

Compute Unified Device Architecture

Andrei Costa

Universidade Federal de Pelotas

14 de Agosto de 2013

- A Performance Study of General-Purpose Applications on Graphics Processors Using CUDA
- Shuai Che, Michael Boyer, Jiayuan Meng, David Tarjan, Jeremy W. Sheaffer, Kevin Skadron
- University of Virginia, Department of Computer Science, Charlottesville, VA, USA
- Journal of Parallel and Distributed Computing, Elsevier

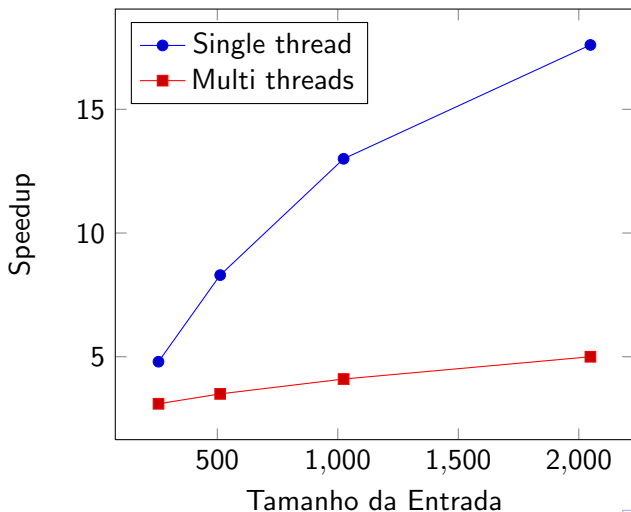
- Unidades de Processamentos Gráficos (GPUs):
 - Simplicidade
 - Paralelização de dados
 - Multithreaded cores
 - Alta banda de memória

- Utilizar CUDA (GTX 260 GPU) para explorar a eficácia de GPUs
- Comparar com single-core e multi-core (OpenMP) CPUs
- Discutir vantagens e ineficiências do modelo de programação CUDA

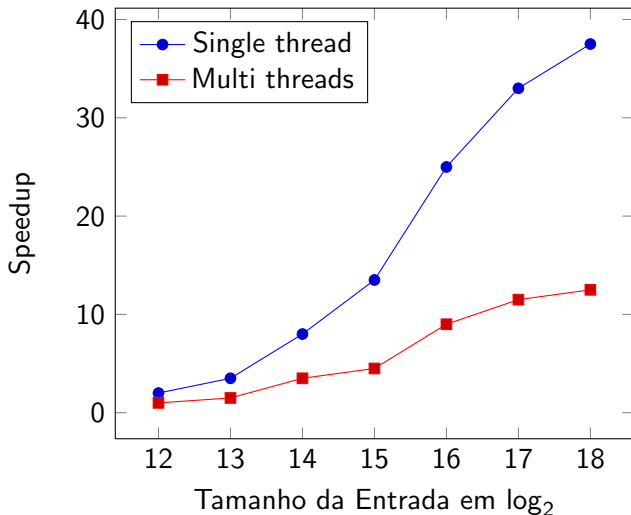
- falar do pc

Speckle Reducing Anisotropic Diffusion (SRAD)

- Método para remover ruído de imagens de ultra-som ou de radar aplicando equações diferenciais parciais



- Várias operações no nível lógico



Modelo de Programação CUDA

- Threads são escaláveis, e seu kernel é um escalonador
 - Acessos a memória não precisam ser contíguos, embora acessos não contíguos reduzirem a banda de memória
- CUDA não é um modelo puramente data-parallel
- Sincronizações são muito ineficientes
 - Pode ser atenuado com tarefas de pouca granulosidade
- O programador especifica explicitamente o número de threads, e estas não podem gerar dinamicamente novas threads

- Melhora expressiva de desempenho, principalmente com granulosidade pequena
- Faltam APIs para programação em alto nível