



Design of optical neural networks with component imprecisions

MICHAEL Y. -S. FANG,^{1,2,3,*} SASIKANTH MANIPATRUNI,⁴ CASIMIR WIERZYNSKI,⁵ AMIR KHOSROWSHAHI,⁵ AND MICHAEL R. DEWEESE^{1, 2,6}

¹Department of Physics, University of California Berkeley, Berkeley, CA, USA

²Redwood Center for Theoretical Neuroscience, University of California Berkeley, Berkeley, CA, USA

³Molecular Biophysics & Integrated Bioimaging, Lawrence Berkeley National Lab, Berkeley, CA, USA

⁴Components Research, Intel Corporation, Hillsboro, OR, USA

⁵Intel AI Research, San Diego, CA, USA

⁶Helen Wills Neuroscience Institute, University of California Berkeley, Berkeley, CA, USA

*michaelfang@berkeley.edu

Abstract: For the benefit of designing scalable, fault resistant optical neural networks (ONNs), we investigate the effects architectural designs have on the ONNs' robustness to imprecise components. We train two ONNs – one with a more tunable design (GridNet) and one with better fault tolerance (FFTNet) – to classify handwritten digits. When simulated without any imperfections, GridNet yields a better accuracy (~ 98%) than FFTNet (~ 95%). However, under a small amount of error in their photonic components, the more fault tolerant FFTNet overtakes GridNet. We further provide thorough quantitative and qualitative analyses of ONNs' sensitivity to varying levels and types of imprecisions. Our results offer guidelines for the principled design of fault-tolerant ONNs as well as a foundation for further research.

© 2019 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

1. Introduction

Motivated by the increasing capability of artificial neural networks in solving a large class of problems, optical neural networks (ONNs) have been suggested as a low power, low latency alternative to digitally implemented neural networks. A diverse set of designs have been proposed, including Hopfield networks with LED arrays [1], optoelectronic implementation of reservoir computing [2, 3], spiking recurrent networks with microring resonators [4, 5], convolutional networks through diffractive optics [6], and fully connected, feedforward networks using Mach-Zehnder interferometers (MZIs) [7].

We will focus on the last class of neural networks, which consist of alternating layers of modules performing linear operations and element-wise nonlinearities [8]. The N -dimensional complex-valued inputs to this network are represented as coherent optical signals on N single-mode waveguides. Recent research into configurable linear optical networks [9–13] enables the efficient implementation of linear operations with photonic devices. These linear multipliers, layered with optical nonlinearities form the basis of the physical design of ONNs. In Sec. 2, we provide a detailed description of two specific architectures – GridNet and FFTNet – both built from MZIs.

While linear operations are made much more efficient with ONNs in both power and speed, a major challenge to the utility of ONNs lies in their susceptibility to fabrication errors and other types of imprecisions in their photonic components. Therefore, realistic considerations of ONNs require that these imprecisions be taken into account. Previous analyses of the effects of fabrication errors on photonic networks were in the context of post-fabrication optimization of unitary networks [14–16]. Our study differs in three main areas.

First, In the previous work, unitary optical networks were optimized to simulate randomly

sampled unitary matrices. We, instead, train optical neural networks to classify structured data. ONNs, in addition to unitary optical multipliers, include nonlinearities, which add to its complexity.

Second, rather than optimization towards a specific matrix, the linear operations learned for the classification task is not, *a priori*, known. As such, our primary figure of merit is the classification accuracy instead of the fidelity between the target unitary matrix and the one learned.

Lastly, the aforementioned studies mainly focused on the optimization of the networks after fabrication. The imprecisions introduced generally reduced the expressivity of the network – how well the network can represent arbitrary transformations. Evaluation of this reduction in tunability and mitigating strategies were provided. However, such post-fabrication optimization requires the characterization of every MZI, the number of which scales with the dimension (N) of the network as N^2 . Protocols for self configuration of imprecise photonic networks have been demonstrated [17, 18]. While measurement of MZIs were not necessary in such protocols, each MZI needed to be configured progressively and sequentially. Thus, the same N^2 scaling problem remained. Furthermore, if multiple ONN devices are fabricated, each device, with unique imperfections, has to be optimized separately. The total computational power required, therefore, scales with the number of devices produced.

In contrast, we consider the effects of imprecisions introduced after software training of ONNs (Code 1 [19]), details of which we present in Sec. 3. This pre-fabrication training is more scalable, both in network size and fabrication volume. An ideal ONN (i.e., one with no imprecisions) is trained in software only once and the parameters are transferred to multiple fabricated instances of the network with imprecise components. No subsequent characterization or tuning of devices are necessary. In addition to the benefit of better scalability, fabrication of static MZIs can be made more precise and cost effective compared to re-configurable ones.

We evaluate the degradation of ONNs from their ideal performances with increasing imprecision. To understand how such effects can be minimized, we investigate the role that the architectural designs have on ONNs' sensitivity to imprecisions. The results are presented in Sec. 4.1. Specifically, we study the performance of two ONNs in handwritten digit classification. GridNet and FFTNet are compared in their robustness to imprecisions. We found that GridNet achieved a higher accuracy ($\sim 98\%$) when simulated with ideal components compared to FFTNet ($\sim 95\%$). However, FFTNet is much more robust to imprecisions. After the introduction of realistic levels of error, the performance of GridNet quickly degrades to below that of FFTNet. We also show, in detail, the effect that specific levels of noise has on both networks.

In Sec. 4.2, we demonstrate that this is due to more than the shallow depth of FFTNet and that FFT-like architectures is more robust to error when compared to Grid-like architectures of the same depth.

In Sec. 4.3, we investigate the effects localized imprecisions have on the network by constraining the imprecisions to specific groups of MZIs. We demonstrate that the network's sensitivity to imprecisions is dependent on algorithmic choices as well as its physical architecture.

With a growing interest in optical neural networks, a thorough analysis of the relationship between ONNs' architecture and its robustness to imprecisions and errors is necessary. From the results that follow, in this article, we hope to provide a reference and foundation for the informed design of scalable, error resistant ONNs.

2. Physical design of optical neural networks

The ONN consists of multiple layers of programmable optical linear multipliers with intervening optical nonlinearities (Fig. 2). The linear multipliers are implemented with two unitary multipliers and a diagonal layer in the manner of a singular-value decomposition (SVD). These are, in turn, comprised of arrays of configurable MZIs, which each consist of two phaseshifters and two beamsplitters (Fig. 1(a)).

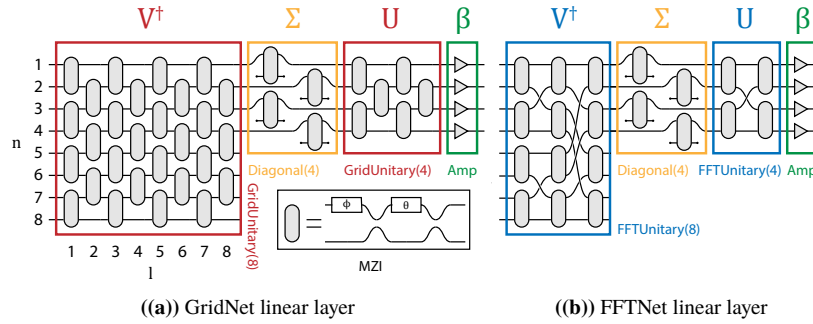


Fig. 1. a) A schematic of a universal 8×4 optical linear multiplier with two unitary multipliers (red) consisting of MZIs in a grid-like layout and a diagonal layer (yellow). The MZIs of GridUnitary multipliers are indexed according to their layer depth (l) and dimension (d). Symbols at the top represent the mathematical operations performed by the various modules. Inset: A MZI with two 50:50 beamsplitters and two tunable phaseshifters b) An FFT-like, non-universal multiplier with FFTUnitary multipliers (blue).

Complex-valued N -dimensional input vectors are encoded as coherent signals on N waveguides. Unitary mixing between the channels is effected by MZIs and forms the basis of computation for ONNs. A single MZI consists of two beamsplitters and two phaseshifters (PS) (Fig. 1(a) inset). While the fixed 50:50 beamsplitters are not configurable, the two phaseshifters, parameterized by θ and ϕ , are to be learned during training. Each MZI is characterized by the following transfer matrix (see App. A for details):

$$U_{MZ}(\theta, \phi) = U_{BS}U_{PS}(\theta)U_{BS}U_{PS}(\phi) = ie^{i\theta/2} \begin{pmatrix} e^{i\phi} \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \\ e^{i\phi} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \end{pmatrix}. \quad (1)$$

Early work has shown that universal optical unitary multipliers can be built with a triangular mesh of MZIs [9]. These multipliers enabled the implementation of arbitrary unitary operations and were incorporated into the ONN design by Shen et al. [7]. Its asymmetry prompted the development of a symmetric grid-like network with more balanced loss [10]. By relaxing the requirement on universality, a more compact design, inspired by the Cooley-Tukey FFT algorithm [20], has been proposed [11]. It can be shown that FFT transforms, and therefore convolutions, can be achieved with specific phase configurations (see appendix H). We allow the phase configurations to be learned for implementation of a greater class of transformations.

