



OPEN Automating multi-task learning on optical neural networks with weight sharing and physical rotation

Shanglin Zhou^{1,5}, Yingjie Li^{2,5}, Weilu Gao³, Cunxi Yu² & Caiwen Ding⁴✉

The democratization of AI encourages multi-task learning (MTL), demanding more parameters and processing time. To achieve highly energy-efficient MTL, Diffractive Optical Neural Networks (DONNs) have garnered attention due to extremely low energy and high computation speed. However, implementing MTL on DONNs requires manually reconfiguring & replacing layers, and rebuilding & duplicating the physical optical systems. To overcome the challenges, we propose LUMEN-PRO, an automated MTL framework using DONNs. We first propose to automate MTL utilizing an arbitrary backbone DONN and a set of tasks, resulting in a high-accuracy multi-task DONN model with small memory footprint that surpasses existing MTL. Second, we leverage the rotability of the physical optical system and replace task-specific layers with rotation of the corresponding shared layers. This replacement eliminates the storage requirement of task-specific layers, further optimizing the memory footprint. LUMEN-PRO provides flexibility in identifying optimal sharing patterns across diverse datasets, facilitating the search for highly energy-efficient DONNs. Experiments show that LUMEN-PRO provides up to 49.58% higher accuracy and 4× better cost efficiency than single-task and existing DONN approaches. It achieves memory lower bound of MTL, with memory efficiency matching single-task models. Compared to IBM-TrueNorth, LUMEN-PRO achieves an 8.78× energy efficiency gain, while it matches Nanophotonic in efficiency but surpasses it in per-operator efficiency due to its larger system.

The increasing accessibility of AI encourages the concurrent execution of multiple interrelated tasks, i.e., more than one model simultaneously on a single resource-constrained device. This presents challenges due to the escalated computation, energy, and storage costs¹. To tackle this, multi-task learning (MTL) provides a promising solution by facilitating joint learning of a task set and enabling parameter sharing to reduce costs. In general, MTL has high model complexity. For example,² shows their MTL model is 4.5× slower at inference and requires 2.4× more parameters than the single-task model for depth estimation when using the same backbone network. The MTL model in³ also shows ~ 2× more parameters than the single-task model and requires ~ 4× longer to train.

On the other hand, with the demise of Moore's law, the demand for efficient deep neural network accelerators has surged⁴. To achieve real-time and ultra-low power DNN processing, advanced device/circuit technologies surpassing the Complementary metal-oxide-semiconductor (CMOS) technology are essential. As a representative, the Diffractive Optical Neural Network (DONN) has emerged to overcome the energy efficiency drawbacks associated with CMOS-based DNN systems⁵. The all-optical processing capabilities of DONNs are achieved by leveraging inherent physical phenomena, such as light diffraction and light signal phase modulation, occurring naturally at the speed of light. It offers (i) significantly less energy and thermal constraints, and more bandwidth compared to CMOS-based systems; and (ii) remarkable computational speed, transmitting information at the speed of light^{6,7}.

To realize highly energy efficient MTL, DONN technology appears to be a highly compelling and natural candidate, as evidenced by VanillaMT⁸ and RubikONN⁹. Despite the success, the integration of MTL with DONN is still challenging due to the need (i) to rebuild and duplicate the physical hardware of the system, leading to

¹School of Computing, University of Connecticut, Storrs 06269, USA. ²A. James Clark School of Engineering, University of Maryland, College Park 20742, USA. ³Electrical and Computer Engineering, University of Utah, Salt Lake City 84112, USA. ⁴Department of Computer Science & Engineering, University of Minnesota Twin Cities, Minneapolis 55455, USA. ⁵Shanglin Zhou and Yingjie Li contributed equally to this work. ✉email: dingc@umn.edu

energy disadvantage and cost inefficiency; (ii) of domain knowledge when designing resource-efficient MTL models, as substantial exploration efforts are needed to determine task-specific elements that shared across tasks. In this work, we propose an extremely energy-efficient automating multi-task learning framework on optical neural networks, LUMEN-PRO. LUMEN-PRO utilizes rotatability of the physical system and takes an arbitrary DONN backbone and a set of vision tasks as inputs, where the backbone defines the computation graph with layers functioning as shared operator nodes. The difference between our method and state-of-the-arts (SOTAs) are summarized in Table 1. Our contributions are summarized as follows:

- LUMEN-PRO automates the transformation of a user-provided DONN backbone into an operator-level supermodel, enabling gradient-based architecture search for efficient task sharing and cost optimization.
- LUMEN-PRO leverages the rotatability of the physical system to fine-tune the multi-task DONN architecture for resource efficiency. The task-specific layers are replaced by the physical rotation of the shared layers, therefore requiring zero memory footprint. Our method achieves a memory lower bound of multi-task learning, i.e., having the same memory storage as the single-task model.
- LUMEN-PRO can greatly reduce the cost and energy that required for MTL DONN applications, while still maintaining high prediction accuracy.

Experimental results show that LUMEN-PRO framework achieves up to 13.51% higher accuracy and 4× better cost efficiency than single-task and existing DONN baselines on MNIST family, with an improvement in prediction accuracy of up to 49.58% on the CelebA. LUMEN-PRO achieves at least 9.6× and 8.78× energy efficiency gain compared to ReRAM-based and ASIC-based implementations, respectively, while matching photonic CNNs in efficiency but surpassing them in per-operator efficiency due to its larger system.

Results
Training and Evaluation Setup

Datasets and evaluation metrics
We evaluate the performance of LUMEN-PRO on two popular multi-task learning datasets. The first is MNIST family, which consists of four public image classification datasets: MNIST-10 (MNIST)¹⁰, Fashion-MNIST (FMNIST)¹¹, Kuzushiji-MNIST (KMNIST)¹², and Extension-MNIST-Letters (EMNIST)¹³. For EMNIST, we customize the dataset by selecting the first ten classes “A-J”. The second used is CelebFaces Attributes Dataset (CelebA)¹⁴. We choose four attributes with relatively more balanced labels, namely *Smiling*, *Mouth_Slightly_Open*, *Male*, and *Attractive*. We transform all images to grayscale to ensure compatibility with DONN physical system. F_1 -score and accuracy are used as evaluation metrics for CelebA dataset.

DONN system parameters and training setup
We utilize a system configured with five diffractive layers, each of dimension 200×200 , hence both the layer and the detector regions share these dimensions. The system is configured in 532nm laser wavelength (green laser). Original input images in the evaluated datasets are interpolated to 200×200 to align with our optical system and encoded. We maintain a uniform physical distance of 27.94cm among layers, between the first layer and the source, and from the final layer to the detector. The distinct detector regions, corresponding to the number of classes, are uniformly situated on the detector plane, each sized 20×20 . The aggregate intensity of the detector regions equates will return a vector in float32 type. The final prediction results are computed using argmax. During the training process, we use learning rate as 0.1 under Adam optimizer, batch-size being 200. StepLR scheduler is adopted with step-size being 6000. All the implementations are constructed using PyTorch v1.8.1. Experiments are conducted on a Nvidia Quadro RTX6000.

