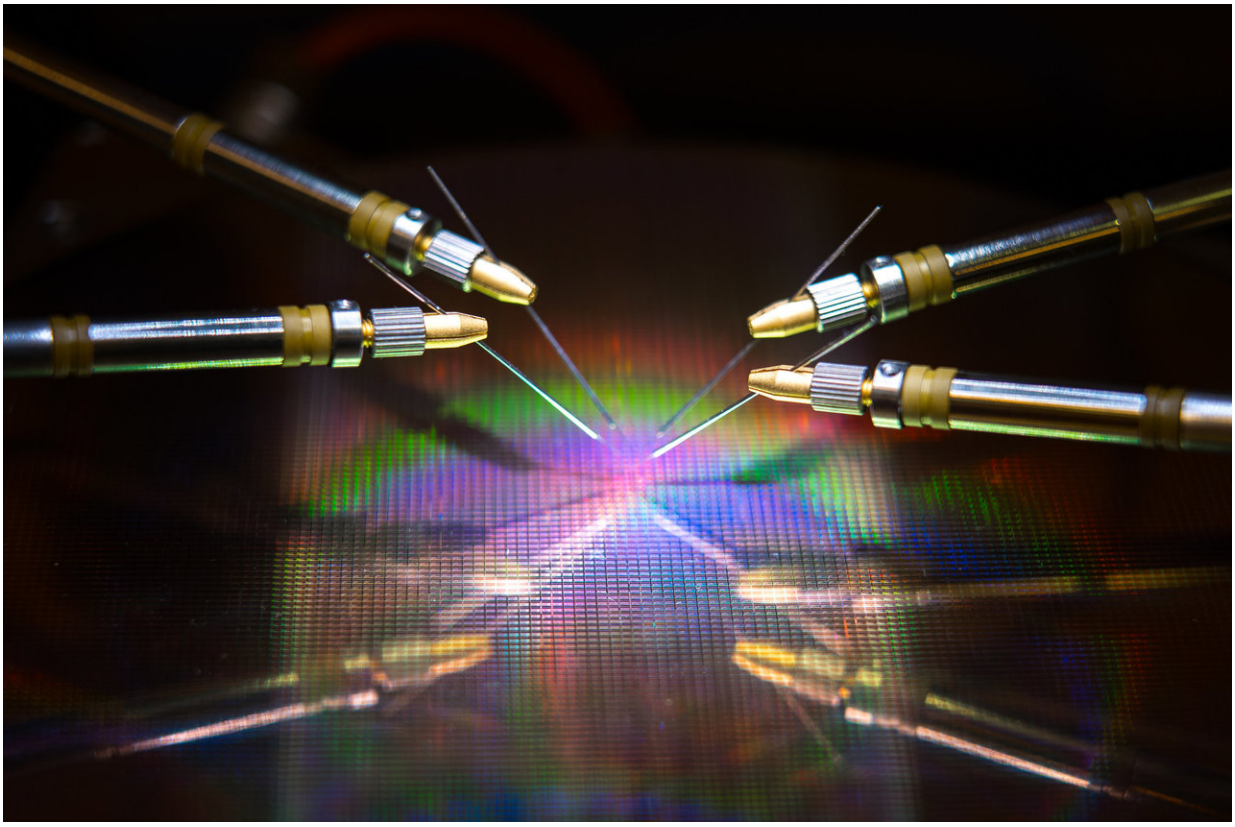
Hardware-software co-design approach could make neural networks less power hungry

December 19 2018



A UC San Diego-led team has developed hardware and algorithms that could cut energy use and time when training a neural network. Credit: David Baillot/UC San Diego Jacobs School of Engineering

A team led by the University of California San Diego has developed a neuroinspired hardware-software co-design approach that could make

neural network training more energy-efficient and faster. Their work could one day make it possible to train neural networks on low-power devices such as smartphones, laptops and embedded devices.

The advance is described in a paper published recently in *Nature Communications*.

Training neural networks to perform tasks like recognize objects, navigate self-driving cars or play games eats up a lot of computing power and time. Large computers with hundreds to thousands of processors are typically required to learn these tasks, and training times can take anywhere from weeks to months.

