Figure 9-1. A CPU from Intel. CPUs are still the dominant form of computer hardware and are present in all modern laptops, desktops, servers, and phones. Most software is written to execute on CPUs. Numerical computations (such as neural network training) can be executed on CPUs, but might be slower than on customized hardware optimized for numerical methods.

GPU Training

GPUs were first developed to perform computations needed by the graphics community. In a fortuitous coincidence, it turned out that the primitives used to define graphics shaders could be repurposed to perform deep learning. At their mathematical hearts, both graphics and machine learning rely critically on matrix multiplications. Empirically, GPU matrix multiplications offer speedups of an order of magnitude or two over CPU implementations. How do GPUs succeed at this feat? The trick is that GPUs make use of thousands of identical threads. Clever hackers have succeeded in decomposing matrix multiplications into massively parallel operations that can offer dramatic speedups. Figure 9-2 illustrates a GPU architecture.

Although there are a number of GPU vendors, Nvidia currently dominates the GPU market. Much of the power of Nvidia's GPUs stems from its custom library CUDA (compute unified device architecture), which offers primitives that make it easier to write GPU programs. Nvidia offers a CUDA extension, CUDNN, for speeding up deep networks (Figure 9-2). TensorFlow has built-in CUDNN support, so you can make use of CUDNN to speed up your networks as well through TensorFlow.

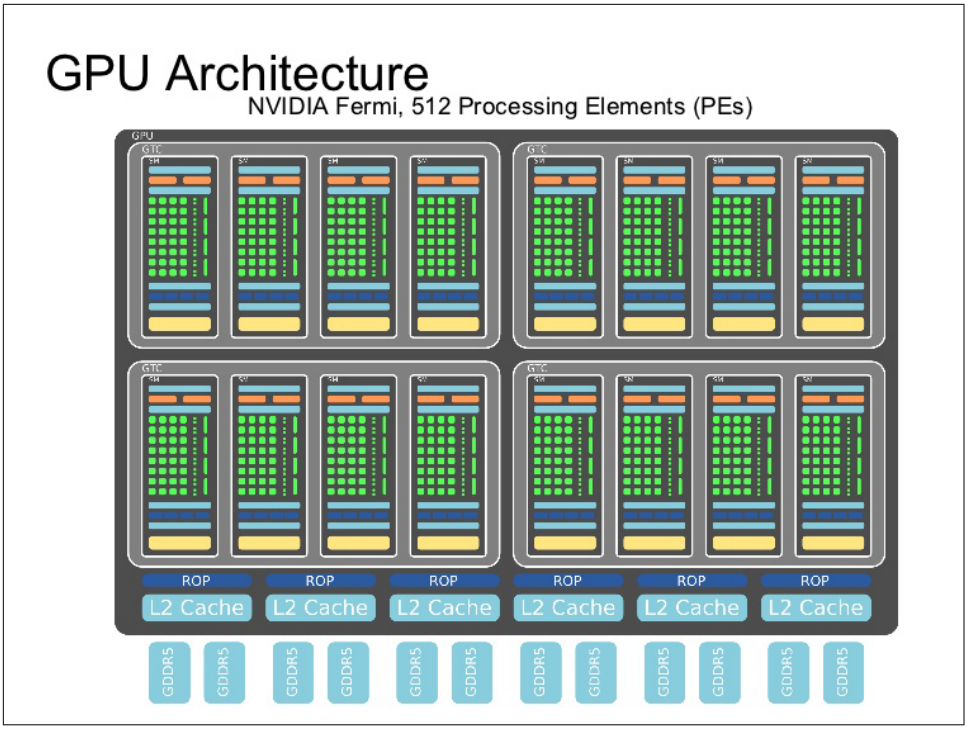

Figure 9-2. A GPU architecture from Nvidia. GPUs possess many more cores than CPUs and are well suited to performing numerical linear algebra, of the sort useful in both graphics and machine learning computations. GPUs have emerged as the dominant hardware platform for training deep networks.

How Important Are Transistor Sizes?

For years, the semiconductor industry has tracked progression of chip speeds by watching transistor sizes. As transistors got smaller, more of them could be packed onto a standard chip, and algorithms could run faster. At the time of writing of this book, Intel is currently operating on 10-nanometer transistors, and working on transitioning down to 7 nanometers. The rate of shrinkage of transistor sizes has slowed significantly in recent years, since formidable heat dissipation issues arise at these scales.

Nvidia's GPUs partially buck this trend. They tend to use transistor sizes a generation or two behind Intel's best, and focus on solving architectural and software bottlenecks instead of transistor engineering. So far, Nvidia's strategy has paid dividends and the company has achieved market domination in the machine learning chip space.

It's not yet clear how far architectural and software optimizations can go. Will GPU optimizations soon run into the same Moore's law roadblocks as CPUs? Or will clever architectural innovations enable years of faster GPUs? Only time can tell.

Tensor Processing Units

The tensor processing unit (TPU) is a custom ASIC (application specific integrated circuit) designed by Google to speed up deep learning workloads designed in TensorFlow. Unlike the GPU, the TPU is stripped down and implements only the bare minimum on-die needed to perform necessary matrix multiplications. Unlike the GPU, the TPU is dependent on an adjoining CPU to do much of its preprocessing work for it. This slimmed-down approach enables the TPU to achieve higher speeds than the GPU at lower energy costs.

The first version of the TPU only allowed for inference on trained models, but the most recent version (TPU2) allows for training of (certain) deep networks as well. However, Google has not released many details about the TPU, and access is limited to Google collaborators, with plans to enable TPU access via the Google cloud. Nvidia is taking notes from the TPU, and it's quite likely that future releases of Nvidia GPUs will come to resemble the TPU, so downstream users will likely benefit from Google's innovations regardless of whether Google or Nvidia wins the consumer deep learning market. Figure 9-3 illustrates the TPU architecture design.