

BY JOHN MELONAKOS



Parallel Computing on a Personal Computer

Anyone who has ever waited minutes, hours, or even days for software to complete a biomedical computation will be happy to hear that almost every personal computer is capable of better. Today, most standard PCs, both desktops and laptops, come with a graphics processing unit (GPU) in addition to the central processing unit (CPU). And, thanks to the video gaming market, GPU hardware has advanced at a much faster pace than CPU hardware. In fact, GPUs have advanced so quickly that today they have ten times more computational power than CPUs (see the graph).

Why are GPUs faster than CPUs for most biomedical computations? A CPU is a serial computing device, processing data sequentially. A GPU is a parallel computing device, processing many chunks of data all at the same time. Since most biomedical computations are parallelizable, GPU computing provides a powerful alternative to traditional CPU computing without the expense of purchasing a room full of clustered computers.

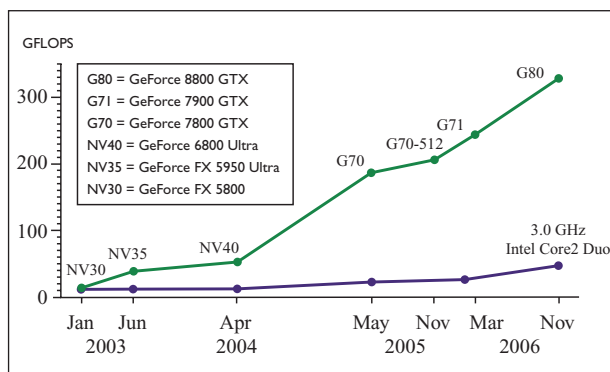
The big bottleneck for GPU computing is writing software specialized for the GPU. Since GPU computing is in its infancy relative to CPU computing, only a small fraction of programmers around the world are familiar with GPU-based programming languages such as CUDA, Brook+, or Ct. GPU software developers must scale a serious learning curve if GPUs are to serve the mainstream.

To solve this problem, easy-to-use GPU programming toolboxes are now available, such as the Matlab-based one from AccelerEyes (see the framework). The utility of these tools is to help researchers tap into the benefits of GPU computing.

But GPUs are already having an impact in biomedical computing. Examples include image-guided brain surgery, molecular dynamics simulations, and genomics.

Complex algorithms which take hours when run on a CPU can now be used in real-time. And the computing power made available by a GPU on a standard PC now costs hundreds of times less than that of a cluster of PCs having similar computing power.

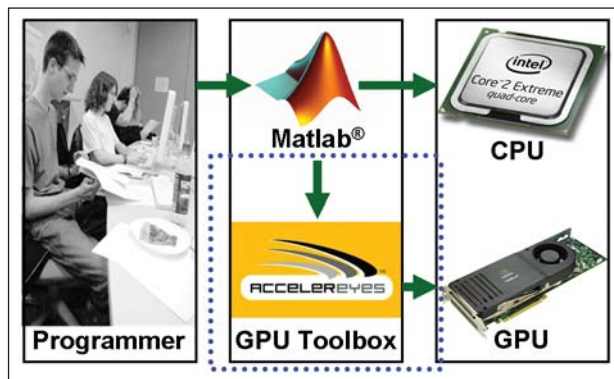
With these kinds of speed improvements and cost benefits, GPU programming is sure to become mainstream. It's clearly faster than running software on your CPU (especially when that same computer already has the hardware necessary to go faster); and it's clearly cheaper than buying a room full of clustered computers. Now the software world just needs to catch up. □



Over the last few years, the power of GPUs had increased dramatically compared to that of CPUs, as shown in this chart comparing NVidia graphics processors with Intel processors.

DETAILS

John Melonakos, a PhD student at Georgia Tech, is an active participant in the National Alliance for Medical Image Computing (NA-MIC), one of the National Centers for Biomedical Computing. He joined with Tauseef ur Rehman, Gallagher Pryor, and James Malcolm to start AccelerEyes LLC, which is developing technologies that enable CPU-based code to run on GPUs. The AccelerEyes Jacket Product, connecting Matlab to the GPU, is available by visiting www.accelereyes.com. For more information or to inquire about joining the AccelerEyes team, please send an email to: john.melonakos@accelereyes.com.



AccelerEyes is a new programming tool that allows researchers to use GPUs for Matlab tasks. Courtesy of John Melonakos.