Perforated Neural Networks for Semantic Segmentation on the Edge

Chris Dunkers
Thoro AI
Pittsburgh, PA
cdunkers@thoro.ai

Cameron Kisailus
Thoro AI Pittsburgh, PA
ckisailus@thoro.ai

Erin Yanacek
Perforated AI
Pittsburgh, PA
Erin@perforatedai.com

Rorry Brenner
Perforated AI
Pittsburgh, PA
rorry@perforatedai.com

Abstract: Semantic segmentation on edge hardware demands simultaneously high accuracy and low parameter count, two objectives that are in direct tension under traditional neural network training. This paper applies perforated learning [1], growing neural networks during training with artificial dendrite nodes, to the task of human detection via semantic segmentation for an industrial autonomous robotics platform, using a DeepLabV3 [2] architecture with an Atrous Spatial Pyramid Pooling (ASPP) head and a flexible backbone. We report results from a systematic hyperparameter sweep across three backbone families: ResNet-18, MobileNet V2, and ShuffleNet V2. We compare traditional and perforated networks at matched parameter counts. Perforated models outperform traditional models at every parameter budget tested, achieving higher Person-class F1 scores across all backbone sizes. These results extend the perforated learning track record into real-time semantic segmentation and industrial edge deployment, and demonstrate that the dendritic advantage is architecture-agnostic within the DeepLabV3 family.

Keywords: Semantic Segmentation, Deep Learning, Perforated Learning, Dendritic Neural Networks, Autonomous Robotic Systems, Perforated Backpropagation, Edge Segmentation, Edge AI

1 Introduction

Autonomous robotic systems operating in industrial environments must make real-time decisions from continuous visual streams. Semantic segmentation, the task of assigning a class label to every pixel in an image, is central to this perception pipeline. It enables robots to detect humans, avoid obstacles, and navigate safely. Deployment requirements often dictate that this segmentation is performed on the edge as opposed to the cloud, which means these models must run on hardware with strict constraints on memory and compute, and must produce predictions fast enough for real-time control loops [2, 3].

The fundamental challenge for edge segmentation is the same as for all edge machine learning: high-accuracy models tend to be large and slow, while small and fast models tend to sacrifice accuracy. This tradeoff is especially consequential in safety-critical applications such as human detection, where a false negative, failing to detect a person, carries significant risk. A model with insufficient Person-class recall may pass on-device size constraints but fail on deployment safety requirements.

Dendritic models are those in which neuron nodes in a network have dendritic input nodes defined by having a single output to one parent neuron [4]. Perforated networks grow models during training

with the addition of dendrite nodes. Perforated Backpropagation (PB) [1] is a neuroscience-inspired training framework that enables multiple learning rules within a single network. Unlike traditional compression techniques such as pruning [5], quantization, or knowledge distillation [6], which trade accuracy for compactness, PB has consistently demonstrated the ability to improve both accuracy and model size simultaneously across a range of domains [1, 7, 8]. The key mechanism is the introduction of a secondary learning signal, such as a modified Cascade Correlation rule [9], applied to a subset of nodes in the network, such as the dendrite nodes, which sit outside the backpropagation graph, in analogy to the nonlinear computation performed by biological dendrites [10].

This paper reports the application of PB to semantic segmentation for industrial autonomy. The experimental context is a collaboration with Thoro.ai [11], whose autonomy platform enables industrial OEMs to operate their equipment as autonomous robots. Their pipeline requires accurate, real-time human detection from a continuous camera stream, and is subject to strict on-device memory and latency constraints. Section 2 reviews the biological and algorithmic foundations of Perforated Backpropagation. Section 3 situates this work in the literature on semantic segmentation, edge deployment, and dendritic ML. Section 4 describes the DeepLabV3 architecture, backbone variants, and hyperparameter sweep methodology. Section 5 presents results. Section 6 discusses limitations of the current work. Section 7 concludes.

2 Background

2.1 Active Dendrites in Biological Neurons

The standard artificial neuron, introduced in 1943 [12] and formalized as the perceptron in 1958 [13], models a biological neuron as a weighted sum of inputs followed by a nonlinear activation function. This outdated model omits a significant feature of biological computation characterized by 21st century neuroscience: the active, nonlinear processing performed by dendrites as signals travel from synapses to the soma.