In this section, we focus on the last two designs, referring to them as GridUnitary (Fig. 1(a)) and FFTUnitary (Fig. 1(b)), respectively. GridUnitary can implement unitary matrices directly by setting the phaseshifters using an algorithm by Clements et al. [10]. Despite being non-universal and lacking a decomposition algorithm, FFTUnitary can be used to reduce the depth of the unitary multipliers from N to $\log_2(N)$. Reducing the number of MZIs leads to lower overall noise and loss in the network. However, due to the FFT-like design, waveguide crossings are necessary. To overcome this challenge, low-loss crossings [21] or 3D layered waveguides [22,23] could be utilized.

MZIs can also be used to attenuate each channel separately without mixing. This way, a diagonal multiplier can be built. Because signals can only be attenuated by MZIs, subsequent global optical amplification [24] is needed to emulate arbitrary diagonal matrices. Through SVD, a universal linear multiplier can be created from two unitary multipliers and a diagonal multiplier

(Fig. 1(a)). Formally, a linear transformation represented by matrix M can be decomposed as

$$M = \beta \cdot U \Sigma V^\dagger. \quad (2)$$

Here both U and V^\dagger are unitary transfer matrices of GridUnitary multipliers while Σ represents a diagonal layer with eigenvalues no greater than one. β is a compensating scaling factor.

Along with linear multipliers, nonlinear layers are required for artificial neural networks. In fact, the presence of nonlinearities sets the study of ONNs apart from earlier research in linear photonic networks [25]. One possible implementation is by saturable absorbers such as monolayer graphene [26]. This has the advantage of being easily approximated with a Softplus function (see Sec. 3 for details on implementation). However, it has been demonstrated that Softplus underperforms, in many regards, when compared to rectified linear units (ReLU) [27]. Indeed, a complex extension of ReLU, ModReLU, has been proposed [28]. While it is physically unrealistic to implement ModReLU, the nonoptimality of Softplus functions still motivates the exploration of other optical nonlinearities, such as optical bistability in microring resonators [29], and two-photon absorption [30, 31] as alternatives.

3. Neural network architecture and software implementation

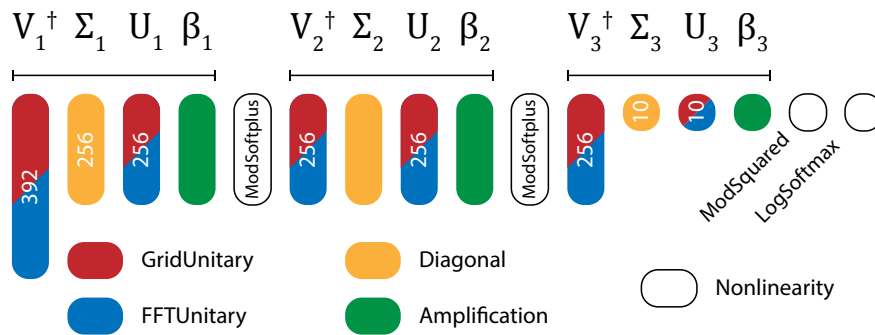


Fig. 2. Network design used for the MNIST classification task. GridNet used universal unitary multipliers while FFTNet used FFT-Unitary multipliers. See Fig. 1 for details of physical implementation of the three linear layers.

We considered a standard deep learning task of MNIST handwritten digit classification [32]. Fully connected feedforward networks with two hidden layers of 256 complex-valued neurons each were implemented with GridNet and FFTNet architectures (Fig. 2) and simulated in PyTorch [33]. The $28^2 = 784$ dimensional real-valued input was converted into $392 = 784/2$ dimensional complex-valued vectors by taking the top and bottom half of the image as the real and imaginary part. This was done to ensure the data is distributed evenly throughout the complex plane rather than just along the real number line.

Each network consists of linear multipliers followed by nonlinearities. The linear layers of GridNet and FFTNet were described in the previous section and illustrated in Fig. 1. The response curve of the saturable absorption is approximated by the Softplus function [34] (App. C), a commonly used nonlinearity available in most deep learning libraries such as PyTorch. The nonlinearity is applied to the modulus of the complex numbers. A modulus squared nonlinearity modeling an intensity measurement is then applied. The final SoftMax layer allows the (now real) output to be interpreted as a probability distribution. A cross-entropy [35] loss function is used to evaluate the output distribution against the ground truth.

An efficient implementation of GridNet requires representing matrix-vector multiplications as element-wise vector multiplications [36]. Nevertheless, training the phaseshifters directly was

still time consuming. Instead, a complex-valued neural network [37] was first trained. An SVD (Eq. (2)) was then performed on each complex matrix. Finally, phaseshifters were set to produce the unitary (U, V^\dagger) and diagonal (Σ) multipliers through a decomposition scheme by Clements et al. [10].

However, note that SVD is ambiguous up to permutations (Π) of the singular values and the columns of U and V .

$$U\Sigma V^\dagger = (U\Pi^{-1})(\Pi\Sigma\Pi^{-1})(\Pi V^\dagger). \quad (3)$$

Conventionally, the ambiguity is resolved through ordering the singular values from largest to smallest. In Sec. 4.3 we show that randomizing the singular values increases the error tolerance of GridNet. FFTNet is trained directly and its singular values are naturally unordered. For a fair comparison, we randomly permute the singular values of GridNet.

After 10 training epochs with standard stochastic gradient descent [38], classification accuracies of 97.8% (GridNet) and 94.8% (FFTNet) were achieved. Better accuracies can be achieved through convolutional layers [39], Dropout regularization [40], better training methods, etc. However, we omitted these in order to focus purely on the effects of architecture.

The networks were trained assuming ideal components represented with double-precision floating point values. Under realistic conditions, due to imprecision in fabrication, calibration, etc., the realizable accuracy could be much lower. During inference, we modeled these imprecisions by adding independent zero-mean Gaussian noise of standard deviation σ_{PS} and σ_{BS} to the phases (θ, ϕ) of the phaseshifters and the transmittance T of the beamsplitters, respectively. Reasonable values for such imprecisions can be taken to be approximately $\sigma_{PS} \approx 0.01\text{rad}$ and $\sigma_{BS} \approx 1\% = 0.01$ [41,42]. Note that the dynamical variation due to laser phase noise can be modeled by σ_{PS} as well. However, we show in App. B that typical values would be well below 0.01 rad.

4. Results

4.1. Degradation of network accuracy

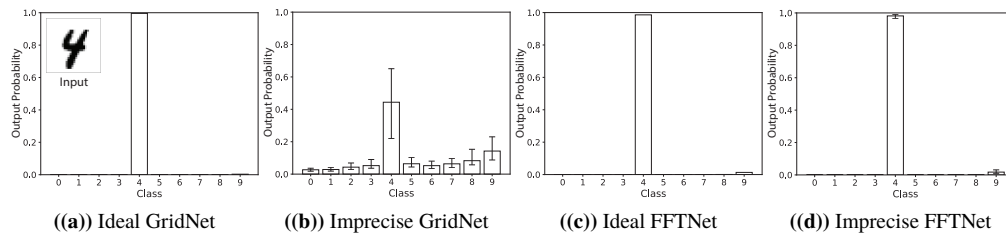


Fig. 3. Visualizing the degradation of ONN outputs, FFTNet is seen to be much more robust than GridNet. Identical input is fed through GridNet (a, b) and FFTNet (c, d), simulated with ideal components (a, c) and imprecise components (b, d) with $\sigma_{BS} = 0.01$ and $\sigma_{PS} = 0.01\text{rad}$. Imprecise networks are simulated 100 times and their mean output is represented by bar plots. Error bars represent the 20th to 80th percentile range.

To investigate the degradation of the networks due to imprecisions, we started by simulating 100 instances of imprecise networks with $\sigma_{BS} = 1\%$ and $\sigma_{PS} = 0.01\text{rad}$. Identical inputs of a digit “4” (Fig. 3(a) inset) are fed through each network. The mean and spread of the output of the ensemble is plotted and compared against the output from the ideal network (Fig. 3).

The degradation of classification output is significant for GridNet. Without imprecisions in the photonic components, the digit is correctly classified with near 100% confidence (Fig. 3(a)).

When imprecisions are simulated, we see a large decrease in classification confidence (Fig. 3(b)). In particular, the image is often misclassified when the prediction probability for class “9” is greater than that for class “4”. Repeating these experiments on FFTNet demonstrated that they were much more resistant to imprecisions (Fig. 3(c), 3(d)). In Appendix D, we show confusion matrices of both networks with increasing error to further support this conclusion.

