



# INSIDE AN AI 'BRAIN' - WHAT DOES MACHINE LEARNING LOOK LIKE?

Written by  
Matt Fyles

Posted  
6 months ago

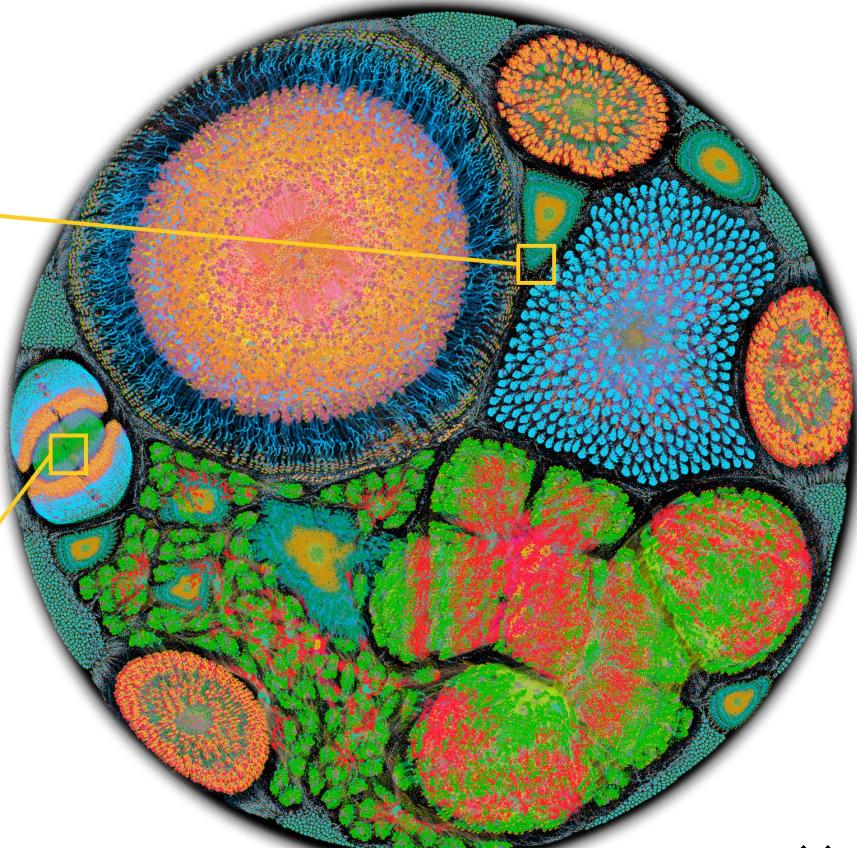
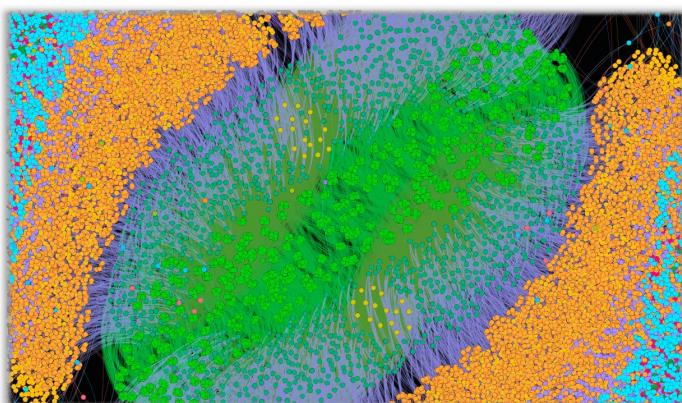
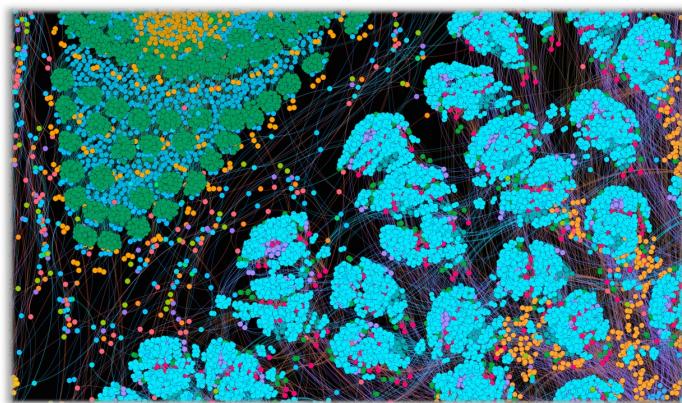
One aspect all recent machine learning frameworks have in common - TensorFlow, MxNet, Caffe, Theano, Torch and others - is that they use the concept of a computational graph as a powerful abstraction.