Dendrites are not passive cables. They generate localized N-methyl-D-aspartate (NMDA) receptor-mediated spikes whose threshold voltage is itself voltage-dependent, producing a thresholded nonlinear response that allows a single dendritic branch to perform coincidence detection and feature gating [10, 14]. Some researchers have proposed that dendrites, and not neurons, should be regarded as the fundamental computational unit of human cognition [15]. The gap between biological neurons and their artificial counterparts is therefore not merely one of scale but of fundamental computational architecture [16].

PB was designed to close part of this gap by introducing a dendritic-like module to standard deep network neurons while remaining compatible with existing PyTorch architectures [17].

2.2 Cascade Correlation and Perforated Backpropagation

PB draws on Cascade Correlation [9], an algorithm for single-layer networks in which candidate nodes are trained to maximize their correlation with the network's output error while existing weights are frozen. Once trained, the best candidate is frozen and incorporated as an additional input to the output layer. PB generalizes this idea to deep networks by targeting each neuron's back-propagated error gradient rather than the network-level output error. Training alternates between a *neuron phase* (standard gradient descent until validation plateau) and a *dendrite phase* (neuron weights frozen; dendrite nodes added to selected neurons and trained via a modified Cascade Correlation rule until their correlations plateau; the best node per neuron is then frozen and incorporated into the forward pass). Because error is never propagated *through* dendrite nodes, they remain outside the gradient-descent graph, in analogy to the biological process of neurons learning from signals from other neurons with dendrites as supporting components. Full mathematical details are given in the original Perforated Backpropagation Paper [1].

3 Related Work

Semantic segmentation. DeepLabV3 [2] introduced the use of atrous (dilated) convolutions and Atrous Spatial Pyramid Pooling (ASPP) for semantic segmentation, enabling multi-scale feature extraction. The ASPP module applies parallel atrous convolutions with different dilation rates to the backbone feature map, capturing context across a range of receptive fields. DeepLabV3 supports flexible backbone substitution, making it a natural platform for studying backbone efficiency tradeoffs [18].

Efficient backbones for edge deployment. MobileNetV2 [3] introduced inverted residuals and linear bottlenecks to reduce the computational cost of CNNs for mobile and edge deployment. ShuffleNetV2 [19] further improved efficiency through channel splitting and shuffling, demonstrating that carefully designed lightweight backbones can match the accuracy of heavier networks at a fraction of the parameter count. Both architectures have been widely adopted as backbones in segmentation models targeting resource-constrained hardware [20].

Model compression. Standard approaches to model compression for edge deployment include structured and unstructured pruning [5], post-training quantization, and knowledge distillation [6]. These methods typically trade accuracy for compactness. Neural architecture search (NAS) methods [20] attempt to find efficient architectures automatically, but require substantial compute and do not easily generalize across tasks. PB differs in that it applies to any existing PyTorch model without architectural redesign, and has demonstrated simultaneous improvements in both accuracy and model size [1].

Dendritic models in machine learning. Dendritic-inspired architectures date to at least 2003 [21], with subsequent work showing that morphological perceptrons can perfectly classify any training dataset [22] and that hardware-oriented dendritic architectures can achieve power-efficient inference [23]. A 2021 review found that dendritic models consistently outperform traditional networks at matched parameter counts, even without reaching state-of-the-art absolute accuracy [4]. Perforated learning is distinguished from prior work by growing networks with dendrites during the training process and PB is distinguished by introducing separate learning rules for multiple node types within multi-layer networks.

Prior PB results. PB has been validated across drug toxicity prediction (TrimNet on Tox21 [24]), stock forecasting (HIST on CSI300 [25]), ICU mortality prediction (mTAN on PhysioNet [26]), NLP benchmarks (BERT variants [27]), protein classification (ProteinBERT), image recognition (MobileNetV3 on CIFAR-10 [20]), and keyword spotting on microcontroller-class hardware [8]. The present paper is the first to apply PB to semantic segmentation and to an industrial autonomous robotics deployment context.

4 Method

4.1 Task and Deployment Context

The target task is semantic segmentation of grayscale camera frames for real-time human detection aboard industrial autonomous equipment. The deployment hardware imposes strict constraints on both model size (RAM/flash budget) and inference latency (the control loop requires predictions within a fixed time budget). A ResNet-18-backbone DeepLabV3 model served as the operational point of comparison, running at approximately 58 ms per frame on the target hardware.