Evaluation result
Accuracy comparison

We compare LUMEN-PRO with four baselines: (1) a single-task baseline with independent models for each task, (2) *BaselineMT*¹⁵, a straightforward multi-task approach using a fixed single-task DONN architecture by merging datasets, (3) *VanillaMT*⁸, a specific MTL DONN method with shared backbone and separate diffractive layers at the output stage, and (4) *RubikONN*⁹, a state-of-the-art MTL DONN method with weight aggregation and rotation but manual architecture design. We select the best combination of rotated layers and rotation angles for RubikONN. To ensure fair comparisons, we use the same backbone DONN model across all baselines and LUMEN-PRO.

MNIST Family Figure 1 compares the accuracy of LUMEN-PRO with and without rotation to baseline methods on four MNIST family datasets. We use MNIST family to align with the baselines. Results show that LUMEN-PRO outperforms all MTL baselines, especially BaselineMT, with improvements of 4.42%, 6.23%,

Methods	Multi-task	Automatic	Rotation
VanillaMT	✓	×	×
RubikONN	✓	×	✓
(Ours) LUMEN-PRO	✓	✓	✓

Table 1. Comparison of different MTL methods.

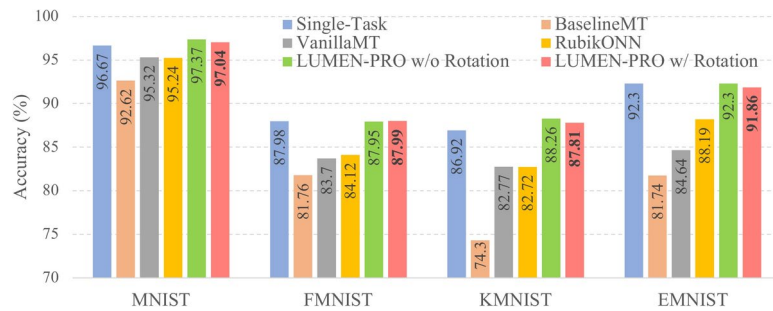


Fig. 1. Comparison with baselines on MNIST Family.

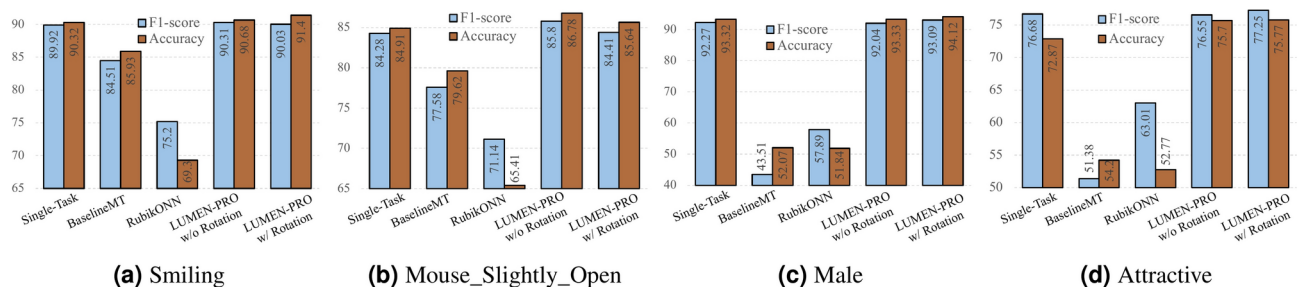


Fig. 2. Comparison of single-task and MTL baselines with LUMEN-PRO on CelebA dataset.

13.51%, and 10.12% in MNIST, FMNIST, KMNIST, and EMNIST datasets, respectively. LUMEN-PRO with rotation achieves significant accuracy gains of 1.8%, 3.87%, 5.09%, and 3.67% compared to RubikONN. Compared to the single-task baselines, LUMEN-PRO with rotation demonstrates accuracy improvements of 0.37% (MNIST) and 0.89% (KMNIST). Notably, there is minimal difference in performance between LUMEN-PRO with and without rotation, and rotation does not incur additional memory usage on the physical system for the multi-task DONN model.

We analyze the learned sharing policies of the multi-task DONN architecture. Figure 3 displays the sampled feature sharing pattern for the four datasets. Earlier diffractive layers tend to have dataset-specific weight selection, while operator sharing primarily occurs in later layers. FMNIST consistently shares across all five diffractive layers. These results contrast with the architecture design from RubikONN, where the first three diffractive layers are shared across tasks, and the fourth and fifth layers are task-specific and rotated from the backbone operators at pre-designed angles.

CelebA Figure 2 compares LUMEN-PRO to baselines in a four-task MTL with CelebA attributes. LUMEN-PRO outperforms the single-task model in all tasks, indicating task correlations that enable our MTL DONN model to leverage shared information and learn common features, improving overall performance. With rotation, LUMEN-PRO outperforms BaselineMT by up to 49.58% and RubikONN by up to 35.2%. Unlike BaselineMT and RubikONN, LUMEN-PRO can automatically explore and tailor architectures for data and task specifics, enhancing correlation and dissimilarity capturing for improved performance across tasks.

Figure 3b provides insights into the architecture of the multi-task DONN model designed by LUMEN-PRO for CelebA baseline. In the initial layers, shared operators are utilized, whereas rotated task-specific operators are employed starting from the fourth layer onwards. Our approach consistently outperforms RubikONN-designed models, accommodating task heterogeneity effectively. The architecture for CelebA differs significantly from the MNIST-based dataset, demonstrating the flexibility of our automated feature in LUMEN-PRO for diverse tasks and datasets.

Accuracy-cost comparison

In this part, we compare the model efficiency of LUMEN-PRO in terms of system cost and accuracy with the MTL baselines and single-task models. We use the “Accuracy-Cost evaluation metric” in Eq. 1 for this evaluation⁹.

$$E_{acc-cost} = \frac{\text{Accuracy}_{single-task}}{\text{Accuracy}_{MTL}} \times \frac{\text{F-Cost}_{MTL}}{\text{F-Cost}_{single-task}} \quad (1)$$

Here, F-cost represents the fabrication cost for the diffractive layers and the detectors in the hardware implementation, where the layer fabrication cost is ~\$100 and detector cost is \$1500–\$10,000. We normalize the cost of \$100 as unit 1, thus, the layer cost for a 5-layer ONN is 5 and one detector cost is 10 for the cost efficiency estimation.