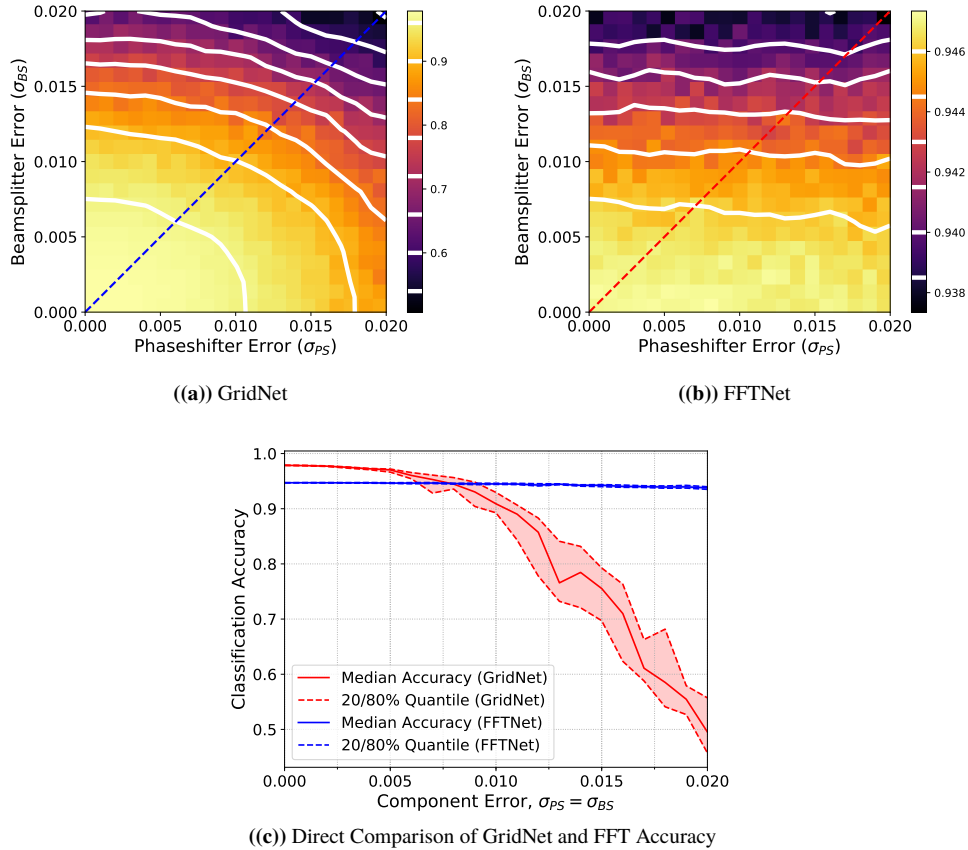


Fig. 4. The decrease in classification accuracy is visualized for GridNet and FFTNet. (a,b) The two networks were tested with simulated noise of various levels for 20 runs. The mean accuracy is plotted as a function of σ_{PS} and σ_{BS} . Note the difference in color map ranges between the two plots. (c) The accuracies of GridNet and FFTNet are compared along the $\sigma_{PS} = \sigma_{BS}$ cutline.

Evaluating the two networks on overall classification accuracy confirms the superior robustness to imprecisions of FFTNet. GridNet and FFTNet were tested at levels of imprecisions with σ_{PS}/rad and σ_{BS} ranging from 0 to 0.02 with a step size of 0.001. At each level of imprecision, 20 instances of each network were created and tested. The mean accuracies are plotted in Fig. 4(a), 4(b). A direct comparison between the two networks along the diagonal (i.e., $\sigma_{PS} = \sigma_{BS}$ cut line, taking 1% = 0.01 rad) is shown in Fig. 4(c).

Starting at roughly 98% with ideal components, the accuracy of GridNet rapidly drops with increasing σ_{PS} and σ_{BS} . By comparison, very little change in accuracy is seen for FFTNet despite starting with a lower ideal accuracy. Also of note are the qualitatively different levels of sensitivity of the different components to imprecision. In particular, FFTNet is much more resistant to phaseshifter error compared to beamsplitter error.

The experiments described in this section confirm the significant effect component imprecisions have on the overall performance of ONNs, as well as the importance of architecture in determining the network's robustness to these imprecisions. Despite having a better classification accuracy in the absence of imprecisions, GridNet is surpassed by FFTNet when a small amount of error ($\sigma_{PS} = 0.01$, $\sigma_{BS} = 1\%$ rad) is present. In Appendix E, we demonstrate that FFTNet is also more robust to quantization error than GridNet.

4.2. Stacked FFTUnitary and truncated GridUnitary

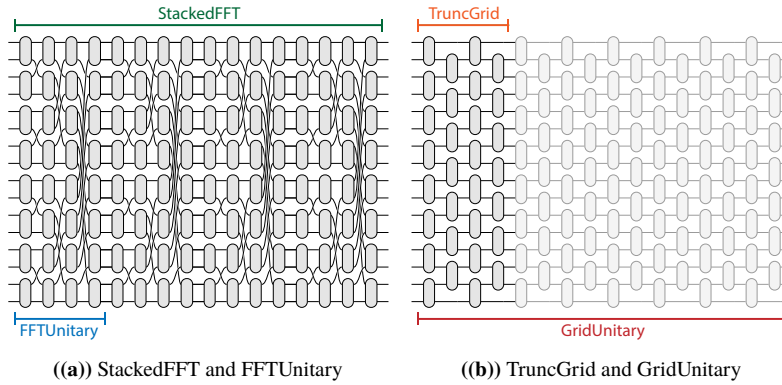


Fig. 5. The architecture of a) StackedFFT and b) TruncGrid shown with FFTUnitary and GridUnitary from which they were derived. For clarity, the dimension, here, is $N = 2^4 = 16$ so FFTUnitary was stacked four times and GridUnitary was truncated at the fourth layer. In the experiments described in this section, the dimension was taken to be $N = 2^8 = 256$.

One obvious reason why FFTNet would be more robust than GridNet is its much lower number of MZI layers. Their respective, constituent unitary multipliers, FFTUnitary and GridUnitary contains $\log_2(N)$ and N layers respectively. For $N = 2^8 = 256$, GridUnitary is 32 times deeper than FFTUnitary which contains only 8 layers.

To demonstrate that FFTUnitary is more robust due architectural reasons beyond its shallow depth, in this section, we introduce two unitary multipliers – StackedFFT (Fig. 5(a)) and TruncGrid (Fig. 5(b)). StackedFFT consists of FFTUnitary multipliers stacked end-to-end 32 times and TruncGrid is the GridUnitary truncated after 8 layers of MZIs. This way, FFTUnitary and TruncGrid have the same depth as do GridUnitary and StackedFFT.

Unitary multipliers by themselves are not ONNs and cannot be trained for classification tasks. Instead, after introducing imprecisions to the each multiplier, we evaluated the fidelity $F(U_0, U)$ between the original, error-free transfer matrix U_0 and the imprecise transfer matrix U . The fidelity, a measure of “closeness” between two unitary matrices, is defined as [43]

$$F(U_0, U) = \left| \frac{\text{Tr}(U^\dagger U_0)}{N} \right|^2. \quad (4)$$

Ranging from 0 to 1, $F(U_0, U) = 1$ only when $U = U_0$. Using this metric of fidelity, we show that StackedFFT is more robust to error than GridUnitary (Fig. 6(a)) and TruncGrid more than FFTUnitary (Fig. 6(b)). Both comparisons are between multipliers with the same number of MZI layers. Yet, the FFT-like architectures are still more robust to their grid-like counterparts.

One possible explanation could be the better mixing facilitated by FFTUnitary. GridUnitary and thus TruncGrid, at each MZI layer, only mixes neighboring waveguides. After P layers, each

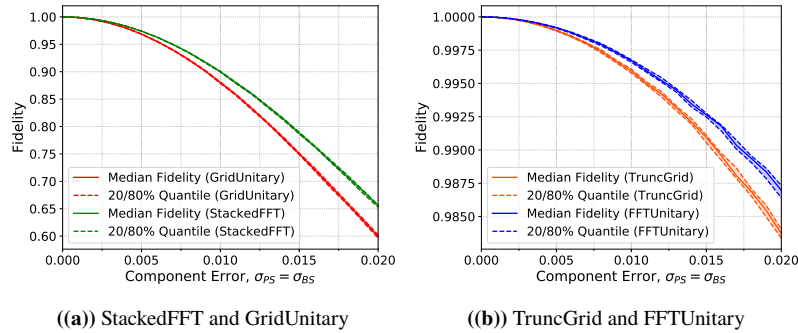


Fig. 6. With the same layer depth, multipliers with FFT-like architectures are shown to be more robust. The fidelity between the error-free and imprecise transfer matrices is plotted as a function of increasing error. Two sets of comparisons between unitary multipliers of the same depth are made. a) Both StackedFFT and GridUnitary have $N = 256$ layers of MZIs. b) TruncGrid and FFTNet have $\log N = 8$ layers.

waveguide is connected to, at most, to its $2P$ nearest neighbors. In comparison, after P layers, FFTUnitary connects $N = 2^P$.