The primary evaluation metric is **Person-class F1 score**, computed from per-pixel precision and recall on the Person semantic class. Person F1 is the operationally critical metric: the downstream safety system treats any Person-class pixel prediction as a trigger for protective action, so Person recall (avoiding false negatives) and Person precision (avoiding unnecessary stops) both directly affect system safety and operational efficiency.

4.2 Dataset

All training and evaluation data were collected using a Carnegie Robotics MultiSense stereo camera mounted on the target industrial robotic platform. The dataset consists of 960x600 pixel grayscale (single-channel) images captured in industrial, commercial, and retail environments including warehouse settings. Dataset included 12,606 training images and 3,300 validation images, including 10,120 Person labels and 18,877 Drivable Area labels. A single image could contain multiple people or drivable areas. Dataset was labelled using semi-manual annotation with help from SAM2 [28]. Masks were defined as a polygon of points. Camera mounting height was lower than conventional surveillance or research capture configurations, reflecting the robot's operational perspective.

Existing pretrained segmentation models were evaluated but found to underperform in this deployment context for two primary reasons: (1) most publicly available segmentation datasets are dominated by outdoor, street-scene, or general indoor imagery rather than warehouse and industrial environments; and (2) standard models expect three-channel RGB input, whereas the MultiSense camera produces grayscale imagery. These factors motivated training purpose-built models on domain-specific data rather than fine-tuning or deploying off-the-shelf solutions.

4.3 Architecture

We use DeepLabV3 [2] with an ASPP head. The ASPP module applies multiple atrous convolutions with different dilation rates in parallel, concatenates the resulting feature maps along with image-level context from global average pooling, and projects to the target class count. This design enables multi-scale context aggregation without a proportional increase in parameters. The backbone is treated as a substitutable component; we evaluate three backbone families:

- **ResNet-18** [29]: the original operational backbone, a standard residual network with approximately 12.1 M parameters in the segmentation configuration.
- **MobileNetV2** [3]: an inverted-residual lightweight backbone yielding approximately 5.3 M parameters.
- **ShuffleNetV2** [19]: a channel-split-and-shuffle lightweight backbone yielding approximately 3.4 M parameters.

For experimentation each model was perforated with dendrites using the standard PB PyTorch interface [17], requiring no changes to model definitions or data pipelines. For this set of experimentation only the ASPP head is perforated.

4.4 Hyperparameter Sweep

We conducted a systematic hyperparameter sweep comparing traditional (non-perforated) and perforated DeepLabV3 models across all three backbone families. The following hyperparameters were explored:

- **Backbone:** ResNet-18, MobileNetV2, ShuffleNetV2
- **ASPP width scaling:** Scaling factor applied to the ASPP head channel dimensions.
- **Decoder intermediate and output width scaling:** Independently scaled.
- **Regularization:** Dropout rate, weight decay, label smoothing.
- **Data augmentation:** Enabled or disabled.
- **Learning rate:** Multiple orders of magnitude.
- **Perforation parameters:** Maximum dendrites per neuron.
- **Model type:** Traditional neural network, or dendritic network.

Test scores were recorded at the epoch of maximum validation performance, consistent with standard model selection practice and prior PB evaluations [1].

5 Experimental Results

5.1 Sweep Results

Table 1 reports the best-performing traditional and perforated models from the sweep at each approximate parameter budget, ranked by Person F1 score. Figure 1 illustrates the Pareto frontier of Person F1 versus parameter count for both model types.