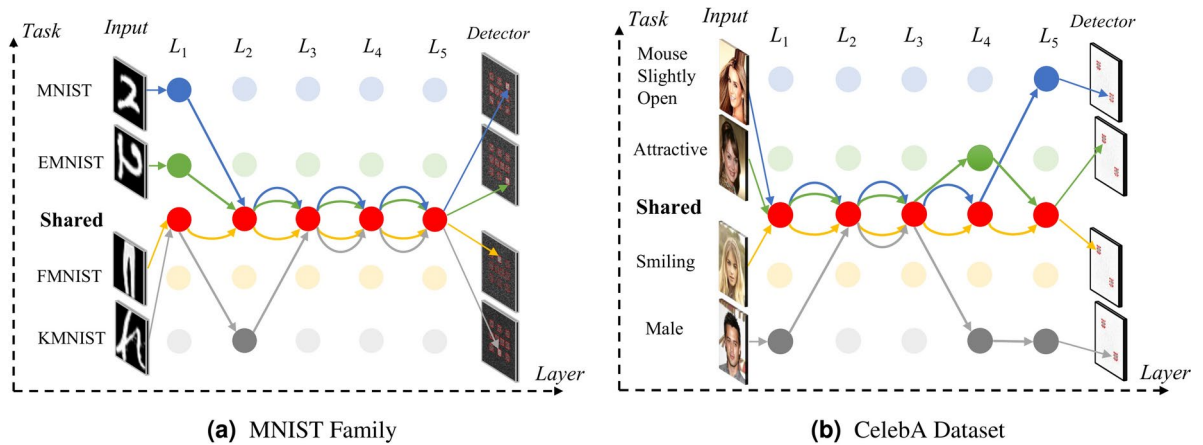


Fig. 3. Policy Visualization for (left) MNIST Family and (right) CelebA dataset. Weights on non-red nodes derived via rotational transformation based on red (shared) node weights. Semi-transparent nodes mean the operators are not selected.

Table 2 compares the accuracy and model efficiency of LUMEN-PRO with single-task models and MTL baselines. The model efficiency of single-task models is normalized to a unit value of 1, which is used to calculate the improvement achieved by the MTL method using Eq. 1. The sharing ratio indicates the proportion of shared layers among all four tasks. Both VanillaMT and RubikONN share the first three layers and diverge at the last two, with RubikONN reusing the rotation layers. As shown, LUMEN-PRO with rotation surpasses the single-task model and VanillaMT with $4\times$ and $2\times$ model efficiency improvements, respectively. It has a 50% higher sharing ratio than VanillaMT, reducing fabrication costs during deployment. Compared to RubikONN, LUMEN-PRO consistently achieves higher model efficiency and accuracy across all tasks, demonstrating its effectiveness in diverse datasets and real-world applications.

Comparing our LUMEN-PRO with and without rotation algorithm, we find that without rotation, LUMEN-PRO achieves higher accuracy but lacks the cost-saving benefits of reusing middle layers. This highlights the importance of balancing accuracy and cost in real-world deployment. By incorporating the rotation algorithm, LUMEN-PRO enables high-performing structures that excel in both accuracy and model efficiency, improving practical applicability.

Energy efficiency comparison In Table 3, we provide the comparison results on accuracy, throughput, power, and energy efficiency on MNIST. We select the state-of-the-art extremely energy efficiency implementation from other technologies (e.g., FPGA, ASIC, ReRAM) as baselines, including FINN¹⁸ (binary neural network (BNN)), IBM TrueNorth¹⁹ (spiking neural network (SNN)), FORMS²⁰, ISAAC²¹, DaDianNao²², and Holylight¹⁶ (photonic CNN). The energy estimation of LUMEN-PRO is based on the practical realization of the DONN system in¹⁷. The power consumption of a continuous-wave 532nm laser source is $\sim 5\text{mW}$. The diffractive layers are passive optical devices and require no extra energy for computation. Theoretically, the detector refreshing frequency can reach up to 10^4kFPS estimated by Holylight¹⁶ with the power consumption of 68.3W. As a result, the total power consumption for LUMEN-PRO is 68.31W, while the throughput of the system is $1 \times 10^4\text{kFPS}$.

Compared to the reference FPGA-based implementation at a similar accuracy level, we achieve $2.1\times$ energy efficiency gain. In comparison to ReRAM-based and ASIC-based implementations, the energy efficiency gain is at least $9.6\times$ and $8.78\times$ respectively. Additionally, while the energy efficiency of LUMEN-PRO is comparable to that of photonic CNNs, LUMEN-PRO demonstrates superior energy efficiency per operator theoretically, as LUMEN-PRO operates with a system size of 200×200 , compared to the 28×28 frame size used in Holylight.

Methods

DONN Preliminary

DONN system is designed with three major components (Fig. 4): (1) laser source encoding the input images, (2) diffractive layers encoding trainable phase modulation, and (3) detectors capturing the output of the forward propagation. The input image is first encoded with the laser source. The information-encoded light signal is diffracted in the free space between diffractive layers and modulated via phase modulation at each layer. Finally, the diffraction pattern after light propagation w.r.t light intensity distribution will be captured at the detector plane for predictions.

First, the input information (e.g., an image) is encoded with the coherent light signal from the laser source, and the information-encoded wavefunction is $f^0(x_0, y_0)$. The wavefunction after light diffraction from the input plane to the first diffractive layer over diffraction distance z can be seen as the summation of the outputs at the input plane, i.e.,

$$f^1(x, y) = \iint f^0(x_0, y_0) h(x - x_0, y - y_0, z) dx_0 dy_0 \quad (2)$$

	Single-task			VanillaMT				RubikONN				LUMEN-PRO w/o Rotation				LUMEN-PRO w/ Rotation				
	Mnst	Fmst	Kmst	Emst	Mnst	Fmst	Kmst	Emst	Mnst	Fmst	Kmst	Emst	Mnst	Fmst	Kmst	Emst	Mnst	Fmst	Kmst	Emst
Task																				
Accuracy (%)	96.4	86.5	86.1	90.9	95.3	83.7	82.8	84.6	95.2	84.1	82.7	88.2	97.4	87.9	88.3	92.3	97.0	88.0	87.8	91.7
Layer Cost (Norm.)	5	5	5	5	11				5				8				5			
Detector Cost (Norm.)	10	10	10	10	20				10				10				10			
Model Eff. (Norm.)	1	1	1	1	1.8×	1.76×	1.75×	1.69×	3.95×	3.89×	3.84×	3.88×	2.48×	2.46×	2.44×	2.46×	4.03×	4.07×	4.08×	4.04×
Sharing Ratio	-				12/20				16/20				16/20				17/20			

Table 2. Accuracy-Cost trade-off comparison of MNIST family: MNIST (Mnst), FMNIST (Fmst), KMNIST (Kmst), EMNIST (Emst). LUMEN-PRO follows the model structure shown in Fig. 3a.