Here, we have compared the robustness of different unitary multipliers in isolation. We stress that the overall robustness of neural networks is a much more complex and involved problem. A rough understanding can be formulated as follows. A trained neural network defines a decision boundary throughout the input space. Introduction of errors perturbs the decision boundary which can lead to misclassification. To reduce this effect, we can make the decision boundary of ONNs more robust to errors. However, it is also important to consider the robustness of misclassification due to perturbations of decision boundaries. Indeed, it has been shown that robustness of neural networks are dependent on the geometry of the boundary [44].

A complete analysis of the robustness of neural networks to various forms of perturbations is outside the scope of this paper. Nonetheless, it is important to understand the dependence of ONNs on both architectural and algorithmic design.

4.3. Localized imprecisions

To better understand the degradation of network accuracy, we mapped out the sensitivity of GridNet to specific groups of MZIs. A relatively large amount of imprecision ($\sigma_{PS} = 0.1\text{rad}$) was introduced to 8×8 blocks of MZIs in layer 2 (Fig. 2) of an otherwise error-free GridNet. The resulting change in classification accuracy is plotted as a function of the position of the MZI block (Fig. 7). We see no strong correlation between the change in accuracy and the spatial location of the introduced error. In fact, error in many locations led to small increases in accuracy, suggesting that much of the effect is due to chance.

This result seems to contradict previous studies on the spatial tolerance of MZIs in a GridUnitary multiplier [14–16]. It was discovered that the central MZIs of the multiplier had a much lower tolerance than those near the edges. When learning randomly sampled unitary matrices, the central MZIs needed to have phase shift values very close to 0 (π , following the convention used in this paper). This would only be achievable with MZIs with extremely high extinction ratios and thus low fabrication error.

Empirically, this distribution of phases was observed in GridUnitary multipliers of trained ONNs (See app. F). However, the idea of tolerance of a MZI to beamsplitter fabrication imprecision, while related, is not the same as the network sensitivity to localized imprecisions. To elaborate, tolerance is implicitly defined, in references [14–16], as roughly the allowable

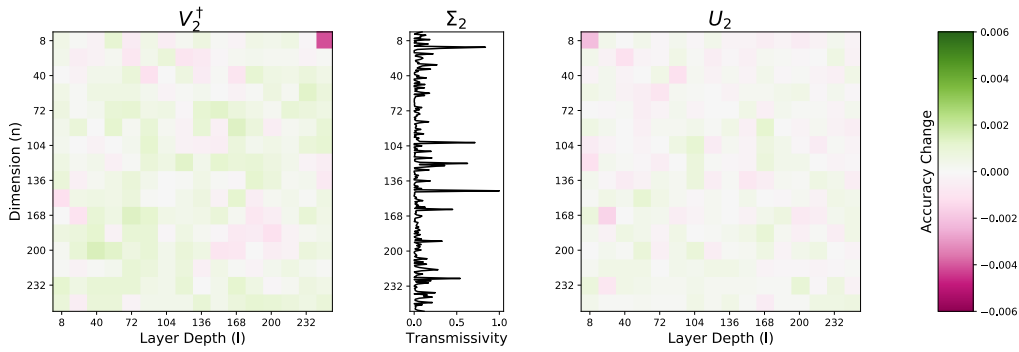


Fig. 7. Change in accuracy due to localized imprecision in layer 2 of GridNet with randomized singular values. A large amount of imprecision ($\sigma_{PS} = 0.1\text{rad}$) is introduced to 8×8 blocks of MZIs in an otherwise error-free GridNet. The resulting change in accuracy of the network is plotted as a function of the position of the MZI block in GridUnitary multipliers V_2^\dagger and U_2 (coordinates defined as in Fig. 1(a)). The transmissivity of each waveguide through the diagonal layer Σ_2 is also plotted (center panel).

beamsplitter imperfection (deviation from 50:50) that still permits post-fabrication optimization of phase shifter towards arbitrary unitary matrices. In our pre-fabrication optimization approach, we take sensitivity to be the deviation from ideal classification accuracy when imprecision is introduced to the MZI with no further reconfiguration. See App. G for this difference further illustrated by experiments with another architecture.

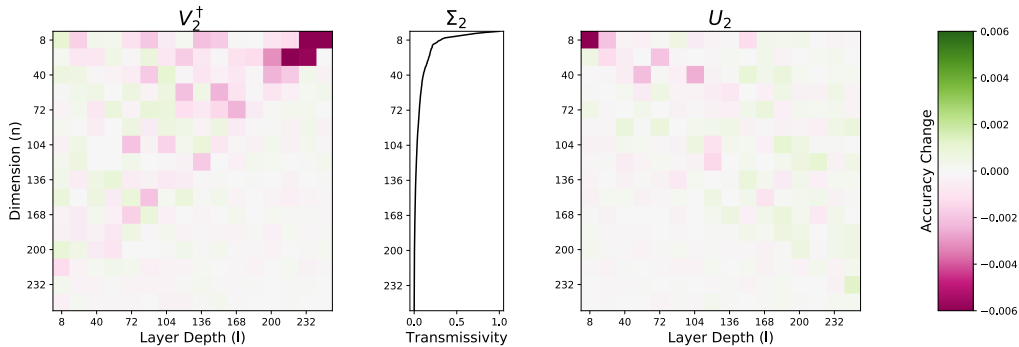


Fig. 8. Effects of localized imprecision in layer 2 of GridNet with ordered singular values. Similar to Fig. 7, except GridNet has its singular values ordered. Therefore, the transmissivity is also ordered (center panel).

Recall that the singular values Σ of the GridNet's linear layers could be permuted together with columns and rows of U and V^\dagger respectively without changing the final transfer matrix (Eq. (3)). The singular values were randomized to provide a fair comparison with FFTNet. We then performed the same experiment on GridNet where the singular values of each layer were not randomized but ordered from largest to smallest. Therefore, the transmissivity $T = |\sin(\theta/2)|^2$ of the diagonal multiplier Σ is also ordered (Fig. 8). In this case, there is a significant, visible pattern because most of the signal travels through the top few waveguides of Σ_2 due to the ordering of transmissivities. Only MZIs connected to those waveguides have a strong effect on the network. In fact, the network is especially sensitive to imprecisions in MZIs closest to this bottleneck (Fig. 8, top-right of V_2^\dagger and top-left of U_2). It is important to note that this bottleneck only exist due

to the locality of connections in GridNet where only neighboring waveguides are connected by MZIs. In FFTNet, due to crossing waveguides, no such locality exist.

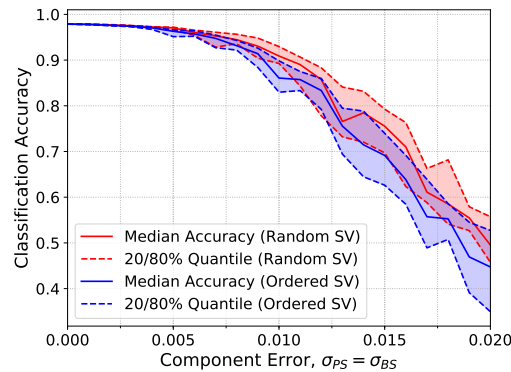


Fig. 9. The degradation of accuracies with increasing $\sigma_{PS} = \sigma_{BS}$ compared between two GridNets one with ordered and another with randomized (but fixed) singular values.

In addition to, and likely due to the spatial non-uniformity in error sensitivity, GridNet with ordered singular values is more susceptible to uniform imprecisions (Fig. 9). The same GridNet architecture, could be made more resistant by shuffling its singular values. This difference between two identical architectures implementing identical linear and non-linear transformations demonstrates that the resistance to error in ONNs is effected by more than architecture.

5. Conclusion

Having argued that pre-fabrication, software optimization of ONNs is much more scalable than post-fabrication, on-chip optimization, we compared two types of networks—GridNet and FFTNet in their robustness to error. These two networks were selected to showcase the trade-off between expressivity and robustness. We demonstrated in Sec. 4.1 that the output of GridNet is much more sensitive to errors than FFTNet. We have illustrated the robustness of FFTNet by a providing a thorough evaluation of both networks operating with imprecisions ranging between $0 \leq \sigma_{BS}, \sigma_{PS} \leq 0.02$. With ideal accuracies of 97.8% and 94.8% for GridNet and FFTNet respectively, GridNet accuracy dropped rapidly to below 50% while FFTNet maintained near constant performance. Under conservative assumptions of errors associated with the beamsplitter ($\sigma_{BS} > 1\%$) and phase shifter ($\sigma_{PS} > 0.01$ rad), a more robust network (FFTNet) can be favorable over one with greater expressivity (GridNet).

