US Patent 12,242,946-B1 - MLiglon Corporation

14 July 2025. Integer Gate Logic (IGL) Benchmark Comparison Study.

IGL produces networks that are smaller and more accurate than those produced by back propagation using the same parameter memory sizes. Also, IGL uses no floating point math and uses integer values only. This means deployed models produce results quicker and with less power, and fit in smaller memory spaces. Lookup tables can be used as well for all necessary multiplications and additions, greatly enhancing computation speeds for inference as well as training.

This is useful for large language models, AI datacenters, robotics, remote sensing, drones, microcontrollers, image and speech recognition.

The benchmarks below show results from the MNIST dataset and comparative parameter memory size and accuracy on training, verification, and final test data. Sample MNIST handwritten digits (70,000 images are included in the dataset, 60,000 for training, 10,000 normally for verification, and 10,000 for final testing):

PyTorch using Backpropagation with **37.908** kilobytes of parameters and 4 byte (32-bit) floating point values for parameters being trained. Final test average over all 10 digits: **99.22%.** *All data was used for training and none for validation to increase final test accuracy.*

MNIST	Digit	Train	Validate	Final Test
9477p	0	99.82	Unused	99.62
BackProp	1	99.91		99.68
4 Byte	2	99.73		99.08
37.908	3	99.52		99.07
kb	4	99.83		99.25
	5	99.76		98.94
	6	99.78		99.42
	7	99.66		99.11
	8	99.65		99.09
	9	99.57		98.89
	Average	99.72		99.22

PyTorch using Backpropagation with **12.636** kilobytes of parameters and 4 byte (32-bit) floating point values for parameters being trained. Final test average over all 10 digits: **98.58%.**

MNIST	Digit	Train	Validate	Final Test
3159p	0	99.56	Unused	99.35
BackProp	1	99.55		99.50
4 Byte	2	99.06		98.77
12.636	3	98.15		97.89
kb	4	98.56		98.39
	5	98.94		98.67
	6	99.68		99.36
	7	99.35		98.83
	8	98.67		98.39
	9	97.04		96.66
	Average	98.86		98.58

IGL with **9.537** kilobytes of parameters and 1 byte (8-bit) integer values for parameters being trained. Final test average over all 10 digits: **99.28%.**

MNIST	Digit	Train	Validate	Final Test
9537p	0	99.85	99.90	99.64
IGL	1	99.80	99.92	99.74
1 Byte	2	99.67	99.89	99.22
9.537	3	99.57	99.86	99.11
kb	4	99.86	99.93	99.27
	5	99.81	99.85	99.20
	6	99.89	99.98	99.59
	7	99.69	99.89	99.11
	8	99.60	99.83	99.10
	9	99.48	99.71	98.86
	Average	99.72	99.88	99.28

Conclusion: Integer Gate Logic (IGL) using 9.537 kilobytes of parameters produces 99.28% accuracy across all 10 MNIST digits as compared to 37.908 kilobytes (4x) of parameters using backpropagation which produces 99.22% accuracy across the 10 MNIST digits. IGL produces significantly smaller and more accurate models than backpropagation when compared on a parameter-size basis.

This in consequence produces models which train and infer much more efficiently than backpropagation as well for several reasons. One of these reasons is integer math only is used, another reason is the possibility to use extremely small memory lookup tables instead of math altogether.