A Hybrid Quantum-Classical Approach Based on the Hadamard Transform for the Convolutional Layer

Hongyi Pan*, Xin Zhu, Salih Atici, Ahmet Enis Cetin
University of Illinois Chicago
hpan21@uic.edu

Motivation and Background

- We want to reduce the parameters and the computational cost for the deep neural networks, with the accuracy is maintained or improved.
- The convolutional layer, which is the critical component in the deep neural networks, is our target to be revised.
- Transforming the input data into the frequency domain can help capture the most important features while reducing the redundancy in the input data.

Background: Hadamard Transform

Hadamard Transform (HT) and its inverse (IHT):

- HT: $\mathbf{X} = \mathbf{H}_N\mathbf{x}$, IHT: $\mathbf{X} = \frac{1}{N}\mathbf{H}_N\mathbf{x}$.
- HT can be considered a binary version of DFT.

$$\mathbf{H}_N = \begin{cases} 1, N = 1 \\ \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, N = 2 \\ \begin{bmatrix} \mathbf{H}_{\frac{N}{2}} & \mathbf{H}_{\frac{N}{2}} \\ \mathbf{H}_{\frac{N}{2}} & -\mathbf{H}_{\frac{N}{2}} \end{bmatrix}, N \geq 4 \end{cases}.$$

$$\mathbf{H}_N = \mathbf{H}_2 \otimes \mathbf{H}_{\frac{N}{2}}, N \geq 4.$$

Hybrid Quantum-Classical HT

Method	1D Complexity	2D Complexity
Classical HT (Matrix-Vector Product)	$O(N^2)$	$O(N^3)$
Classical Fast HT	$O(N \log N)$	$O(N^2 \log N)$
Hybrid Quantum-Classical HT	$O(N)$	$O(N^2)$

Algorithm 1 The hybrid quantum-classical HT algorithm.

Input: The input vector $\mathbf{x} = [x_0\ x_1\ \ldots\ x_{N-1}]^T \in \mathbb{R}^N$ where $N = 2^M$, $M \in \mathbb{N}$.

Output: The output vector $\mathbf{X} = [x_0\ x_1\ \ldots\ x_{N-1}]^T$ which is the HT of the $\mathbf{x}$.

$b = \epsilon + \sum_{k=0}^{N-1} |x_k|$, where ϵ is any positive number;

$\tilde{\mathbf{x}} = [b\ x_1\ x_2\ \ldots\ x_{N-1}]^T$;

$c = ||\tilde{\mathbf{x}}|| = \sqrt{b^2 + \sum_{k=1}^{N-1} x_k^2}$.

$\bar{\mathbf{x}} = [\tilde{x}_0\ \tilde{x}_1\ \ldots\ \tilde{x}_{N-1}]^T = \tilde{\mathbf{x}}/c$;

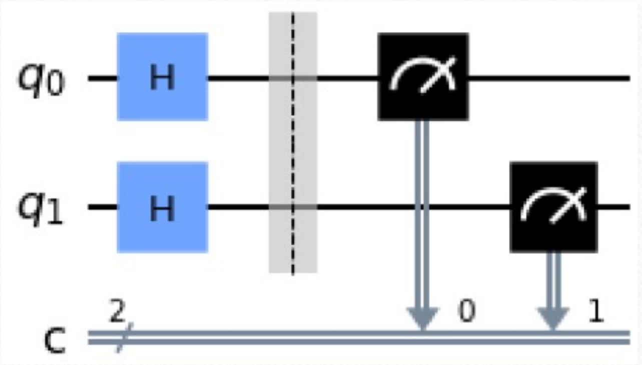
Prepare the state $|\Psi\rangle = \sum_{k=0}^{N-1} \bar{x}_k |k\rangle$ using M qubits;

Apply $\mathbf{H}^{\otimes M}$ on $|\Psi\rangle$;