We then demonstrate, in Sec. 4.2, through modified unitary multipliers, TruncGrid and StackedFFT, that controlling for MZI layer depth, FFT-like designs are inherently more robust than grid-like ones.

To gain a better understanding of GridNet's sensitivity to imprecision, in Sec. 4.3, we probed the response of the network to localized imprecisions by introducing error to small groups of MZIs at various locations. The sensitivity to imprecisions was found to be less affected by the MZIs' physical position within the grid and more so by the flow of the optical signal. We then demonstrated that beyond architectural designs, small procedural changes to the configuration of an ONN, such as shuffling the singular values, can change affect the its robustness.

Our results, presented in this paper, provide clear guidelines for the architectural design of efficient, fault-resistant ONNs. In looking forward, it would be important to investigate algorithmic and training strategies as well. A central problem in deep learning is to design neural networks complex enough to model the data while being regularized to prevent over-fitting of noise in the training set [8]. To this end, a wide variety of regularization techniques such as

Dropout [40], Dropconnect [45], data augmentation, etc. have been developed. This problem parallels the trade-off between an ONN's expressivity and its robustness to imprecisions presented here. Indeed, an important conclusion in Sec. 4.3 is that in addition to architecture, even minor changes in the configuration of ONNs also have a great effect on the network's robustness to faulty components.

The robustness of neural networks to perturbations [44] is a well studied and open problem that is outside of the scope of this article on architectural design. Nevertheless, a complete analysis of ONNs with imprecise components requires an understanding of robustness due to architectural design as well as due to software training, possibly under a unifying framework. A natural direction for further exploration is to consider analogies to regularization in the context of imprecise photonic components and to focus on the development of algorithms and training strategies for error-resistant optical neural networks.

Appendix

A. MZI transfer matrix

Because MZIs are comprised of beamsplitters and phaseshifters, we state their respective transfer matrix first.

$$U_{BS}(r) = \begin{pmatrix} r & it \\ it & r \end{pmatrix} \quad (5)$$

where $t \equiv \sqrt{1 - r^2}$ and

$$U_{PS}(\theta) = \begin{pmatrix} e^{i\theta} & 0 \\ 0 & 1 \end{pmatrix}. \quad (6)$$

With the construction of PS-BS-PS-BS (Fig. 1(a), inset), the MZI transfer matrix is the following matrix product:

$$U_{MZI}(\theta, \phi; r, r') = U_{BS}(r)U_{PS}(\theta)U_{BS}(r')U_{PS}(\phi) \quad (7)$$

$$= \begin{pmatrix} r & it \\ it & r \end{pmatrix} \begin{pmatrix} e^{i\theta} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} r' & it' \\ it' & r' \end{pmatrix} \begin{pmatrix} e^{i\phi} & 0 \\ 0 & 1 \end{pmatrix} \quad (8)$$

$$= \begin{pmatrix} e^{i\phi} (e^{i\theta} r r' - t t') & i (t \rho + e^{i\theta} r t') \\ i e^{i\phi} (e^{i\theta} t r' + r t') & r \rho - e^{i\theta} t t' \end{pmatrix} \quad (9)$$

Assuming that the beamsplitter ratios are 50:50, we can take $r = t = 1/\sqrt{2}$ so that

$$U_{BS} \equiv U_{BS}(\pi/2) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix} \quad (10)$$

and therefore,

$$U_{MZI}(\theta, \phi) = i e^{i\theta/2} \begin{pmatrix} e^{i\phi} \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \\ e^{i\phi} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \end{pmatrix} \quad (11)$$

In our convention, the transmission and reflection coefficient is

$$T = |\cos \theta/2|^2 \text{ and } R = |\sin \theta/2|^2 \quad (12)$$

respectively. In particular, the MZI is in the bar state ($T = 0$) when $\theta = \pi$ and in the cross state ($T = 1$) when $\theta = 0$.

However, in other conventions, the beamsplitter is often taken to be the Hadamard gate.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \quad (13)$$

We note however, that

$$U_{BS} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} H \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} = U_{PS}(-\pi/2) H U_{PS}(-\pi/2) \quad (14)$$

up to a global phase. We then can express the MZI transfer matrix as

$$U_{MZ}(\theta, \phi) = U_{PS}(-\pi/2) H U_{PS}(\theta - \pi) H U_{PS}(\phi - \pi/2). \quad (15)$$

Note in this convention the internal phase shift is now $\theta + \pi$ and thus the bar and cross states are now at $\theta = 0$ and $\theta = \pi$ respectively.

B. Laser phase noise

The variance in phase for typical lasers can be modeled as [46]

$$\sigma_\phi(\tau)^2 = 2\pi \cdot \delta f \cdot \tau. \quad (16)$$

Here, τ is the time of integration and δf the linewidth of the laser. For an order of magnitude calculation, we ignore the refractive index and take $\tau = L/c$ where L is the distance between two subsequent phaseshifters on an MZI. Again, as an order of magnitude estimate, we take $L = 100\mu\text{m} = 10^{-4}\text{m}$ and thus $\tau \approx 3 \times 10^{-13}$. We wish to solve for the linewidth required for $\sigma_\phi = 0.01\text{rad}$:

$$\sigma_\phi^2 = 10^{-4} = 2\pi \cdot \delta f \cdot \tau \quad (17)$$

$$\approx 6 \cdot 3 \times 10^{-13} s \delta f \quad (18)$$

$$\delta f \approx 5 \times 10^7 \text{Hz} \quad (19)$$

$$= 50\text{MHz}. \quad (20)$$

A linewidth of 50 MHz is easily achieved by modern lasers. For example, Bragg reflector lasers have been shown to achieve a linewidth of 300 kHz [47]. Thus, the contribution to phase noise from the laser is roughly two orders of magnitude smaller than that from MZIs.

C. Approximating saturable absorption

Saturable absorption can be modeled by the relation [48]

$$u_0 = \frac{1}{2} \frac{\log(T/T_0)}{1 - T} \quad (21)$$

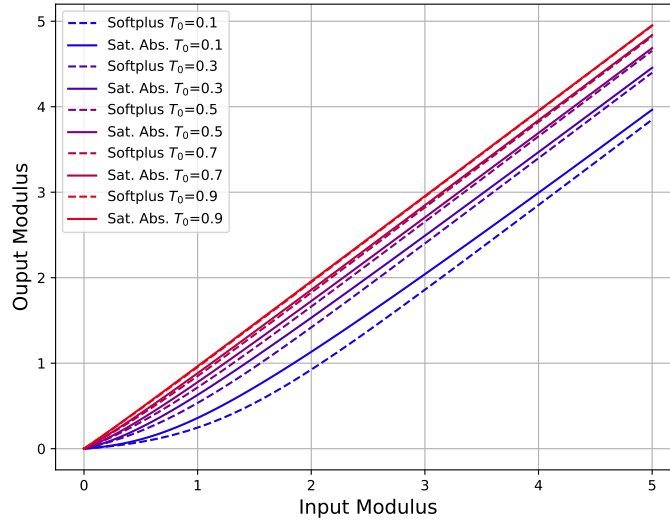


Fig. 10. The saturable absorption response curve compared to the corresponding Softplus approximation with various values of T .

where $T = u/u_0$ and $u = \sigma\tau_s I$ and $u_0 = \sigma\tau_s I_0$. I_0, I are the incidental and transmitted intensities, respectively. The above equation can be solved to be

$$u = \frac{1}{2}W(2T_0u_0e^{2u_0}) \equiv f(u_0) \quad (22)$$

Where W is the product log function or Lambert W function. However, since W is not readily available in most deep learning libraries and difficult to implement, we wish to approximate the above by the shifted and biased Softplus non-linearity of the form

$$\sigma(u) = \beta^{-1} \log \left(\frac{1 + e^{\beta(u-u_0)}}{1 + e^{-\beta u_0}} \right). \quad (23)$$

The bias of $-\beta^{-1} \log(1 + e^{-\beta u_0})$ was chosen to ensure that $\sigma(0) = f(0) = 0$. We now choose β and u_0 to ensure that

1. $\sigma'(0) = f'(0) = T_0$,
2. $\lim_{u \rightarrow \infty} \sigma(u) - u = \lim_{u \rightarrow \infty} f(u) - u = \frac{1}{2} \log T_0$.