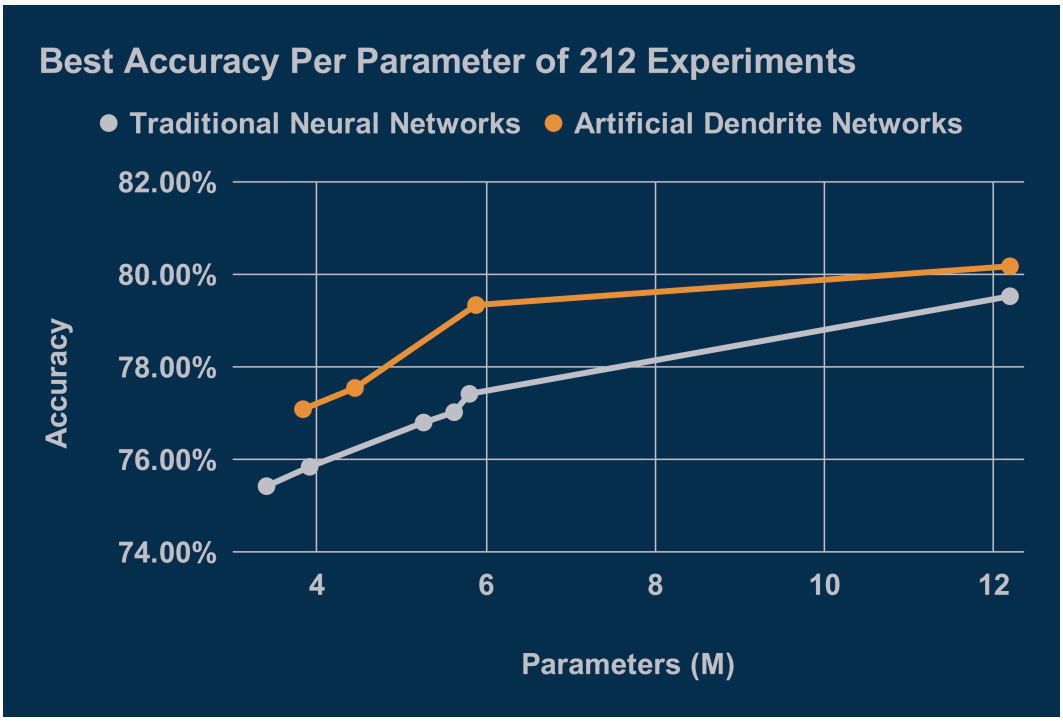

Figure 1: Best traditional and dendritic models from the hyperparameter sweep, plotted by parameter count (x-axis) versus Person F1 score (y-axis). At every parameter budget, the highest Person F1 is achieved by a perforated model.

Table 1: Best traditional and perforated models from the hyperparameter sweep, by parameter count. Person F1 is the primary evaluation metric. Backbone abbreviations: RN18 = ResNet-18, MNV2 = MobileNetV2, SNV2 = ShuffleNetV2.

Type	Backbone	Parameters	Person F1
Traditional	SNV2	3.40M	75.43%
Traditional	SNV2	3.91M	75.85%
Traditional	MNV2	5.26M	76.80%
Traditional	MNV2	5.62M	77.03%
Traditional	MNV2	5.80M	77.42%
Traditional	RN18	12.19M	79.53%
Perforated	SNV2	3.84M	77.09%
Perforated	SNV2	4.45M	77.55%
Perforated	MNV2	5.88M	79.34%
Perforated	RN18	12.19M	80.18%

Two patterns are immediately apparent:

- **At every parameter budget**, the highest Person F1 is achieved by a perforated model.
- **At every Person F1 level**, the smallest model achieving that score is a perforated model.

5.2 Key Findings

Efficiency gain: matched accuracy at 52% fewer parameters. The best dendritic MobileNetV2 model achieves 79.34% Person F1 at 5.88M parameters, essentially matching the best traditional ResNet-18 model at 79.53% Person F1 while using 52% fewer parameters (5.88M vs. 12.19M). This cross-backbone comparison illustrates the practical value of dendritic training for edge deployment: a practitioner constrained by memory or compute can substitute a dendritic MobileNetV2 for a traditional ResNet-18 and retain comparable detection performance at less than half the model size.

Accuracy gain at matched parameter count. Within each backbone family, dendritic models consistently outperform traditional models at matched parameter counts. For ShuffleNetV2, the best traditional network reaches 75.85% Person F1 at 3.91M parameters, while the best dendritic network reaches 77.09% at a slightly smaller 3.84M parameters, a 1.24 percentage point improvement with marginally fewer parameters. For MobileNetV2, the best traditional network reaches 77.42% at 5.80M parameters, while the best dendritic network reaches 79.34% at 5.88M parameters, a 1.92 percentage point gain at essentially the same size. For ResNet-18, the traditional network reaches 79.53% at 12.19M parameters, while the dendritic network reaches 80.18% at an identical 12.19M parameters, a 0.65 percentage point improvement with no increase in model size.