	FINN	TrueNorth	FORMS	ISAAC	DaDianNao	Hollylight	LUMEN-PRO
Technology	FPGA	ASIC	ReRAM	ReRAM	ASIC	Photonic	Free-space Optics
Network Type	BNN	SNN	CNN	CNN	CNN	CNN	DONN
Accuracy (%)	98.4	95	99.17	99.1	99.18	98.9	97
Throughput (kFPS)	1561	1.0	500	$1 \times 10^3^*$	100*	$1 \times 10^4^*$	1×10^4
Power (Watt)	22.6	0.06	66.36	65.8	20.1	68.3	68.31
Energy Eff. (kFPS/W)	69.07	16.67	7.53	15.2	4.98	146.4	146.4**

Table 3. Comparison on accuracy, performance, energy efficiency of LUMEN-PRO and baselines. *Our retrieved numbers from the histograms in¹⁶, to the best of our knowledge. **The energy efficiency is calculated with the hardware used in the visible range setups in¹⁷ with high-end detector in¹⁶

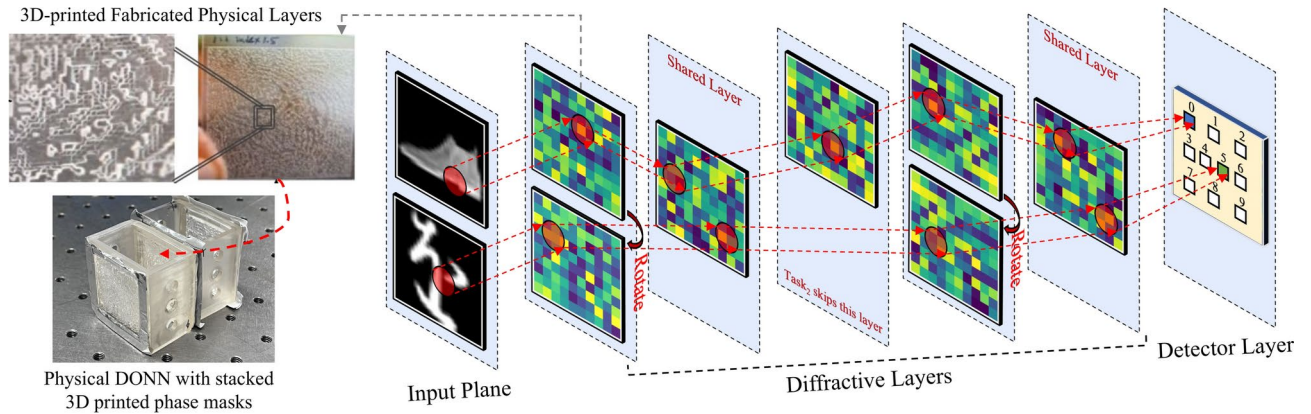


Fig. 4. A multi-task DONN framework for two-tasks classification.

where (x, y) is the coordinate on the receiver plane, i.e., the first diffractive layer, h is the impulse response function of free space. Here we use Fresnel approximation, thus the impulse response function h is as Eq. 3, where $i = \sqrt{-1}$, λ is the wavelength of the laser source, $k = 2\pi/\lambda$ is free-space wavenumber.

$$h(x, y, z) = \frac{\exp(ikz)}{i\lambda z} \exp\left\{\frac{ik}{2z}(x^2 + y^2)\right\} \quad (3)$$

Equation 2 can be calculated with spectral algorithm, where we employ Fast Fourier Transform (FFT) for fast and differentiable computation, i.e., $U^1(\alpha, \beta) = U^0(\alpha, \beta)H(\alpha, \beta, z)$, where U and H are the Fourier transformation of f and h .

After light diffraction, the wavefunction resulting $U^1(\alpha, \beta)$ is first transformed to time domain with inverse FFT (iFFT). Then the phase modulation $W(x, y)$ provided by the diffractive layer is applied to the light wavefunction in time domain by matrix multiplication, i.e., $f^2(x, y) = \text{iFFT}(U^1(\alpha, \beta)) \times W_1(x, y)$, where $W_1(x, y)$ is the phase modulation in the first diffractive layer, $f^2(x, y)$ is the input light wavefunction for the light diffraction between the first diffractive layer and the second diffractive layer. The computation module with one computation round of light diffraction and phase modulation at one diffractive layer is named **DiffMod**, i.e., $\text{DiffMod}(f(x, y), W) = L(f(x, y), z) \times W(x, y)$, where $f(x, y)$ is the input wavefunction, $W(x, y)$ is the phase modulation, $L(f(x, y), z)$ is the wavefunction after light diffraction over a constant distance z in time domain, i.e., $\text{iFFT}(U(\alpha, \beta))$.

The forward function for a multiple-diffractive-layer-constructed DONN system is computed iteratively for the stacked diffractive layers. For example, the forward function for the 5-layer system shown in Fig. 4 can be expressed as,

$$I(f^0(x, y), W) = \text{DiffMod}(\text{DiffMod}(\text{DiffMod}(\text{DiffMod}(\text{DiffMod}(f^0(x, y), W_1(x, y)), W_2(x, y)), W_3(x, y)), W_4(x, y)), W_5(x, y)) \quad (4)$$

where $f^0(x, y)$ is the input wavefunction to the system and W_{1-5} is phase modulation provided at each diffractive layer.

The final diffraction pattern w.r.t the light intensity I as denoted in Eq. 4 is projected onto the plane of the detector. By defining the coordinates of the detector region across the entire detector plane for each class according to the user's specifications, it becomes feasible to devise diverse detector patterns for various tasks. As an illustration, in the case of FMNIST datasets, the output plane is partitioned into **ten** distinct detector regions to emulate the outcomes of conventional neural networks that predict **ten** classes. The classification result is

determined by employing the argmax function on the sums of intensities from the ten detector regions. For instance in Fig. 4, by examining the label indices of the ten detector regions corresponding to the image “boots”, the highest energy is observed in the first region of the first row. Consequently, the predicted class is “0”. By utilizing the one-hot encoded representation of the ground truth class denoted as t , the loss function L can be obtained through the utilization of **MSELoss**, i.e., $L = \|\text{Softmax}(I) - t\|_2$. Thus, the whole system is designed to be differentiable and compatible with conventional automatic differential engines.

LUMEN-PRO framework

We aim to develop a precise, cost-efficient automating multi-task learning DONN system by sharing parameters across tasks. Diffractive layers in DONN systems are typically 3D printed and feature permanent phase parameters once fabricated. However, their square shape allows for relocating and rotating the layers, enabling weight alterations and modification of the DONN system’s forward function¹⁵. This rotation changes light modulation patterns and enhances the performance and computational efficiency of the MTL DONN system. Figure 5 provides an overview of our LUMEN-PRO framework.

Automating multi-task learning framework

The multi-task supermodel is generated from a backbone DONN as shown in Fig. 5, encoding the search space for all tasks. A gradient-based architecture search algorithm is employed to find the optimal architecture, ensuring accuracy and compactness. Task-specific aspects are then addressed, where task-specific copies replace weights with rotated weights from the shared operator of the backbone model. This approach tailors the model for each task without increasing energy and cost compared to a single-task model.