The derivative of $\sigma(u)$ is easily found to be

$$\sigma'(0) = \frac{e^{-\beta u_0}}{1 + e^{-\beta u_0}} \quad (24)$$

$$= (1 + e^{\beta u_0})^{-1}. \quad (25)$$

Requiring that it equals to $f'(0) = T_0$ allows us to solve for

$$u_0 = \beta^{-1} \log \left(T_0^{-1} - 1 \right). \quad (26)$$

Next, in the large u limit, the biased Softplus converges to

$$\sigma(u) \rightarrow (u - u_0) - \beta^{-1} \log(1 + e^{-\beta u_0}). \quad (27)$$

Solving for equality with $f(u) \rightarrow u + \frac{1}{2} \log T_0$ gives

$$u_0 + \beta^{-1} \log(1 + e^{-\beta u_0}) = -\frac{1}{2} \log T_0 \quad (28)$$

$$\beta u_0 + \log\left(1 + \frac{1}{T_0^{-1} - 1}\right) = -\frac{1}{2} \beta \log T_0 \quad (29)$$

$$\log(T_0) = -\frac{1}{2} \beta \log T_0 \quad (30)$$

$$\beta = 2. \quad (31)$$

Going back to Eq. (26), we obtain

$$u_0 = \frac{1}{2} \log(T_0^{-1} - 1). \quad (32)$$

Fig. 10 plots the saturable absorption response curve compared to the Softplus approximation derived above.

D. Confusion matrices

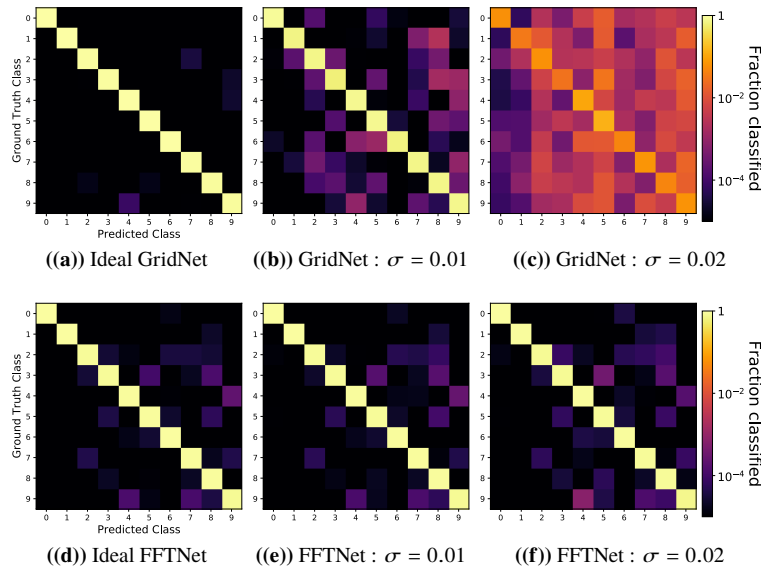


Fig. 11. The degradation of ONN outputs visualized through confusion matrices. Each confusion matrix shows how often each target class (row) is predicted as each of the ten possible classes (column). Both networks, GridNet (a, b, c) and FFTNet (d, e, f) are evaluated. First in the ideal case (a, d) then, with increasing errors (b, e and c, f). Note the logarithmic scaling.

To investigate the degradation of the networks due to imprecisions, we produce confusion matrices for both networks in the ideal case, with no imprecisions, and with different levels of error. $\sigma_{BS} = 1\%$, $\sigma_{PS} = 0.01\text{rad}$ and $\sigma_{BS} = 2\%$, $\sigma_{PS} = 0.02\text{rad}$ (Fig. 11).

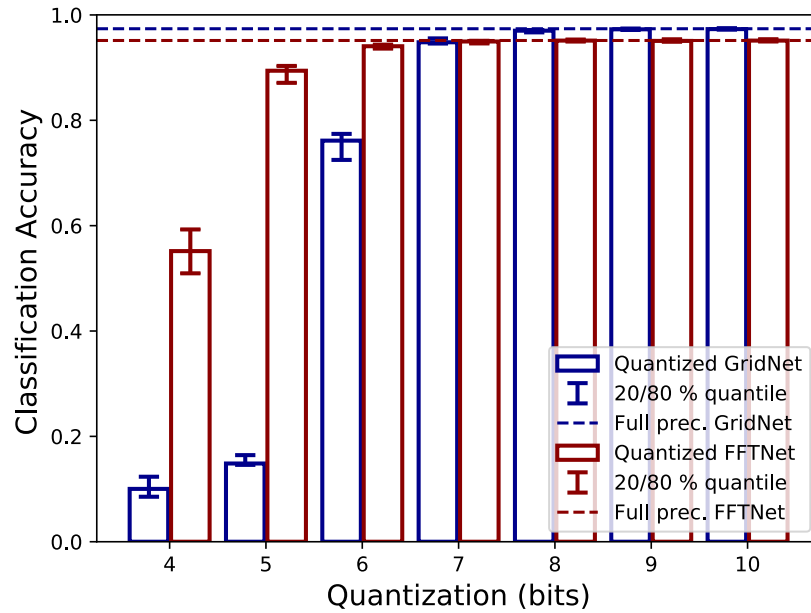


Fig. 12. The effects of quantization is shown for both GridNet and FFTNet. 10 instances of GridNet (blue) and FFTNet (red) were trained then quantized to varying levels. The mean classification accuracy at each level is shown by bar plots. The 20-80% quantiles are shown with error bars. The dotted horizontal line denotes the full precision accuracy.

The imprecisions were simulated 10 times and the mean of the output was used in generating the confusion matrices.

E. Quantization error

In this section, we explore the quantization error introduced by thermo-optic phaseshifters. Assuming a linear relationship between refractive index and temperature and quadratic relationship between temperature and voltage, we have

$$\begin{aligned}\theta &\propto V^2 \\ \theta &= 2\pi \left(\frac{V}{V_{2\pi}} \right)^2 \\ &\equiv 2\pi u^2 \\ \sqrt{\frac{\theta}{2\pi}} &= u.\end{aligned}$$

We have taken $V_{2\pi}$ to be the voltage required for a 2π phaseshift and defined the dimensionless voltage $u = V/V_{2\pi}$. Assuming that the voltage can be set with B -bit precisions, u must take on values of

$$u \in \{2^{-B}i : i = 0, \dots, 2^B - 1\}.$$

The quantization procedure then takes

$$\theta \rightarrow \tilde{\theta} \in \left\{ \frac{2\pi}{2^{2B}} i^2 : i = 0, \dots, 2^B - 1 \right\}.$$

To evaluate the sensitivity to quantization, we quantized GridNet and FFTNet with varying levels of precision. Since quantization is deterministic, we trained 10 instances of both networks with randomized initialization and thus different configuration but similar ideal accuracies ($\sim 95\%$ and $\sim 98\%$). The networks were then quantized at varying levels – from 4 to 10 bits. Their classification accuracy at each level is shown in Fig. 12.

Similar to results with simulated Gaussian noise, FFTNet is more robust than GridNet. Note that in this case, the quantization was applied after training has finished. Neural networks in which quantization happens as part of the training procedure has been demonstrated to have accuracies very near their full precision counterpart, down to even binary weights [49, 50].

F. Empirical distribution of phases

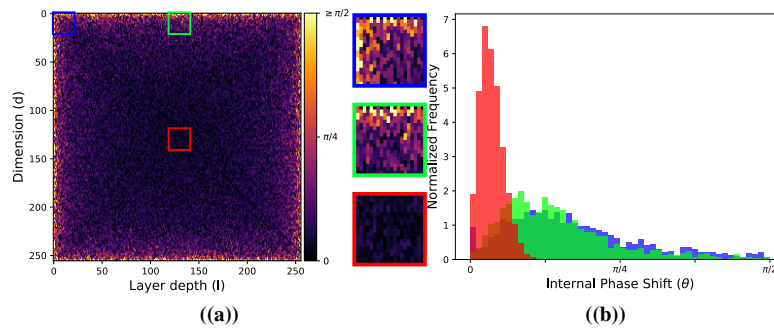


Fig. 13. The central MZIs of GridNet has lower variance in internal phase shifts (θ). a) The spatial distribution of internal phase shift ($\theta_{d,l}$) of MZIs in U_2 of GridNet. Reference Fig. 1(a) for coordinates and Fig. 2 for location of U_2 in context of network architecture. b) Histogram of phase shifts near the center (red), edge (green), and corner (blue) of the GridUnitary multiplier. These phases are obtained from multiple instances of trained GridNets with random initialization.

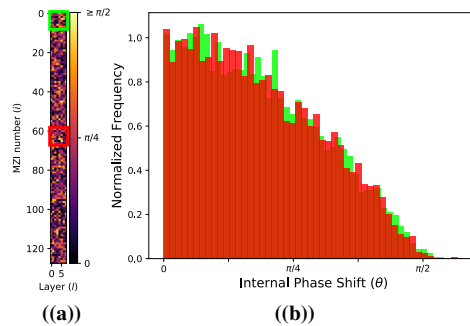


Fig. 14. The variance of internal phase shifts of FFTNet is uniform spatially (a) Spatial distribution of phase shifts for a FFTUnitary multiplier. The MZIs are ordered as shown in Fig. 1(b). (b) Histogram of phase shifts of FFTUnitary near the center (red) and top (green). These phases are obtained from multiple trained FFTNets with random initialization.