Architecture generality. The perforated advantage is consistent across all three backbone families tested. This generality is a stronger result than a single high-performing outlier: it indicates that the mechanism underlying perforated learning provides a consistent learning signal improvement regardless of backbone architecture within the DeepLabV3 family.

Table 2: Summary of key results comparing the best perforated ShuffleNetV2 model against traditional models at similar parameter counts, and the dendritic ResNet-18 result. The operational ResNet-18 baseline was evaluated on a separate test set and is therefore excluded from sweep comparisons. On that alternative test set, the baseline achieved a Person F1 of 75.544; the best perforated ShuffleNetV2 model achieved 75.62, confirming parity.

.

Model	Type	Params	Person F1	Latency
ResNet-18 operational baseline	Traditional	14.0M	—	58 ms
Best trad. ShuffleNetV2	Traditional	3.91M	75.85%	—
Best trad. MobileNetV2	Traditional	5.80M	77.42%	—
Best trad. ResNet-18	Traditional	12.19M	79.53%	—
Best perforated ShuffleNetV2	**Perforated**	**3.84M**	**77.09%**	**∼25 ms**
Best perforated MobileNetV2	Perforated	5.88M	79.34%	—
Best perforated ResNet-18	Perforated	12.19M	80.18%	—

6 Limitations

This study has several limitations that future work should address.

Single deployment environment. All training and evaluation data were collected from a single industrial robotics platform using a single camera configuration. Generalization to other robotic form factors, mounting heights, environments, or sensor types remains untested.

Grayscale imagery. The dataset consists entirely of single-channel grayscale images produced by the Carnegie Robotics MultiSense camera.

Partial perforation. In these experiments, only the ASPP segmentation head was perforated; the backbone was left unmodified. Perforating backbone layers may yield further gains and is a natural direction for future exploration.

Single segmentation architecture. Results are reported only for DeepLabV3 with an ASPP head. The extent to which the dendritic advantage transfers to other segmentation architectures, has not been evaluated.

6.1 Discussion

The consistent dominance of perforated models across backbone families and parameter budgets mirrors results from prior PB evaluations in other domains. In the keyword spotting study [8], dendritic models dominated the Pareto frontier of accuracy versus parameter count across 800 hyperparameter trials. The present result replicates this pattern for a fundamentally different task (pixel-level segmentation vs. audio classification), a different architecture family (DeepLabV3 with ASPP vs. convolutional classifiers), and a different evaluation metric (Person F1 vs. test accuracy).

Semantic segmentation presents additional challenges relative to full image classification: the model must produce spatially coherent pixel-level predictions, and the ASPP head introduces multi-scale feature aggregation that is not present in standard classifiers. The fact that perforation improves performance within this more complex architecture suggests that the dendrite node learning signal generalizes to the convolutional feature maps and intermediate representations used in segmentation, not only to the fully-connected layers of classifiers.

The Person-class specificity of the evaluation is also noteworthy. Person F1 is a more challenging and more operationally meaningful metric than mean Intersection over Union (mIoU) in this context because the class is rare relative to background and ground-class pixels, and because detection failures carry safety consequences. The dendritic advantage persisting at this granularity suggests that dendrite nodes improve feature discrimination for the minority class, not only aggregate performance.

7 Conclusion

We have presented the application of perforated neural networks to semantic segmentation for industrial autonomous robotics, demonstrating that perforated models consistently outperform traditional networks at every parameter budget across three backbone families within the DeepLabV3 architecture. The best perforated MobileNet model achieves Person F1 comparable to the best traditional ResNet model with fewer parameters and a perforated ShuffleNet model matched the operational baseline ResNet model's performance running at approximately 25 ms per frame, a 57% latency reduction that enables dual-stream perception on the same hardware.

These results add semantic segmentation and industrial edge deployment to the list of domains where perforated learning has demonstrated consistent improvements [1, 7, 8]. The technique's generality across backbone architectures (ResNet-18, MobileNetV2, ShuffleNetV2) and its efficacy on the ASPP segmentation head suggest that the perforated learning signal is not tied to any particular layer type or architectural pattern.