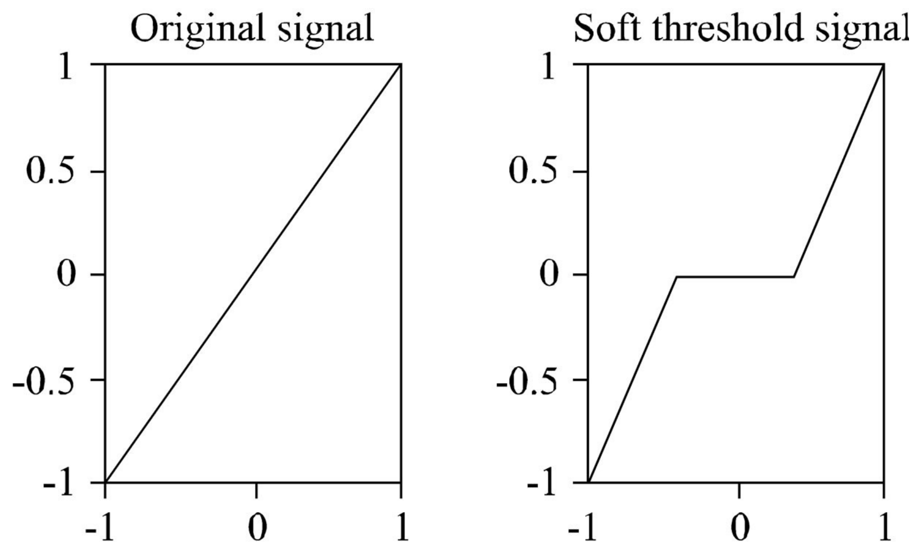
Measure all the M qubits to compute the probability p_k of obtaining the state $|k\rangle$ for $k = 0$ to $N-1$;

$\delta = \sqrt{\frac{1}{N}}(b - \mathbf{x}[0])$;

$\mathbf{X} = [c\sqrt{p_0} - \delta\ c\sqrt{p_1} - \delta\ \ldots\ c\sqrt{p_{N-1}} - \delta]^T$.



[1] Shukla, Alok, and Prakash Vedula. "A hybrid classical-quantum algorithm for solution of nonlinear ordinary differential equations." *Applied Mathematics and Computation* 442 (2023): 127708.

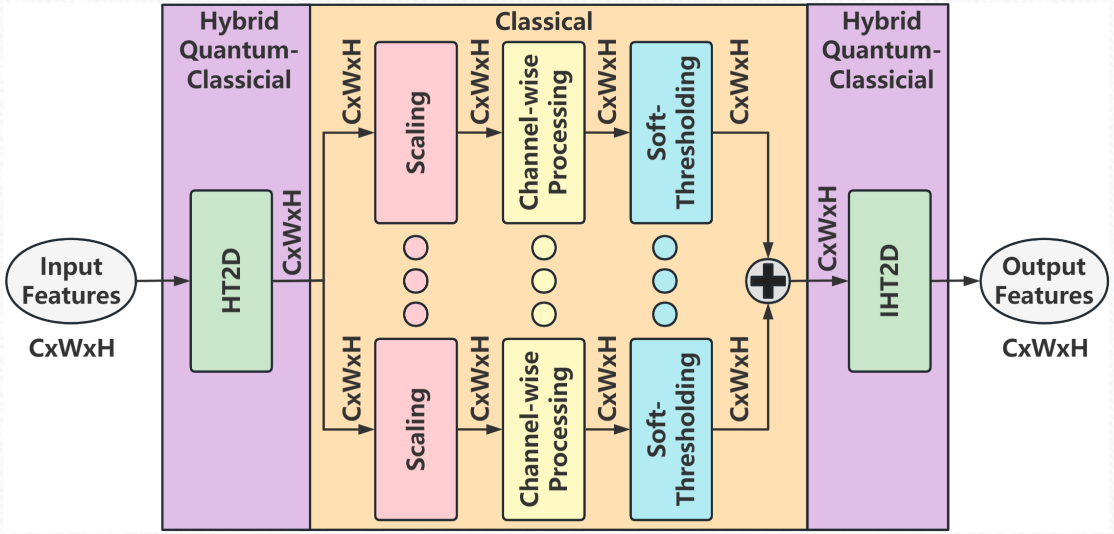


Soft-thresholding

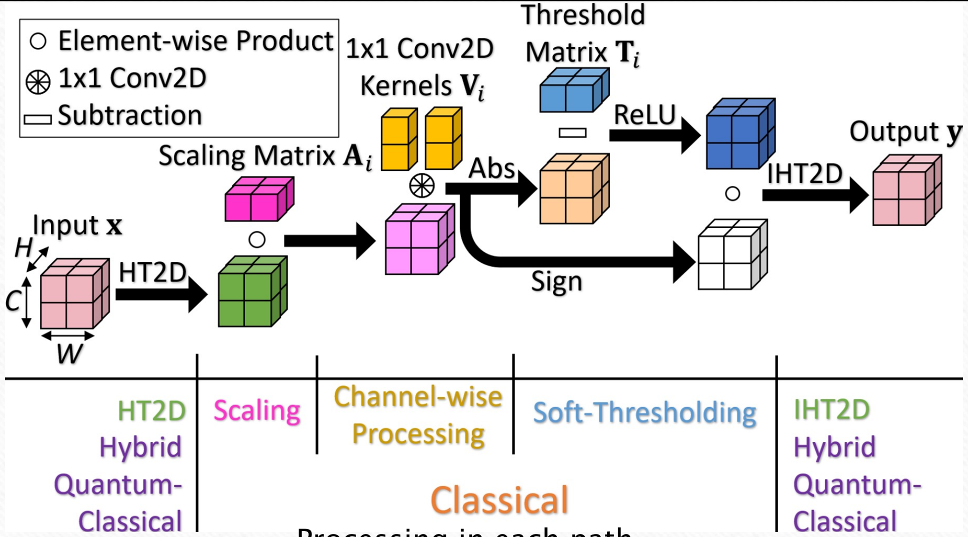
- $Y = S_T(X) = \mathrm{sign}(X)(|X| - T)_+$
- Soft-threshold is widely used in the wavelet to remove small noises in the transform domain.
- We don't use ReLU with the bias term because the positive and negative frequency components are equally important.