Analyses has been done on the distribution of the internal phase shift (θ) of MZIs of GridUnitary multipliers when used to implement randomly sampled unitary matrices [14–16]. It was shown that the phases are not uniformly distributed spatially. To be more concrete, We denote d the

waveguide number and l the layer number (see Fig. 1(a)). The distribution of the MZI reflectivity ($r = \sin(\theta/2)$) is [15]

$$r_{d,l} \sim \text{Beta}(1, \beta_{d,l}). \quad (33)$$

For large dimensions N ,

$$\beta \approx N - 2 \max(|d - N/2|, |l - N/2|) \quad (34)$$

$$= N - 2 \|(d, l) - (N/2, N/2)\|_{\infty}. \quad (35)$$

β decreases from N at the center of the grid layout to 0 at the edge of the grid. For large β (i.e. near the center), the mean and variance of $r_{d,l}$ are approximately

$$\mu_r \approx \beta^{-1}; \sigma_r^2 \approx \beta^{-2}.$$

Consequently, the reflectivity, and therefore the internal phases, of MZIs near the center of Grid Unitary multipliers are distributed very close to 0, with low variance. This effect is magnified with larger dimensions N .

This result was derived with the assumption of Haar-random unitary matrices. Such a distribution is not guaranteed and not expected for layers of trained neural networks. (Fig. 13(a)) shows the spatial distribution of phases in the GridUnitary multiplier U_2 (see Fig. 2). While the empirical histogram (Fig. 13(b)) does not match the theoretical distribution (Eq. (33)), the general trend of lower variance near the center of GridUnitary multipliers is evident. This is claimed to translates to a lower tolerance for error [14].

A similar analysis was conducted for FFTNet. Immediately we notice that the distribution of phase shifts is mostly uniform across the MZIs (Fig. 14(a)). This can be attributed to the non-local connectivity of FFTUnitary multipliers. Histograms constructed from an ensemble of 100 trained FFTNets with random initial weights (Fig. 14(b)) confirms this observation. The histogram for the region near the center (red) is nearly identical to the top (green).

We reiterate the distinction, made in Section 4.3, between pre-fabrication error tolerance and the sensitivity of error introduced post-fabrication. Pertinent to the first concept is how well the network can be optimized after a known set of imperfections are introduced to the network. The latter concept, which is relevant for our discussion, describes the sensitivity of the network with no further reconfiguration to unknown errors. In contrast to pre-fabrication error tolerance, our analysis in 4.3 does not show significant spatial dependence for post-fabrication error sensitivity.

G. BlockFFTNet

We introduce a network with similar depth as GridUnitary but with non-local, crossing waveguides in between as those seen in FFTUnitary (Fig. 15(a)). This is similar to the coarse-grained rectangular design mesh in [14] which was motivated to produce a spatially uniform distribution of phase and thus better tolerance for post-fabrication optimization. We also empirically observe that when incorporated as part of a ONN (BlockFFTNet), the distribution of phases are also uniformly distributed (Fig. 15(b)). We directly demonstrate that better tolerance for post-fabrication optimization does not directly to better error-resistance for a network optimized pre-fabrication. The accuracy loss due to increasing imprecision is shown in Fig. 16.

H. FFT algorithm and convolution

We show that the actual Cooley-Tukey FFT algorithm can be implemented with appropriate configurations of the phases of FFTUnitary multiplier.

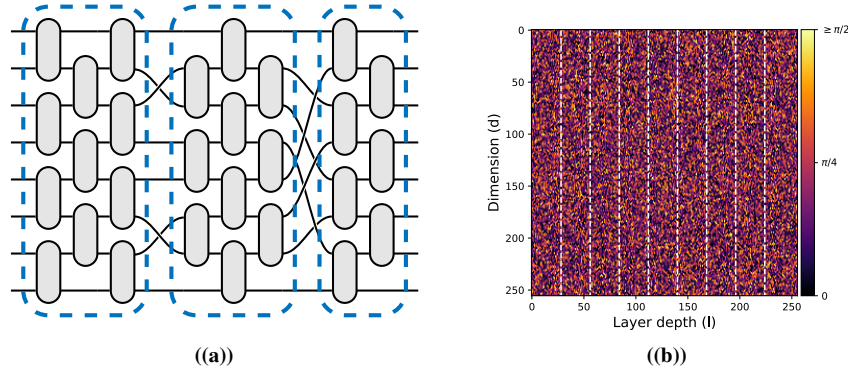


Fig. 15. a) A schematic of BlockFFTUnitary. Blocks of MZIs in dashed, blue boxes are similar to GridNet. The crossing waveguide, similar to those in FFTNet are between the blocks. b) The distribution of phases after being trained. The dashed white lines denote the locations of the crossing waveguides.

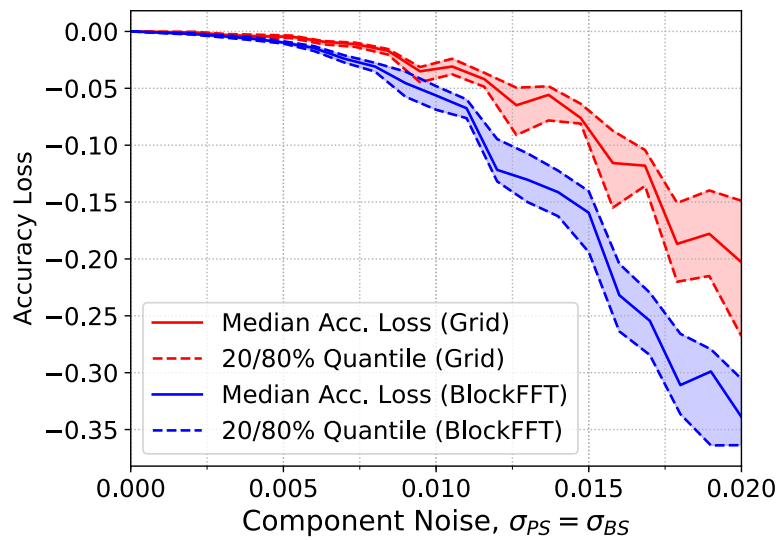


Fig. 16. No improvement in robustness to imprecision is seen with BlockFFTNet over GridNet. In fact, there is a significant decrease.

If we denote the input as $x_n \in \mathbb{C}^N$, its Fourier transform is

$$X_k = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} x_m e^{-\frac{2\pi i}{N} nk}. \quad (36)$$

The FFT algorithm, in short, is to rewrite the above as

$$X_k = \frac{1}{\sqrt{2}} \left(E_k + e^{-\frac{2\pi i}{N}k} O_k \right) \quad (37)$$

$$X_{k+N/2} = \frac{1}{\sqrt{2}} \left(E_k - e^{-\frac{2\pi i}{N}k} O_k \right). \quad (38)$$

Here, we have defined O_k and E_k to be the Fourier transform on the odd and even elements of x_n respectively. The calculation of E_k and O_k are done recursively. For $N = 2^K$, a total of K iterations are needed. It is well known that if x_n is in bit-reversed order, the calculations can be done in place.

Furthermore, in matrix form,

$$\begin{pmatrix} X_k \\ X_{k+N/2} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{-\frac{2\pi i}{N}k} \\ 1 & -e^{-\frac{2\pi i}{N}k} \end{pmatrix} \begin{pmatrix} E_k \\ O_k \end{pmatrix} \equiv U_k \begin{pmatrix} E_k \\ O_k \end{pmatrix}.$$

From Eq. (1), we note that $U_k = U_{MZ}(\theta = \pi/2, \phi = 2\pi k/N)$, up to some global phase. Therefore, if x_n is in bit-reversed order, and passed through a FFTUnitary multiplier where the k th layer is configured with $\theta = \pi/2, \phi = 2\pi k/N$, FFT can be performed.

Going further, a convolution can be easily performed through multiplication of the Fourier transformed signal by the Fourier transformed convolutional kernel, followed by an inverse Fourier transform.

Funding

MRD was partially supported by the U. S. Army Research Laboratory and the U. S. Army Research Office under contract W911NF-13-1-0390.