For edge AI model builders, the implication is practical: swapping a traditional training pipeline for a perforated one can simultaneously reduce deployment cost and improve safety-critical detection metrics, without changes to the model architecture or training data. Future work will explore application to additional segmentation architectures, multi-class edge scenarios, and hardware-aware co-optimization of backbone scaling and dendritic depth.

We encourage PyTorch users to try this on their own models with the instructions in our GitHub repository [17].

Acknowledgments

The authors thank the team at Thoro AI for the collaboration, domain expertise, and access to the industrial deployment environment and dataset that made this work possible. We also thank Carnegie Robotics for the MultiSense stereo camera platform used for data collection.

References

[1] R. Brenner and L. Itti. Perforated backpropagation: A neuroscience inspired extension to artificial neural networks. *arXiv preprint arXiv:2501.18018*, 2025.

[2] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[3] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[4] S. Chavlis and P. Poirazi. Drawing inspiration from biological dendrites to empower artificial neural networks. *Current Opinion in Neurobiology*, 70:1–10, 2021.

[5] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. In *International Conference on Learning Representations (ICLR)*, 2016.

[6] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[7] R. Brenner, E. Davis, R. Chaudhari, R. Morse, J. Chen, X. Liu, Z. You, and L. Itti. Exploring the performance of perforated backpropagation through further experiments. *arXiv preprint arXiv:2506.00356*, 2025.

[8] R. Brenner, V. Gopal, A. I. Goutis, R. Crewe, and E. Yanacek. Dendritic neural networks for keyword spotting. Edge Impulse Imagine 2025 Contest, Best Model Development Award, 2025. URL https://www.edgeimpulse.com/blog/edge-impulse-contest-2025-winners/.

[9] S. E. Fahlman and C. Lebiere. The cascade-correlation learning architecture. In *Advances in Neural Information Processing Systems*, volume 2, 1989.

[10] G. Major, M. E. Larkum, and J. Schiller. Active properties of neocortical pyramidal neuron dendrites. *Annual Review of Neuroscience*, 36:1–24, 2013.

[11] thoro.ai: Autonomous platform for industrial OEMs. https://thoro.ai, 2025.

[12] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, 1943.

[13] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[14] G. Major, A. Polsky, W. Denk, J. Schiller, and D. W. Tank. Spatiotemporally graded NMDA spike/plateau potentials in basal dendrites of neocortical pyramidal neurons. *Journal of Neurophysiology*, 99(5):2584–2601, 2008.

[15] T. Branco and M. Häusser. The single dendritic branch as a fundamental functional unit in the nervous system. *Current Opinion in Neurobiology*, 20(4):494–502, 2010.

[16] B. Widrow and M. A. Lehr. 30 years of adaptive neural networks: Perceptron, madaline, and backpropagation. *Proceedings of the IEEE*, 78(9):1415–1442, 1990.

[17] Perforated AI. Perforated AI PyTorch library. https://github.com/PerforatedAI/PerforatedAI, 2024.

[18] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[19] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[20] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[21] G. X. Ritter, L. Iancu, and G. Urcid. Morphological perceptrons with dendritic structure. In *The 12th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2003.

[22] H. Sossa and E. Guevara. Efficient training for dendrite morphological neural networks. *Neurocomputing*, 131:132–142, 2014.

[23] X. Li, J. Tang, Q. Zhang, B. Gao, J. J. Yang, S. Song, W. Wu, W. Zhang, P. Yao, N. Deng, L. Deng, Y. Xie, H. Qian, and H. Wu. Power-efficient neural network with artificial dendrites. *Nature Nanotechnology*, 2020.

[24] P. Li, Y. Li, C.-Y. Hsieh, S. Zhang, X. Liu, H. Liu, S. Song, and X. Yao. TrimNet: Learning molecular representation from triplet messages for biomedicine. *Briefings in Bioinformatics*, 22(4):bbaa266, 2021.

[25] W. Xu, W. Liu, L. Wang, Y. Xia, J. Bian, J. Yin, and T.-Y. Liu. HIST: A graph-based framework for stock trend forecasting via mining concept-oriented shared information. In *arXiv preprint arXiv:2110.13716*, 2021.

[26] S. N. Shukla and B. M. Marlin. Multi-time attention networks for irregularly sampled time series. In *International Conference on Learning Representations (ICLR)*, 2021.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

[28] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[29] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.