Supermodel and search space As shown in Fig. 6, we (i) start with initializing a backbone model using a single-task architecture. For each task t , we (ii) create two branches (a task-specific copy of the basic shared operator (spc^t) and a skip operator (skp^t)) for each node, in parallel with the basic shared operator (bsc). Each branch (skp^t or spc^t) has the same tensor dimension as bsc . A variable/policy P^t determines which branch will be executed for task t . We (iii) iteratively update the weights and policies. We utilize a three-layer DONN as an example, where each layer is represented by a 2×2 matrix. To represent the DONN backbone model as a computation graph, we employ nodes to symbolize the layers. The notation $node_i$ is used to refer to layer i within the backbone model. In order to transition the single-task backbone into a multi-task model, we enhance each node as a computation unit (CU). If there are N tasks in the design, each CU would consist of N blocks, with each block (CU_t) comprising three operators, i.e., bsc , skp^t , and spc^t . All these CUs form our multi-task supermodel.

Gradient-based architecture search Our aim is to discover the most effective sharing policy for a multi-task supermodel that results in the top-performing outcome across all tasks. To effectively explore the search space and identify the ideal sharing policy for a multi-task DONN supermodel, a gradient-based architecture search algorithm is employed²³. This algorithm optimizes the sharing policy and multi-task DONN model parameters simultaneously using standard back-propagation. The non-differentiability and discrete nature of policy variables are handled using the Gumbel-Softmax Approximation²⁴ and soft differentiable policy as Eq. 5.

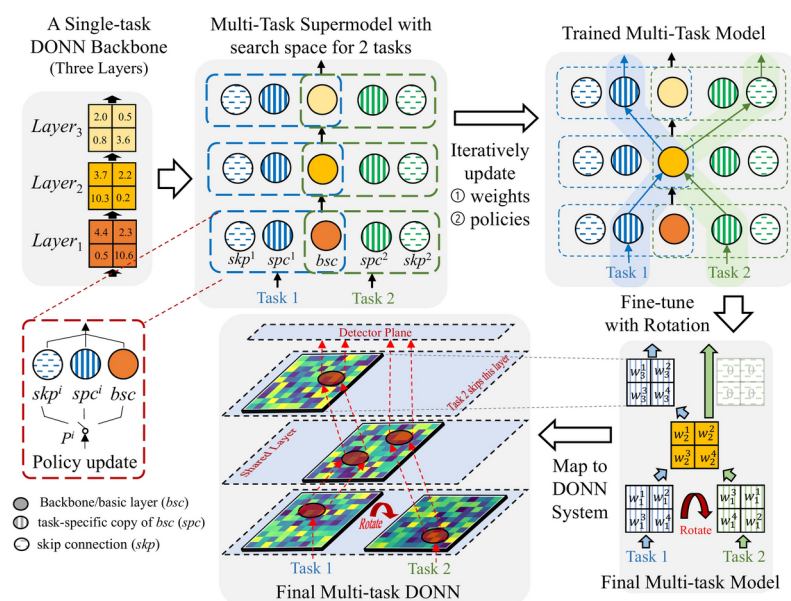


Fig. 5. Overview of LUMEN-PRO framework on DONN system.

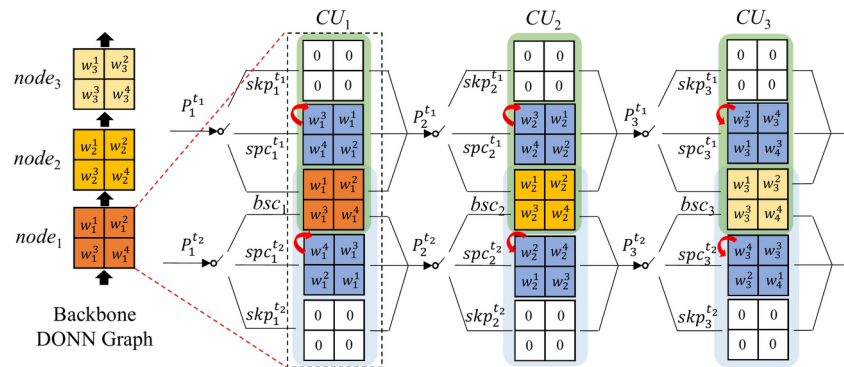


Fig. 6. Computational graph of LUMEN-PRO.

$$P'(k) = \frac{\exp((G_k + \log(\pi_k)) / \tau)}{\sum_{k \in \{0,1,2\}} \exp((G_k + \log(\pi_k)) / \tau)} \quad (5)$$

here, P is the policy variable; k represents the three operator options that 0 is the backbone basic shared operator bsc is chosen for the task, 1 is the rotated task-specific copy spc^t is adopted, and 2 is the skip operator skp^t is selected; $G_k \sim \text{Gumbel}(0, 1)$. Once the distribution π is learned, we sample the discrete task-specific policy P , which determines the operator to execute in each CU for each task. Using this policy, we construct the multi-task DONN architecture, ensuring better performance among all the tasks.

To further optimize the energy and cost overhead, sharing operators across tasks are more encouraged in the multi-task DONN model. Denote the probability of selecting the basic shared operator, the task-specific copy, and the skip operator as $P_n^{t_m}(0)$, $P_n^{t_m}(1)$, and $P_n^{t_m}(2)$ for the m -th task in CU_n , a policy regularization term L_{reg}^{25} is added to the loss function as Eq. 6. Here, T is the total number of tasks, and N is the total number of diffractive layers in the backbone model.

$$L_{reg} = \sum_{m \leq T} \sum_{n \leq N} \frac{N-n}{N} \left\{ \ln \left(1 + e^{P_n^{t_m}(1) - P_n^{t_m}(0)} \right) + \ln \left(1 + e^{P_n^{t_m}(2) - P_n^{t_m}(0)} \right) \right\} \quad (6)$$

In this case, the final loss function can be written as Eq. 7. Here, L_m refers to the loss for each task, α_m and α_{reg} are regularization factors.

$$\mathcal{L} = \sum_{m \leq T} \alpha_m \cdot L_m + \alpha_{reg} \cdot L_{reg} \quad (7)$$

Physical layer rotation algorithm

We use a three-stage training pipeline for the multi-task DONN model. In the first stage (Fig. 5), pre-training is conducted jointly on all tasks to obtain a well-initialized multi-task DONN supermodel. The output of each CU for each task is the average of the backbone shared operator, task-specific copy, and skip connection, ensuring parameter warming. The second stage is policy-training, optimizing the sharing policy and model parameters iteratively. Once the policy distribution parameters converge, a sharing policy is sampled to generate the multi-task DONN model. In the post-training stage, the sharing policy is fixed, and model parameters are trained from scratch. We use the rotation training algorithm to leverage the inherent physical rotation properties of optical systems for MTL with minimal overhead.