That's because doing these computations involves transferring data back and forth between two separate units—the memory and the processor—and this consumes most of the energy and time during neural network training, said senior author Duygu Kuzum, a professor of electrical and computer engineering at the Jacobs School of Engineering at UC San Diego.

To address this problem, Kuzum and her lab teamed up with Adesto Technologies to develop hardware and algorithms that allow these computations to be performed directly in the memory unit, eliminating the need to repeatedly shuffle data.

"We are tackling this problem from two ends—the device and the algorithms—to maximize energy efficiency during neural network training," said first author Yuhan Shi, an electrical engineering Ph.D. student in Kuzum's research group at UC San Diego.

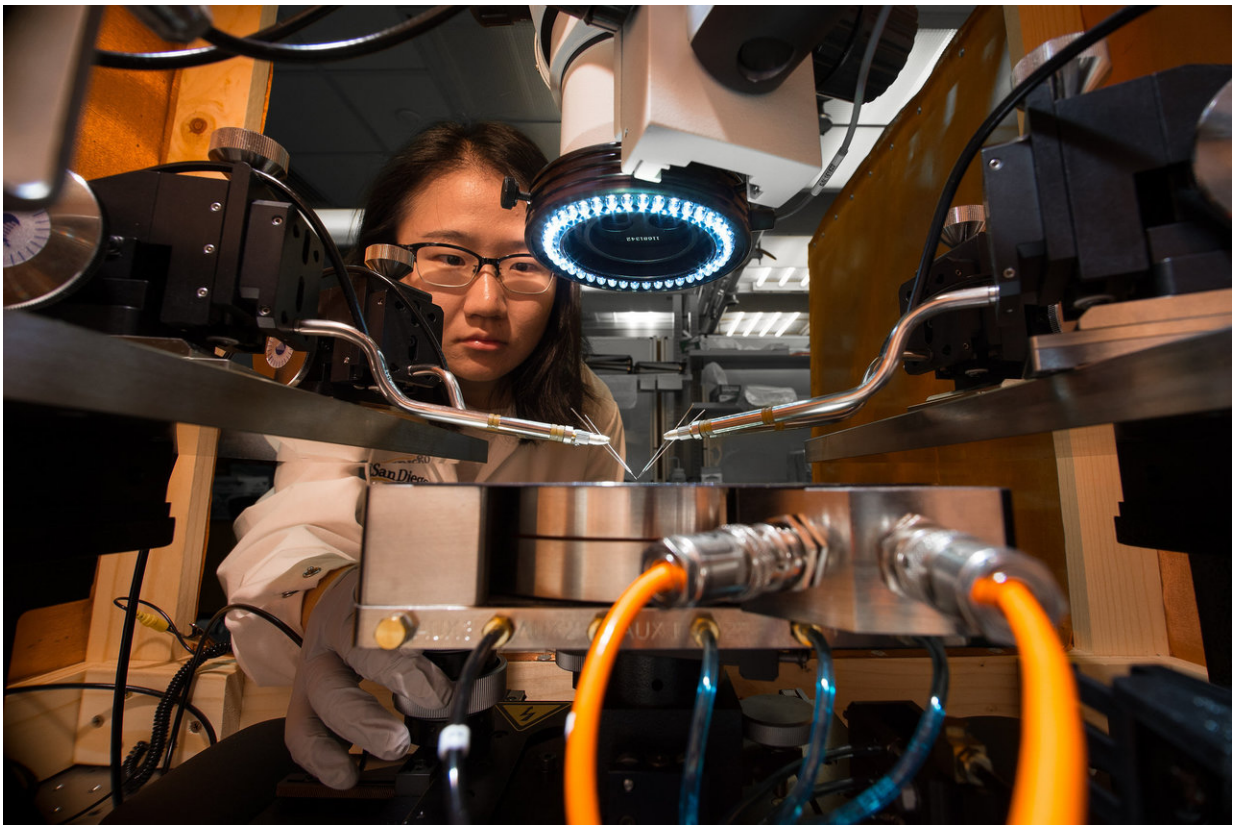
The hardware component is a super energy-efficient type of non-volatile memory technology—a 512 kilobit subquantum Conductive Bridging RAM (CBRAM) array. It consumes 10 to 100 times less energy than

today's leading memory technologies. The device is based on Adesto's CBRAM memory technology—it has primarily been used as a digital storage device that only has '0' and '1' states, but Kuzum and her lab demonstrated that it can be programmed to have multiple analog states to emulate biological synapses in the human brain. This so-called synaptic device can be used to do in-memory computing for neural network training.