A graph is simply the best way to describe the models you create in a machine learning system. These computational graphs are made up of vertices (think neurons) for the compute elements, connected by edges (think synapses), which describe the communication paths between vertices.

Unlike a scalar CPU or a vector GPU, the Graphcore Intelligent Processing Unit (IPU) is a graph processor. A computer that is designed to manipulate graphs is the ideal target for the computational graph models that are created by machine learning frameworks.

We've found one of the easiest ways to describe this is to visualize it. Our software team has developed an amazing set of images of the computational graphs mapped to our IPU. These images are striking because they look so much like a human brain scan once the complexity of the connections is revealed – and they are incredibly beautiful too.

### A machine learning model used in astrophysics data analysis:



We're looking for Engineers to join our team – [find out more](#)



Before explaining what we are looking at in these images, it's useful to understand more about our software framework, Poplar™ which visualizes graph computing in this way.

Poplar is a graph programming framework targeting IPU systems, designed to meet the growing needs of both advanced research teams and commercial deployment in the enterprise. It's not a new language, it's a C++ framework which abstracts the graph-based machine learning development process from the underlying graph processing IPU hardware.

We're also building a comprehensive, open source set of Poplar graph libraries for machine learning. In essence, this means existing user applications written in standard machine learning frameworks, like TensorFlow and MXNet, will work out of the box on an IPU. It will also be a natural basis for future machine intelligence programming paradigms which extend beyond tensor-centric deep learning. Poplar has a full set of debugging and analysis tools to help tune performance and a C++ and Python interface for application development if you need to dig a bit deeper.

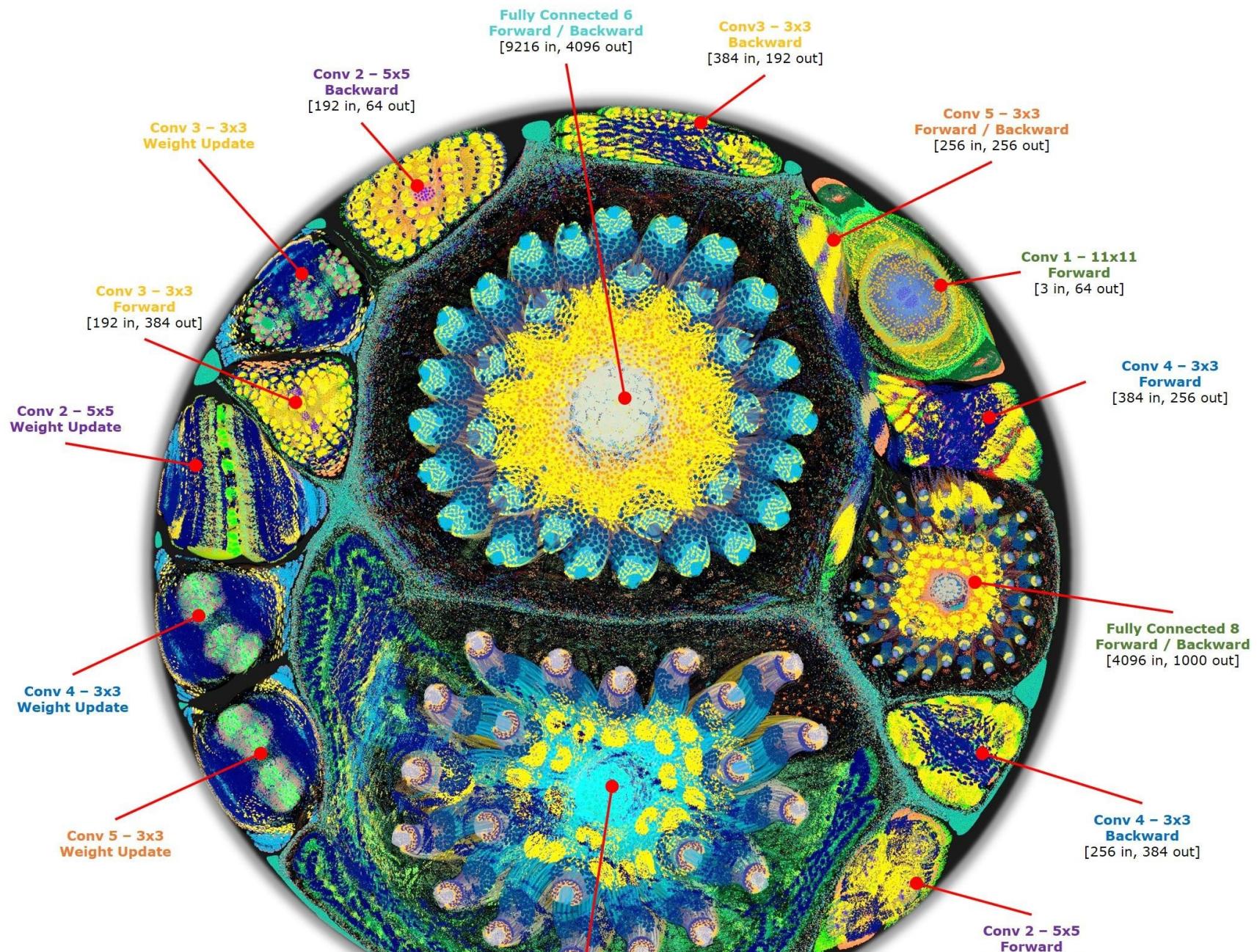
We have designed it to be extensible; the IPU will accelerate today's deep learning applications, but the combination of Poplar and IPU provides access to the full richness of the computational graph abstraction for future innovation.