After the sampling stage, the structure of the multi-task model is finalized, and operator selection for each node in each task is determined. As in Algorithm 1, during the post-training phase, two models are initialized: one for aggregation and another as a virtual model to temporarily store updates for specific rotation patterns and tasks. The virtual model is re-initialized with either the initial weight parameters or the parameters optimized in the previous iteration. As in Fig. 7a, during each training iteration, nodes in certain tasks may choose the shared backbone operator or the task-specific copy in a specific layer of the multi-task DONN model. The weights in the task-specific copies are then replaced with varying degrees of rotation using the weights from the backbone shared operator (lines 3–5). After rotations and substitutions, the parameters in the virtual model are updated (line 6). After one training iteration as in Fig. 7b, all substituted weights in the virtual model are reverse-rotated back to their initial position (line 7–8) as in Fig. 7c. Aggregation is then performed by averaging weights of nodes across tasks in the same layer (lines 9–11), and new weights are copied to re-initialize the aggregation model (line 12–13) as in Fig. 7d.

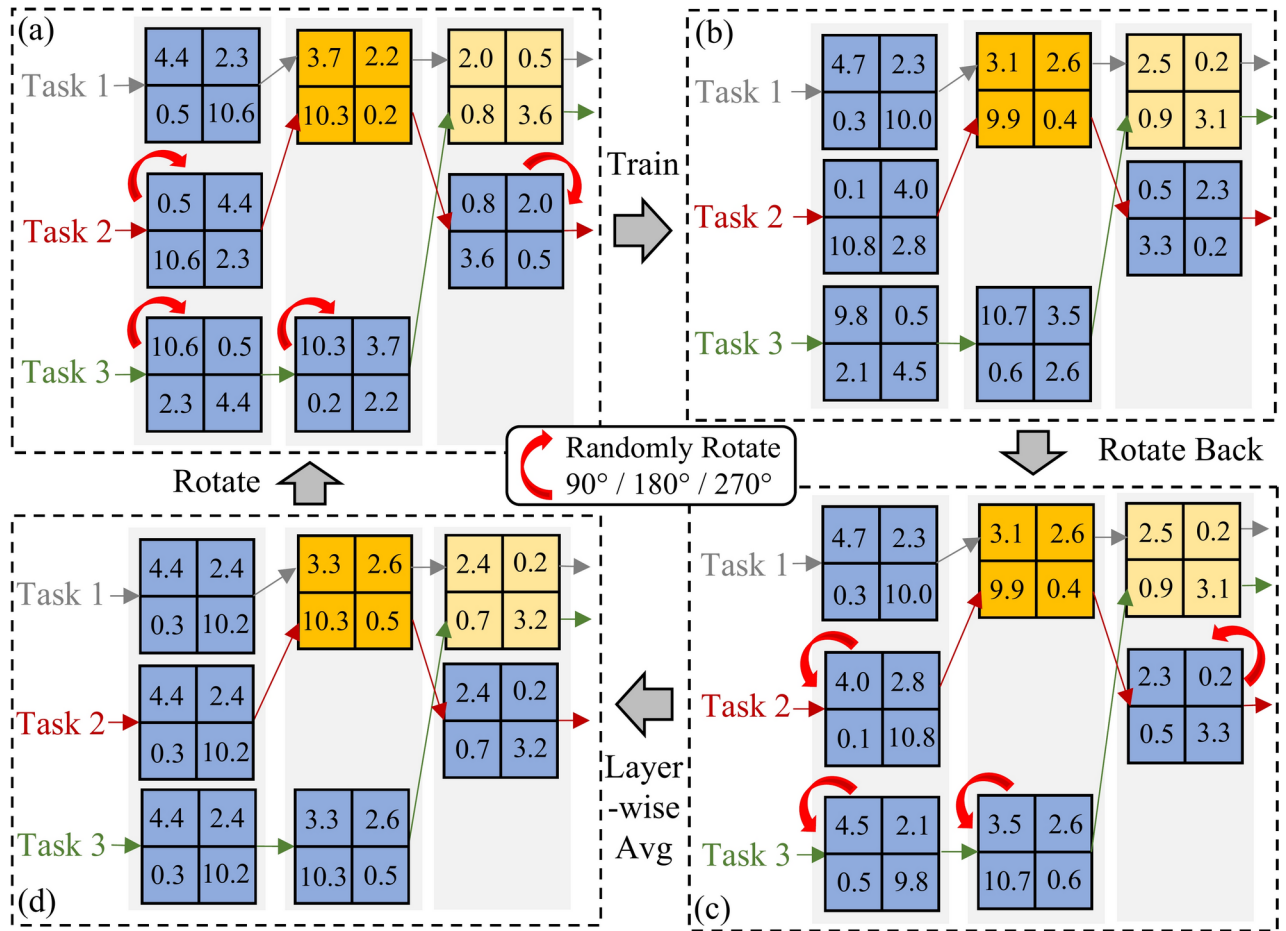


Fig. 7. Rotation Process in LUMEN-PRO framework.

Input: A MTL DONN M with N layers and T tasks

Initialization: A copy of M as M_{agg}

1 Extract the policy set from M as $\{\mathbf{P}\}$

2 **while** $i \leq \text{training\_iterations}$ **do**

3 **for** $n \leq N$ **do**

4 **if** spc_n^t in $\{\mathbf{P}_n\}$ **then**

5 Substitute the weights of t in M with the weights of bsc_n in M_{agg} after applying a rotation of α_n^t degrees.

6 Train and update weights in M .

7 **for** $n \leq N$ **do**

8 Apply a rotation of $4 - \alpha_n^t$ degrees to substituted weights in M .

9 Compute the average of the weights of all tasks in layer n in M , denote as $\mathbf{W}_n^{\text{tmp}}$.

10 **for** $t \leq T$ **do**

11 Replace weight of t in M_{agg} with $\mathbf{W}_n^{\text{tmp}}$.

12 **if** policy $\mathbf{P}_n^t$ is spc_n^t **then**

13 Substitute the weights of t in M with α_n^t -degrees rotated $\mathbf{W}_n^{\text{tmp}}$.

14 Evaluation on M .

Output: Well-trained MTL DONN M .

Algorithm 1. LUMEN-PRO Rotation Algorithm

Discussion and future work

Flexibility and limitations

Our MTL framework, LUMEN-PRO, is designed to handle diverse tasks simultaneously, providing high flexibility and adaptability. The system can accommodate various task combinations, including classification and segmentation, without requiring all tasks to belong to the same category. This is achieved by maintaining a shared backbone while adapting task-specific heads, ensuring scalability with minimal reconfiguration. Additionally, our framework imposes no restrictions on the number of classes for classification tasks, allowing datasets with different class counts to be trained together effortlessly.

The expressivity and modeling capacity of shared phase masks are essential aspects for future research. While LUMEN-PRO effectively leverages layer-wise rotation learning to enhance efficiency, the maximum number of tasks that can be trained together and the impact of dataset diversity remain open questions. Our study primarily focuses on optimizing the integration of multiple tasks within the optical neural network (DONN) framework. Broader investigations into the theoretical limits of multi-task learning in digital systems are beyond the scope of this work but represent promising directions for our further exploration.