"On-chip memory in conventional processors is very limited, so they don't have enough capacity to perform both computing and storage on the same chip. But in this approach, we have a high capacity memory array that can do computation related to neural network training in the memory without data transfer to an external processor. This will enable a lot of performance gains and reduce energy consumption during training," said Kuzum.

Kuzum, who is affiliated with the Center for Machine-Integrated Computing and Security at UC San Diego, led efforts to develop algorithms that could be easily mapped onto this synaptic device array. The algorithms provided even more energy and time savings during neural network training.

The approach uses a type of energy-efficient neural network, called a spiking neural network, for implementing unsupervised learning in the hardware. On top of that, Kuzum's team applies another energy-saving algorithm they developed called "soft-pruning," which makes neural network training much more energy efficient without sacrificing much in terms of accuracy.



Yuhan Shi sets up the synaptic device array for testing. Credit: University of California - San Diego

Energy-saving algorithms

Neural networks are a series of connected layers of artificial neurons, where the output of one layer provides the input to the next. The strength of the connections between these layers is represented by what are called "weights." Training a neural network deals with updating these weights.

Conventional neural networks spend a lot of energy to continuously update every single one of these weights. But in spiking neural networks, only weights that are tied to spiking neurons get updated. This means fewer updates, which means less computation power and time.

The network also does what's called unsupervised learning, which means it can essentially train itself. For example, if the network is shown a series of handwritten numerical digits, it will figure out how to distinguish between zeros, ones, twos, etc. A benefit is that the network does not need to be trained on labeled examples—meaning it does not need to be told that it's seeing a zero, one or two—which is useful for autonomous applications like navigation.

To make training even faster and more energy-efficient, Kuzum's lab developed a new algorithm that they dubbed "soft-pruning" to implement with the unsupervised spiking neural network. Soft-pruning is a method that finds weights that have already matured during training and then sets them to a constant non-zero value. This stops them from getting updated for the remainder of the training, which minimizes computing power.

Soft-pruning differs from conventional pruning methods because it is implemented during training, rather than after. It can also lead to higher accuracy when a neural network puts its training to the test. Normally in pruning, redundant or unimportant weights are completely removed. The downside is the more weights you prune, the less accurate the network performs during testing. But soft-pruning just keeps these weights in a low energy setting, so they're still around to help the network perform with higher accuracy.

Hardware-software co-design to the test

The team implemented the neuroinspired unsupervised spiking neural network and the soft-pruning algorithm on the subquantum CBRAM synaptic device array. They then trained the network to classify handwritten digits from the MNIST database.

In tests, the network classified digits with 93 percent accuracy even

when up to 75 percent of the weights were soft pruned. In comparison, the network performed with less than 90 percent accuracy when only 40 percent of the weights were pruned using conventional pruning methods.

In terms of energy savings, the team estimates that their neuroinspired hardware-software co-design approach can eventually cut energy use during neural network training by two to three orders of magnitude compared to the state of the art.

"If we benchmark the new hardware to other similar memory technologies, we estimate our device can cut energy consumption 10 to 100 times, then our algorithm co-design cuts that by another 10. Overall, we can expect a gain of a hundred to a thousand fold in terms of energy consumption following our approach," said Kuzum.

Moving forward, Kuzum and her team plan to work with memory technology companies to advance this work to the next stages. Their ultimate goal is to develop a complete system in which neural networks can be trained in memory to do more complex tasks with very low power and time budgets.

More information: Yuhan Shi et al, Neuroinspired unsupervised learning and pruning with subquantum CBRAM arrays, *Nature Communications* (2018). DOI: 10.1038/s41467-018-07682-0

Provided by University of California - San Diego