Code repository and results

The code repository, results and scripts used to generate figures in this paper are freely available at https://github.com/mike-fang/impresive_optical_neural_network

References

1. N. H. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical implementation of the hopfield model," *Appl. Opt.* **24**, 1469–1475 (1985).
2. Y. Paquot, F. Dupont, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar, "Optoelectronic reservoir computing," *Sci. Reports* **2**, 287 (2012).
3. L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, "Information processing using a single dynamical node as complex system," *Nat. Commun.* **2**, 468 (2011).
4. A. N. Tait, T. F. Lima, E. Zhou, A. X. Wu, M. A. Nahmias, B. J. Shastri, and P. R. Prucnal, "Neuromorphic photonic networks using silicon photonic weight banks," *Sci. Reports* **7**, 7430 (2017).
5. A. N. Tait, M. A. Nahmias, B. J. Shastri, and P. R. Prucnal, "Broadcast and weight: an integrated network for scalable photonic spike processing," *J. Light. Technol.* **32**, 3427–3439 (2014).
6. J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein, "Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification," *Sci. Reports* **8**, 12324 (2018).
7. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, and M. S. Englund, Dirk, "Deep learning with coherent nanophotonic circuits," *Nat. Photonics* **11**, 441 (2017).
8. I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1 (MIT Cambridge, 2016).
9. M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental realization of any discrete unitary operator," *Phys. Rev. Lett.* **73**, 58 (1994).
10. W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, "Optimal design for universal multiport interferometers," *Optica* **3**, 1460–1465 (2016).
11. R. Barak and Y. Ben-Aryeh, "Quantum fast fourier transform and quantum computation by linear optics," *JOSA B* **24**, 231–240 (2007).

12. J. Carolan, C. Harrold, C. Sparrow, E. Martín-López, N. J. Russell, J. W. Silverstone, P. J. Shadbolt, N. Matsuda, M. Oguma, and G. D. M. M. G. T. J. C. F. M. T. H. J. L. O. A. L. Itoh, Mikitaka, "Universal linear optics," *Science* **349**, 711–716 (2015).
13. N. C. Harris, G. R. Steinbrecher, M. Prabhu, Y. Lahini, J. Mower, D. Bunandar, C. Chen, F. N. Wong, T. Baehr-Jones, M. Hochberg, S. Lloyd, and D. Englund, "Quantum transport simulations in a programmable nanophotonic processor," *Nat. Photonics* **11**, 447 (2017).
14. S. Pai, B. Bartlett, O. Solgaard, and D. A. Miller, "Matrix optimization on universal unitary photonic devices," arXiv preprint arXiv:1808.00458 (2018).
15. N. J. Russell, L. Chakhmakchyan, J. L. O'Brien, and A. Laing, "Direct dialling of haar random unitary matrices," *New J. Phys.* **19**, 033007 (2017).
16. R. Burgwal, W. R. Clements, D. H. Smith, J. C. Gates, W. S. Kolthammer, J. J. Renema, and I. A. Walmsley, "Using an imperfect photonic network to implement random unitaries," *Opt. Express* **25**, 28236–28245 (2017).
17. D. A. Miller, "Perfect optics with imperfect components," *Optica* **2**, 747–750 (2015).
18. C. M. Wilkes, X. Qiang, J. Wang, R. Santagati, S. Paesani, X. Zhou, D. A. Miller, G. D. Marshall, M. G. Thompson, and J. L. O'Brien, "60 db high-extinction auto-configured mach-zehnder interferometer," *Opt. Lett.* **41**, 5318–5321 (2016).
19. M. Y.-S. Fang, "Imprecise optical neural networks," https://github.com/mike-fang/imprecise_optical_neural_network (2019).
20. J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Math. Comput.* **19**, 297–301 (1965).
21. Y. Ma, Y. Zhang, S. Yang, A. Novack, R. Ding, A. E.-J. Lim, G.-Q. Lo, T. Baehr-Jones, and M. Hochberg, "Ultralow loss single layer submicron silicon waveguide crossing for soi optical interconnect," *Opt. Express* **21**, 29374–29382 (2013).
22. R. R. Gattass and E. Mazur, "Femtosecond laser micromachining in transparent materials," *Nat. Photonics* **2**, 219 (2008).
23. G. Panusa, Y. Pu, J. Wang, C. Moser, and D. Psaltis, "Photoinitiator-free multi-photon fabrication of compact optical waveguides in polydimethylsiloxane," *Opt. Mater. Express* **9**, 128–138 (2019).
24. M. J. Connelly, *Semiconductor Optical Amplifiers* (Springer Science & Business Media, 2007).
25. D. A. Miller, "Silicon photonics: Meshing optics with applications," *Nat. Photonics* **11**, 403 (2017).
26. Q. Bao, H. Zhang, Z. Ni, Y. Wang, L. Polavarapu, Z. Shen, Q.-H. Xu, D. Tang, and K. P. Loh, "Monolayer graphene as a saturable absorber in a mode-locked laser," *Nano Res.* **4**, 297–307 (2011).
27. V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, (2010), pp. 807–814.
28. M. Arjovsky, A. Shah, and Y. Bengio, "Unitary evolution recurrent neural networks," in *International Conference on Machine Learning*, (2016), pp. 1120–1128.
29. Q. Xu and M. Lipson, "Optical bistability based on the carrier dispersion effect in soi ring resonators," in *Integrated Photonics Research and Applications*, (Optical Society of America, 2006), p. IMD2.
30. Y. Jiang, P. T. DeVore, and B. Jalali, "Analog optical computing primitives in silicon photonics," *Opt. Lett.* **41**, 1273–1276 (2016).
31. M. Babaeian, P.-A. Blanche, R. A. Norwood, T. Kaplas, P. Keiffer, Y. Svirko, T. G. Allen, V. W. Chen, S.-H. Chi, J. W. Perry, "Nonlinear optical components for all-optical probabilistic graphical model," *Nat. Commun.* **9**, 2128 (2018).
32. Y. LeCun, "The mnist database of handwritten digits," <http://yann.lecun.com/exdb/mnist/>.
33. A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, (2017).
34. C. Dugas, Y. Bengio, F. Bélisle, C. Nadeau, and R. Garcia, "Incorporating second-order functional knowledge for better option pricing," in *Advances in neural information processing systems*, (2001), pp. 472–478.
35. T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)* (Wiley-Interscience, New York, NY, USA, 2006).
36. L. Jing, Y. Shen, T. Dubcek, J. Peurifoy, S. Skirlo, Y. LeCun, M. Tegmark, and M. Soljačić, "Tunable efficient unitary neural networks (eunn) and their application to rnns," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, (JMLR. org, 2017), pp. 1733–1741.
37. C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal, "Deep complex networks," arXiv preprint arXiv:1705.09792 (2017).
38. H. Robbins and S. Monro, "A stochastic approximation method," in *Herbert Robbins Selected Papers*, (Springer, 1985), pp. 102–109.
39. P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *null*, (IEEE, 2003), p. 958.
40. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
41. F. Flamini, N. Spagnolo, N. Viggianiello, A. Crespi, R. Osellame, and F. Sciarrino, "Benchmarking integrated linear-optical architectures for quantum information processing," *Sci. Reports* **7**, 15133 (2017).
42. F. Flamini, L. Magrini, A. S. Rab, N. Spagnolo, V. D'ambrosio, P. Mataloni, F. Sciarrino, T. Zandrini, A. Crespi, R. Ramponi, and R. Osellame, "Thermally reconfigurable quantum photonic circuits at telecom wavelength by

- femtosecond laser micromachining,” *Light. Sci. & Appl.* **4**, e354 (2015).
43. D. F. Walls and G. J. Milburn, *Quantum optics* (Springer Science & Business Media, 2007).
 44. A. Fawzi, S.-M. Moosavi-Dezfooli, and P. Frossard, “Robustness of classifiers: from adversarial to random noise,” in *Advances in Neural Information Processing Systems*, (2016), pp. 1632–1640.
 45. L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, “Regularization of neural networks using dropconnect,” in *International Conference on Machine Learning*, (2013), pp. 1058–1066.
 46. K. Kikuchi, “Characterization of semiconductor-laser phase noise and estimation of bit-error rate performance with low-speed offline digital coherent receivers,” *Opt. Express* **20**, 5291–5302 (2012).
 47. M. Larson, Y. Feng, P.-C. Koh, X.-d. Huang, M. Moewe, A. Semakov, A. Patwardhan, E. Chiu, A. Bhardwaj, K. Chan *et al.*, “Narrow linewidth high power thermally tuned sampled-grating distributed bragg reflector laser,” in *2013 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC)*, (IEEE, 2013), pp. 1–3.
 48. A. Selden, “Pulse transmission through a saturable absorber,” *Br. J. Appl. Phys.* **18**, 743 (1967).
 49. I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, “Quantized neural networks: Training neural networks with low precision weights and activations,” *The J. Mach. Learn. Res.* **18**, 6869–6898 (2017).
 50. M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, “Xnor-net: Imagenet classification using binary convolutional neural networks,” in *European Conference on Computer Vision*, (Springer, 2016), pp. 525–542.