

# Análise comparativa do desempenho de filtros digitais de imagens com a plataforma OpenCL na GPU Nvidia e ATI

Darlisson Marinho de Jesus<sup>1</sup>, Raimundo Correa Oliveira

<sup>1</sup> Universidade do Estado do Amazonas – UEA

Av. Darcy Vargas, 1200, Parque 10 de Novembro – 69.065.020 – Manaus – AM – Brasil

{darlisson11, rcorrea.oliveira}@gmail.com

**Abstract.** *Graphics Processing Units (GPU) are high performance co-processors intended, originally, to improve the use and quality of computer graphics applications. However, the GPU applications has been extended to other fields to general purpos, such as digital image processing. The main objective of this study is to compare the performance of digital filters to images in the context of GPUs from Nvidia and ATI vendors. To achieve this, we will implement these digital filters with OpenCL platform, which allows the writing of code for running cross-platform on these GPUs.*

**Resumo.** *Unidades de processamento gráfico ou GPUs são co-processadores de alto desempenho destinados inicialmente a melhorar ou prover de capacidade gráfica um computador. Porém, as aplicações das GPUs tem sido expandida à outras áreas para fins gerais, como o processamento digital de imagens. O principal objetivo deste trabalho é comparar o desempenho de filtros digitais de imagens no contexto das GPUs das fabricantes Nvidia e ATI. Para conseguir isso, implementaremos esses filtros digitais com a plataforma OpenCL, a qual, permite a escrita de código multi-plataforma para execução nestas GPUs.*

## 1. Descrição do Problema

Alguns dos grandes desafios da área de processamento de imagens são a disponibilidade e o custo financeiro do acesso a um dispositivo com poder computacional suficiente para atingir grandes velocidades na execução de sistemas ligados à visão computacional e à interação homem-computador. Imagens e vídeos digitais são em essência sinais multidimensionais e, portanto, possuem um grande volume de dados [?]. O processamento paralelo pode ajudar a diminuir o tempo para o processamento deste grande volume de dados através da divisão do problema em problemas menores, os quais podem ser executados por múltiplos núcleos de processamento. Como resultado da recente evolução dos processadores gráficos, pesquisadores e desenvolvedores formaram uma comunidade interessada em usufruir do poder destes processadores no contexto da computação científica. Essa iniciativa ficou conhecida como GPGPU (General-Purpose Computation on Graphics Processing Unit), e os fabricantes de placas de vídeo reconhecendo este potencial de mercado, começaram

a investir em APIs e linguagens de programação de alto nível para permitir o desenvolvimento de aplicativos baseados em GPU. Entre as principais características das GPUs modernas, destacam-se o acesso rápido à memória de vídeo, clock de alta frequência e múltiplos pipelines programáveis [?]. Contudo, surge a necessidade de comparar o desempenho computacional dessas placas gráficas de diferentes fabricantes, e assim, usar essas informações como uma ferramenta para auxiliar engenheiros e cientistas na busca do melhor desempenho, aliado ao menor custo das tecnologias com o passar do tempo. Existem duas principais plataformas utilizadas para atender a demanda de desenvolvimento de aplicações não gráficas pela comunidade GPGPU. A primeira, o CUDA (Compute Unified Device Architecture) [?], foi desenvolvida pela pioneira NVIDIA, apesar de ser amplamente utilizada no mercado essa plataforma é restrita para as placas gráficas dessa fabricante, e não permite a portabilidade das aplicações para GPUs de outras fabricantes. A portabilidade da aplicação é um dos requisitos obrigatórios para que possamos comparar o desempenho de GPUs de diferentes fabricantes, por isso, usaremos a plataforma OpenCL [?] como recurso para as implementações, pois com ela é possível escrever programas para plataformas heterogêneas. Além disso, ela possui uma linguagem que é baseada na linguagem de programação C, está sendo proposta como um padrão aberto.