Poplar includes a graph compiler which has been built from the ground up for translating the standard operations used by machine learning frameworks into highly optimized application code for the IPU. The graph compiler builds up an intermediate representation of the computational graph to be scheduled and deployed across one or many IPU devices. The compiler can display this computational graph, so an application written at the level of a machine learning framework reveals an image of the computational graph which runs on the IPU.

The image below shows the graph for the full forward and backward training loop of AlexNet.  
We're looking for Engineers to join our team –[find out more](#)



generated from a TensorFlow description.



We're looking for Engineers to join our team – [find out more](#)



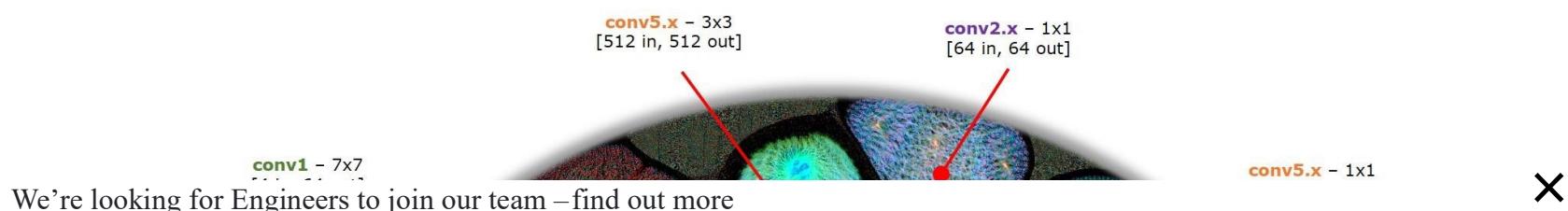


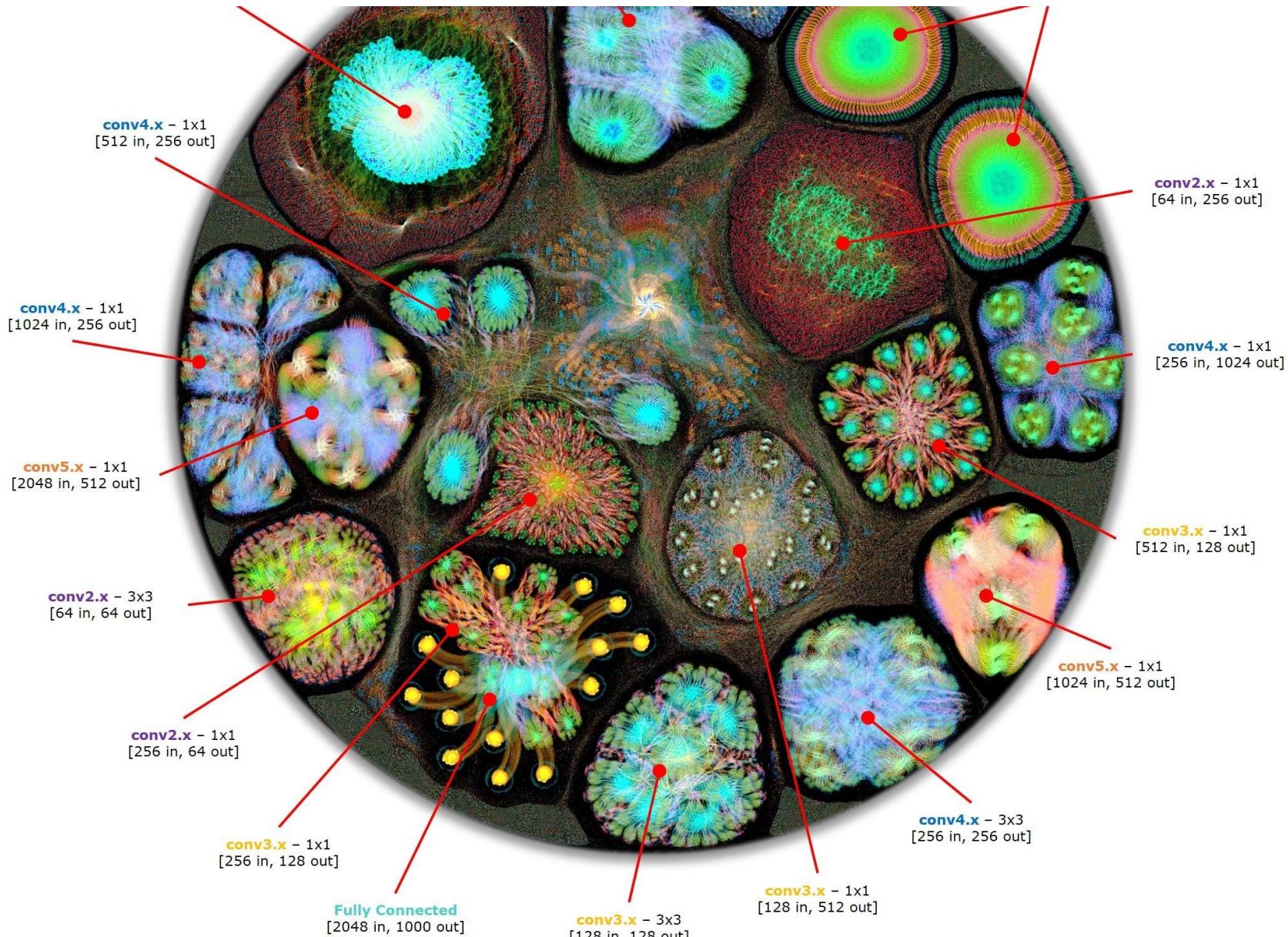
## Alexnet: deep neural network

The AlexNet architecture is a powerful deep neural network (DNN) which uses convolutional and fully-connected layers as its building blocks. It rose to fame in 2012 when it won first place for image classification in the ImageNet Large Scale Visual Recognition Competition (ILSVRC). It was a breakthrough architecture, achieving a large accuracy margin over non-DNN models.

Our Poplar graph compiler has converted a description of the network into a computational graph of 18.7 million vertices and 115.8 million edges. This graph represents AlexNet as a highly-parallel execution plan for the IPU. The vertices of the graph represent computation processes and the edges represent communication between processes. The layers in the graph are labelled with the corresponding layers from the high level description of the network. The clearly visible clustering is the result of intensive communication between processes in each layer of the network, with lighter communication between layers.