Hybrid Quantum-Classical HT-Perceptron Layer

- Layer are based on the convolution theorem: Convolution in the times/space domain is equivalent to the element-wise product in the transform domain.
- We apply width-wise and height-wise element-wise multiplication in the transform domain.
- We use the multi-path structure to utilize more trainable parameters to improve the accuracy.



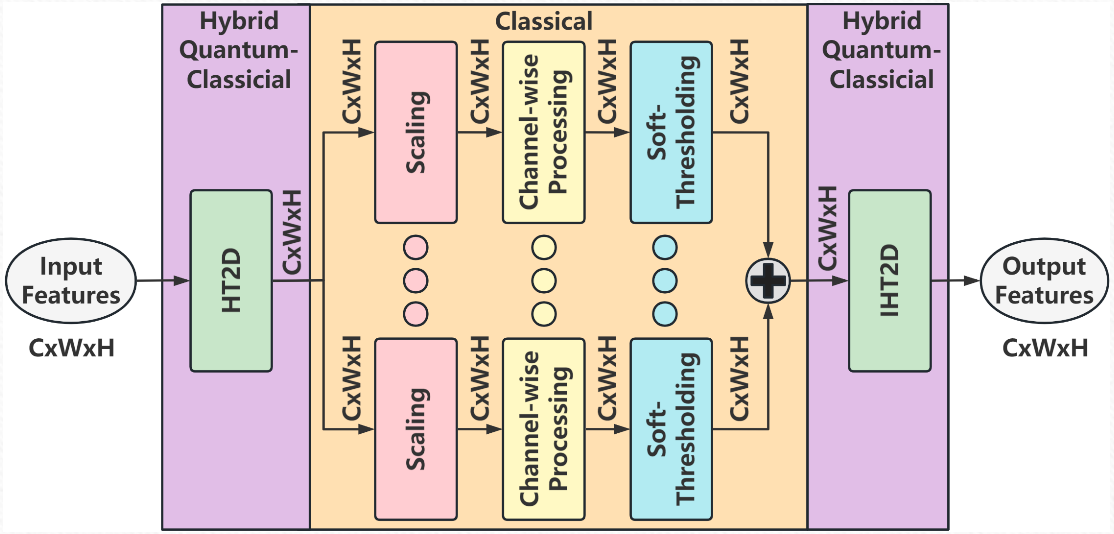
Structure of the proposed HT–perceptron layer



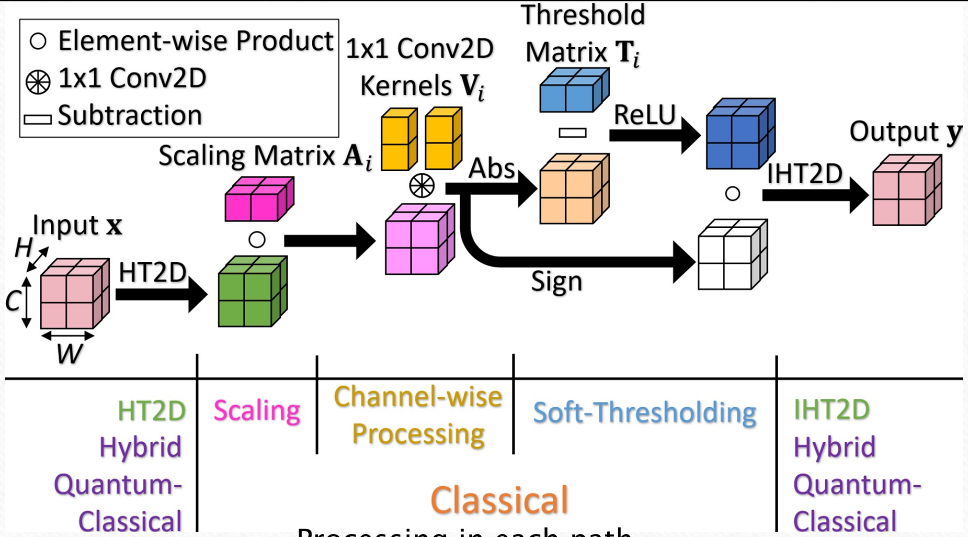
Processing in each path

Hybrid Quantum-Classical HT-Perceptron Layer

- We can use the hybrid quantum-classical HT algorithm to in the HT-perceptron layer to design a quantum-computer-friendly approach for the Conv2D layer.
- The layer is still trained using the classical backpropagation algorithm, because there is not quantum trainable variable. The quantum part is only to compute the HT and IHT.