In terms of scalability, the DONN system is inherently efficient and well-suited for processing large-scale datasets. In our experiments, we expanded input images to higher resolutions (e.g., 200×200 pixels) through numerical interpolation for both training and implementation. The physical properties of DONNs enable light signals to scale naturally, facilitating seamless adaptation to larger input dimensions. This scalability makes DONNs a promising candidate for handling increasingly complex datasets without requiring significant architectural modifications.

Handling richer data representations, such as RGB datasets like ImageNet, poses additional challenges due to the need for processing multi-channel color information. While ONNs can address this by leveraging multiple light channels (R, G, and B), achieving accuracy comparable to conventional neural networks remains an open research problem. However, DONN systems have demonstrated their effectiveness in handling dynamic and multi-variant datasets, such as augmented and moving MNIST, as shown in the previous work²⁶. These findings suggest that, despite existing challenges, DONNs hold significant potential for broader applications in high-dimensional and multi-modal learning tasks.

Rotation mechanism and efficiency

LUMEN-PRO maximizes resource efficiency through layer reuse via physical rotation without introducing additional latency or compromising inference performance. The task-specific rotations are pre-determined during training, enabling seamless switching between tasks without requiring time-intensive reconfiguration. Since the system simply rotates the corresponding layers based on pre-searched configurations, the inference speed remains consistent with single-task systems. Additionally, by reusing fixed diffractive layers, LUMEN-PRO achieves up to $4\times$ cost efficiency in multi-task learning without the need for separate systems or re-fabrication. Our framework is particularly advantageous for DONN systems, which rely on pre-fabricated components that cannot be dynamically adjusted post-deployment. Our rotation mechanism effectively facilitates multi-task learning in non-reconfigurable architectures while preserving the energy and memory benefits of DONNs. Our future work will explore further optimizations and scalability to enhance the framework's applicability.

Related work

Diffractive optical neural networks

Diffractive Optical Neural Networks (DONNs) is an optical system where the information encoding and the computation are realized by the manipulation of the light signal, which features with high energy efficiency, high computational speed and easy parallelism^{15,17,27–30}. The DONN system is composed by stacking diffractive layers in sequence as shown in Fig. 4. The input information is encoded with the coherent light signal on its optical characteristics, e.g., its intensity, amplitude, or phase. The diffractive layers are arrays embedding the phase modulations trained w.r.t the ML task for manipulating and encoding information on the light signal. The connection between layers is realized by the light diffraction when the light signal propagates between layers. At the end of the DONN system, a detector is employed to capture the light intensity pattern for the result readout and the analog-to-digital conversion. Note that the optical manipulation happens by nature with light propagation and modulation and the diffractive layers are implemented with passive optical devices without extra energy needed for functionality, resulting in ultrahigh power efficiency and computational speed. Once the training of a DONN system is completed on the digital computation platform, the trained DONN is deployed on the optical platform with non-configurable fabricated phase masks such as 3D printed phase masks, as diffractive layers for all-optical inference. Thus, DONNs lack reconfigurability for the weight parameters, bring significant energy and system cost overhead in practical application scenarios, especially for MTL.

Multi-task learning

There are several research trends addressing the multi-task learning efficiency: (1) common features extraction with feature transformation^{31–34} and feature selection approaches^{35,36}, (2) low-rank methods for weight parameter approximation and sharing^{37,38}, (3) clustering tasks based on task similarity^{39–41}, (4) simultaneous learning of parameters and pairwise task relations^{42–44}, and (5) decomposition approaches using multi-level parameters to model complex task structures^{45–47}.

Conclusion

In this paper, we propose LUMEN-PRO, an automating multi-task learning optical neural network framework that optimizes MTL DONN using physical principles. LUMEN-PRO converts a user-provided DONN backbone into

a supermodel with operator-level granularity, enabling gradient-based architecture search for optimal possible sharing patterns across multiple tasks, enhancing cost efficiency. LUMEN-PRO uses the rotation algorithm to fine-tune the architecture for higher accuracy and resource efficiency, leveraging physical properties of optical systems. On MNIST family, LUMEN-PRO achieves up to 13.51% higher accuracy and $4\times$ and $2\times$ better cost efficiency than single-task and sota MTL DONN methods. On CelebA dataset, LUMEN-PRO improves accuracy by up to 49.58% compared to SoTA MTL DONN algorithms. On energy efficiency, LUMEN-PRO achieves an $8.78\times$ gain, matching Nanophotonic in overall efficiency while surpassing it in per-operator efficiency due to its larger system. It also outperforms ReRAM-based and FPGA-based implementations with $9.6\times$ and $2.1\times$ energy efficiency gains, respectively. LUMEN-PRO enables flexible adjustment and identification of suitable models across datasets, facilitating efficient search for energy-efficient MTL DONN.

Data availability

The datasets generated and/or analysed during the current study are available in the following repository. MNIST: <http://yann.lecun.com/exdb/mnist/>. Fashion-MNIST (FMNIST): <https://arxiv.org/pdf/1708.07747>. Kuzushiji-MNIST (KMNIST): <https://arxiv.org/pdf/1812.01718>. Extension-MNIST-Letters (EMNIST): <https://ieeexplore.ieee.org/abstract/document/7966217>. CelebFaces Attributes Dataset (CelebA): https://openaccess.thecvf.com/content_iccv_2015/papers/Liu_Deep_Learning_Face_ICCV_2015_paper.pdf