The graph itself is heavily dominated by the fully connected layers towards the end of the network structure, which can be seen in the centre and to the right of the image, marked as Fully Connected 6, Fully Connected 7 and Fully Connected 8. These fully connected layers distinguish AlexNet from more recent architectures such as ResNet from Microsoft Research, an example of which is shown below.





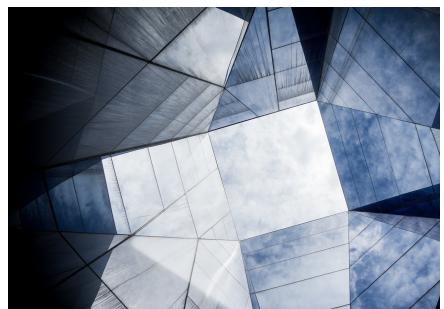
## Resnet 50: deep neural network

We're looking for Engineers to join our team – [find out more](#)

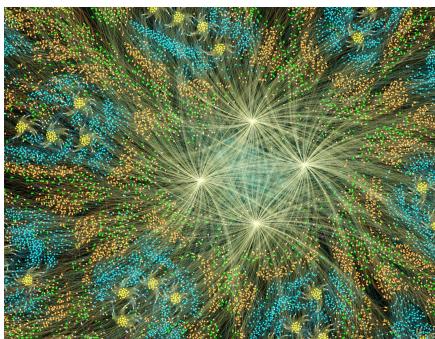


A graph processor such as the IPU is designed specifically for building and executing computational graph networks for deep learning and machine learning models of all types. What's more, the whole model can be hosted on an IPU. This means IPU systems train machine learning models much faster than, and deploy them for inference or prediction much more efficiently than other processors which were simply not designed for this new and important workload. Machine learning is the future of computing and a graph processor like the IPU is the architecture that will carry this next wave of computing forward.

## Read Next



Why is so much memory needed for deep neural networks?



Graph computing for machine intelligence  
with Poplar™

We're looking for Engineers to join our team –[find out more](#)



Sign up to blog updates

We're looking for Engineers to join our team –[find out more](#)