Structure of the proposed HT–perceptron layer



Processing in each path

Comparison with the Conv2D layer: Parameters and MACs on a $C \times N \times N$ tensor

Layer (Operation)	Parameters	Multiply-Accumulate (MACs)
$K \times K$ Conv2D	$K^2 C^2$	$K^2 N^2 C^2$
3×3 Conv2D	$9C^2$	$9N^2 C^2$
Scaling, Soft-thresholding	$2N^2$	$N^2 C$
1×1 Conv2D	C^2	$N^2 C^2$
$\boldsymbol{P}$-Path HT-Perceptron	$\boldsymbol{PC^2 + 2PN^2}$	$\boldsymbol{PN^2C^2 + PN^2C}$
3-Path HT-Perceptron	$\boldsymbol{6N^2 + 3C^2}$	$\boldsymbol{3N^2C^2 + 3N^2C}$

- In the modern neural networks' most hidden layers, $C \gg N$.
- The HT-perceptron layers have a smaller coefficient on C^2 term than the 3×3 Conv2D layer.

Experimental Results: MNIST

Method	Parameters	MACs (M)	Accuracy
CNN	1,059,562	10.85	99.26%
HT-CNN (3-Path)	1,059,562	4.66 (57.1%↓)	99.31%

Experimental Results: CIFAR-10

Method	Parameters	MACs (M)	Accuracy
ResNet-20 (Official [2])	0.27M	-	91.25%
HT-Based ResNet-20 (Our early work [3])	133,082 (51.16%↓)	-	90.12%
ResNet-20 (Our trial, baseline)	272,474	41.32	91.66%
HT-ResNet-20 (1-Path)	199,898 (26.64%↓)	22.53 (45.5%↓)	91.25%
HT-ResNet-20 (3-Path)	199,898 (26.64%↓)	27.42 (33.6%↓)	91.29%
HT-ResNet-20 (5-Path)	248,282 (8.88%↓)	32.31 (21.8%↓)	91.58%

[2] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016.
[3] Pan, Hongyi, Diaa Badawi, and Ahmet Enis Cetin. "Block Walsh–Hadamard Transform-based Binary Layers in Deep Neural Networks." *ACM Transactions on Embedded Computing Systems* 21, no. 6 (2022): 1-25.

Experimental Results: Comparison with HT-based works on ImageNet-1K

Method	Publication	Params (M)	MACs (G)	Top-1 Acc	Top-5 Acc
ResNet-50 (Torchvision, Input 224^2)	CVPR 2015	25.56	4.12	76.13%	92.86%
ResNet-50 (AugSkip) Zero Init	ICLR 2022	25.56	4.12	76.37%	-
HT-ResNet-50 (3-Path, Input 224^2)	**This work**	**22.63 (11.5%↓)**	**3.60 (12.6%↓)**	**76.36%**	**93.02%**
ResNet-50 (Our trial, Input 256^2)	-	25.56	5.38	76.18%	92.94%
HT-ResNet-50 (3-Path, Input 256^2)	**This work**	**22.63 (11.5%↓)**	**4.59 (14.9%↓)**	**76.77%**	**93.26%**

Other HT-based work including [4] [5] contain no ResNet-based result, but their networks produce worse accuracy results than their baseline models according to Table 4 in each paper.

[4] Deveci, T. Ceren, Serdar Cakir, and A. Enis Cetin. "Energy efficient hadamard neural networks." *arXiv preprint arXiv:1805.05421* (2018).

[5] Akhauri, Yash. "Hadanets: Flexible quantization strategies for neural networks." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0-0. 2019.

Conclusion

- We presented a HT-based neural network layer to replace the Conv2D layer, which obtained comparable or higher accuracy results using fewer parameters and less computational cost on CIFAR-10 and ImageNet-1K compared to the vanilla ResNet models.
- We investigated the HT-based neural network layer in the quantum computation field to design a quantum-computer-friendly approach. We employ quantum Hadamard Gates to reduce the computational cost for the 2D HT.

Thank you!

Email: hpan21@uic.edu

International Conference On Machine Learning