Zhiming M Wang, and AI Lvovsky. Backpropagation through nonlinear units for the all-optical training of neural networks. *Photonics Research*, 9(3):B71–B80, 2021.

[5] Logan G Wright, Tatsuhiro Onodera, Martin M Stein, Tianyu Wang, Darren T Schachter, Zoey Hu, and Peter L McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.

[6] Mitsumasa Nakajima, Katsuma Inoue, Kenji Tanaka, Yasuo Kuniyoshi, Toshikazu Hashimoto, and Kohei Nakajima. Physical deep learning with biologically inspired training method: gradient-free approach for physical hardware. *Nature Communications*, 13(1):7847, 2022.

[7] He Zhu, Yang Chen, Guyue Hu, and Shan Yu. Contrastive learning via local activity. *Electronics*, 12(1):147, 2022.

[8] Heung-Chang Lee and Jeonggeun Song. Symba: Symmetric backpropagation-free contrastive learning with forward-forward algorithm for optimizing convergence. *arXiv preprint arXiv:2303.08418*, 2023.

[9] Ali Momeni, Babak Rahmani, Matthieu Mallejac, Philipp del Hougne, and Romain Fleury. Backpropagation-free training of deep physical neural networks. *arXiv preprint arXiv:2304.11042*, 2023.

[10] Hailong Zhou, Jianji Dong, Junwei Cheng, Wenchan Dong, Chaoran Huang, Yichen Shen, Qiming Zhang, Min Gu, Chao Qian, Hongsheng Chen, et al. Photonic matrix multiplication lights up photonic accelerator and beyond. *Light: Science & Applications*, 11(1):30, 2022.

[11] Maxwell G Anderson, Shi-Yuan Ma, Tianyu Wang, Logan G Wright, and Peter L McMahon. Optical transformers. *arXiv preprint arXiv:2302.10360*, 2023.

[12] Maxime W Matthès, Philipp del Hougne, Julien De Rosny, Geoffroy Lerosey, and Sébastien M Popoff. Optical complex media as universal reconfigurable linear operators. *Optica*, 6(4):465–472, 2019.