Received: 13 July 2024; Accepted: 3 April 2025

Published online: 25 April 2025

References

- Zhang, Y. & Yang, Q. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering* **34**, 5586–5609 (2021).
- Kendall, A., Gal, Y. & Cipolla, R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7482–7491 (2018).
- Long, M., Cao, Z., Wang, J. & Yu, P. S. Learning multiple tasks with multilinear relationship networks. *Advances in neural information processing systems* **30** (2017).
- Mack, C. The multiple lives of moore's law. *IEEE Spectrum* **52**, 31–31 (2015).
- Caulfield, H. J. & Dolev, S. Why future supercomputing requires optics. *Nature Photonics* **4**, 261–263 (2010).
- Rios, C. et al. Integrated all-photonic non-volatile multi-level memory. *Nature photonics* **9**, 725–732 (2015).
- Shastri, B. J. et al. Photonics for artificial intelligence and neuromorphic computing. *Nature Photonics* **15**, 102–114 (2021).
- Li, Y., Chen, R., Sensale Rodriguez, B., Gao, W. & Yu, C. Multi-task learning in diffractive deep neural networks via hardware-software co-design. *Scientific Reports* 1–9 (2021).
- Li, Y., Gao, W. & Yu, C. Rubik's optical neural networks: Multi-task learning with physics-aware rotation architecture. arXiv preprint [arXiv:2304.12985](https://arxiv.org/abs/2304.12985) (2023).
- LeCun, Y. The mnist database of handwritten digits. <http://yann.lecun.com/exdb/mnist/> (1998).
- Xiao, H., Rasul, K. & Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint [arXiv:1708.07747](https://arxiv.org/abs/1708.07747) (2017).
- Clanuwat, T. et al. Deep learning for classical japanese literature. arXiv preprint [arXiv:1812.01718](https://arxiv.org/abs/1812.01718) (2018).
- Cohen, G., Afshar, S., Tapson, J. & Van Schaik, A. Emnist: Extending mnist to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, 2921–2926 (IEEE, 2017).
- Liu, Z., Luo, P., Wang, X. & Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)* (2015).
- Lin, X. et al. All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004–1008 (2018).
- Liu, W. et al. Holylight: A nanophotonic accelerator for deep learning in data centers. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 1483–1488 (IEEE, 2019).
- Li, Y., Chen, R., Gao, W. & Yu, C. Physics-aware differentiable discrete codesign for diffractive optical neural networks. In *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 1–9 (2022).
- Umuroglu, Y. et al. Finn: A framework for fast, scalable binarized neural network inference. In *Proceedings of the 2017 ACM/SIGDA international symposium on field-programmable gate arrays*, 65–74 (2017).
- Esser, S. K., Appuswamy, R., Merolla, P., Arthur, J. V. & Modha, D. S. Backpropagation for energy-efficient neuromorphic computing. *Advances in neural information processing systems* **28** (2015).
- Yuan, G. et al. Forms: Fine-grained polarized reram-based in-situ computation for mixed-signal dnn accelerator. In *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*, 265–278 (IEEE, 2021).
- Shafiee, A. et al. Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars. *ACM SIGARCH Computer Architecture News* **44**, 14–26 (2016).
- Chen, Y. et al. Dadiannao: A machine-learning supercomputer. In *2014 47th Annual IEEE/ACM International Symposium on Microarchitecture*, 609–622 (IEEE, 2014).
- Zhang, L., Liu, X. & Guan, H. Automtl: A programming framework for automating efficient multi-task learning. *Advances in Neural Information Processing Systems* **35**, 34216–34228 (2022).
- Jang, E., Gu, S. & Poole, B. Categorical reparameterization with gumbel-softmax. arXiv preprint:1611.01144 (2016).
- Dugas, C., Bengio, Y., Bélisle, F., Nadeau, C. & Garcia, R. Incorporating second-order functional knowledge for better option pricing. *Advances in neural information processing systems* **13** (2000).
- Mengu, D., Rivenson, Y. & Ozcan, A. Scale-, shift-, and rotation-invariant diffractive optical networks. *ACS photonics* **8**, 324–334 (2020).
- Shen, Y. et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics* **11**, 441 (2017).
- Mengu, D., Luo, Y., Rivenson, Y. & Ozcan, A. Analysis of diffractive optical neural networks and their integration with electronic neural networks. *IEEE Journal of Selected Topics in Quantum Electronics* **26**, 1–14 (2019).
- Feldmann, J., Youngblood, N., Wright, C. D., Bhaskaran, H. & Pernice, W. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature* **569**, 208–214 (2019).
- Li, Y. et al. Real-time multi-task diffractive deep neural networks via hardware-software co-design. *Scientific reports* **11**, 11013 (2021).
- Weinberger, K., Dasgupta, A., Langford, J., Smola, A. & Attenberg, J. Feature hashing for large scale multitask learning. In *Proceedings of the 26th annual international conference on machine learning*, 1113–1120 (2009).
- Yim, J. et al. Rotating your face using multi-task deep neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 676–684 (2015).

33. Chu, X., Ouyang, W., Yang, W. & Wang, X. Multi-task recurrent neural network for immediacy prediction. In *IEEE international conference on computer vision*, 3352–3360 (2015).
34. Zhang, Z., Luo, P., Loy, C. C. & Tang, X. Facial landmark detection by deep multi-task learning. In *ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part VI* 13, 94–108 (Springer, 2014).
35. Wang, X., Zhang, C. & Zhang, Z. Boosted multi-task learning for face verification with applications to web image and video search. In *2009 IEEE conference on computer vision and pattern recognition*, 142–149 (IEEE, 2009).
36. Ahmed, A., Aly, M. *et al.* Web-scale multi-task feature selection for behavioral targeting. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, 1737–1741 (2012).
37. Xu, J., Tan, P.-N., Luo, L. & Zhou, J. Gspartan: a geospatio-temporal multi-task learning framework for multi-location prediction. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, 657–665 (SIAM, 2016).
38. Cheng, B., Liu, G., Wang, J., Huang, Z. & Yan, S. Multi-task low-rank affinity pursuit for image segmentation. In *2011 International conference on computer vision*, 2439–2446 (IEEE, 2011).
39. An, Q., Wang, C. *et al.* Hierarchical kernel stick-breaking process for multi-task image analysis. In *Proceedings of the 25th international conference on Machine learning*, 17–24 (2008).
40. Zhang, Y. & Yeung, D.-Y. Multi-task warped gaussian process for personalized age estimation. In *IEEE computer society conference on computer vision and pattern recognition*, 2622–2629 (IEEE, 2010).
41. Almaev, T., Martinez, B. *et al.* Learning to transfer: transferring latent task structures and its application to person-specific facial action unit detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 3774–3782 (2015).
42. Chapelle, O. *et al.* Multi-task learning for boosting with application to web search ranking. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1189–1198 (2010).
43. Liu, A.-A., Su, Y.-T., Nie, W.-Z. & Kankanhalli, M. Hierarchical clustering multi-task learning for joint human action grouping and recognition. *IEEE transactions on pattern analysis and machine intelligence* **39**, 102–114 (2016).
44. Widmer, C., Leiva, J., Altun, Y. & Ratsch, G. Leveraging sequence classification by taxonomy-based multitask learning. In *The 14th Research in Computational Molecular Biology*, 522–534 (Springer, 2010).
45. Hong, Z., Mei, X., Prokhorov, D. & Tao, D. Tracking via robust multi-task multi-view joint sparse representation. In *IEEE international conference on computer vision*, 649–656 (2013).
46. Yan, Y., Ricci, E., Subramanian, R., Lanz, O. & Sebe, N. No matter where you are: Flexible graph-guided multi-task learning for multi-view head pose classification under target motion. In *Proceedings of the IEEE international conference on computer vision*, 1177–1184 (2013).
47. Wan, J. *et al.* Sparse bayesian multi-task learning for predicting cognitive outcomes from neuroimaging measures in alzheimer's disease. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 940–947 (IEEE, 2012).

Acknowledgment

W.G. and C.Y. acknowledge the support from the National Science Foundation through Grants No. 2316627 and 2428520.

Author contributions

C.Y. and C.D. contributed to the overall idea of this paper. S.Z. and Y.L. wrote the main manuscript text and conducted the experiments. W.G. analysed the results. All authors reviewed the manuscript.

Declarations

Competing interests

The author(s) declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to C.D.

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

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

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

© The Author(s) 2025