## 2. Trabalhos Relacionados

Este trabalho faz contribuições interessantes, pois até o presente momento não se tem registro de uma análise comparativa do desempenho de uma aplicação com o OpenCL entre placas gráficas da Nvidia e ATI/AMD. Além disso, este trabalho contém implementações de filtros digitais para imagens em GPU usando o OpenCL. Purcell [5] realizou uma das primeiras experiências envolvendo GPUs programáveis ao explicar a modelagem de um algoritmo de ray tracing para GPUs. Seu trabalho demonstrou que o algoritmo escrito em assembly para a NVIDIA GeForce 3, no processador de fragmentos, conseguia ter um desempenho maior em GPU. Este trabalho publicado nos primórdios da era GPGPU demonstrou o potencial do hardware gráfico programável como um dispositivo para computação genérica e paralela. Os autores em [Karimi et al. 2010], compararam o desempenho do CUDA em contraste com o OpenCL através da implementação de uma aplicação científica com elevado custo computacional, simulação Monte Carlo de um sistema de spin, para ambas plataformas. Eles mediram e compararam o tempo de transferência de dados da GPU para a CPU e vice-versa, os tempos de execução dos kernels e os tempos de execução do ciclo de vida completo das execuções, tanto para o CUDA quanto para o OpenCL. Neste trabalho os testes restringiram-se a uma GPU da própria Nvidia e os autores concluíram que o CUDA é a melhor escolha para aplicações de alta performance. Em, [7] os autores propuseram algoritmos eficientes para o processamento linear de imagem, explorando as extensões fornecidas SIMD em processadores AMD e Intel. O trabalho experimental e os resultados obtidos sugeriram implementações baseadas em OpenCL proporcionaram uma média de 1,8 vezes o rendimento mais baixo do que implementações equivalentes que utilizam diretamente o SIMD intrínsecos suportado pelo compilador Intel. O interessante neste trabalho é a aplicação heterogênea que se pode fazer com o OpenCL, usado tanto para GPUs quanto para processadores multi-cores. Os trabalhos atuais para o processamento com GPU

usando o OpenCL, restringem-se a aplicações para as placas gráficas da Nvidia ou processadores multicore Intel ou AMD, mas não foi investigado pelos outros trabalhos, o comportamento de desempenho do OpenCL em placas de fabricantes diferentes em uma abordagem comparativa.

### **3. Objetivo Geral**

Comparar o desempenho das GPUs ( Graphics Processing Unit) das fabricantes NVIDIA e ATI, por meio do Processamento Digital de Imagens com os filtros digitais da Transformada Rápida de Fourier e do filtro para Detecção de Borda Canny. Assim, identificaremos qual GPU apresenta o melhor desempenho quando se utiliza a plataforma OpenCL (Open Computing Language) como recurso computacional para a implementação destes filtros.

#### **3.1. Objetivos Específicos**

- Identificar os indicadores de desempenho que permitam avaliar comportamento dos filtros digitais para comparação posterior;
- Avaliar o paradigma de programação das GPUs da Nvidia e ATI/AMD, buscando identificar as diferenças que podem afetar no desempenho das implementações;
- Analisar o algoritmo da FFT, afim de indentificar as instruções que poderão ser paralelizáveis e que se tornarão em um kernel no contexto do openCL;
- Analisar o algoritmo do filtro para detecção de borda Canny, afim de indentificar as instruções que poderão ser paralelizáveis e que se tornarão em um kernel no contexto do openCL;
- Realizar testes de desempenho com estas implementações na GPU Nvidia e ATI/AMD;
- Avaliar os resultados dos testes da GPU Nvidia em contraste com os resultados da GPU ATI/AMD, de acordo com os indicadores de desempenho definidos.

### **4. Metodologia**

Os procedimentos adotados para atingir os objetivos propostos serão formados de duas etapas: pesquisa bibliográfica e pesquisa experimental. A pesquisa bibliográfica será constituída de material publicado em periódicos, jornais, revistas e congressos, e também a partir de livros. Neste trabalho será conduzido uma revisão bibliográfica a cerca do tema Processamento Digital de Imagens a fim de esclarecer os principais conceitos de filtragem digital, tanto no domínio do tempo, quanto no domínio da frequência, e especificamente quanto aos filtros implementados neste trabalho, o filtro da Transformada Rápida de Fourier e o filtro de detecção de borda Canny. A etapa seguinte será o estudo de Modelos de Computação Paralela, onde identificaremos qual a classificação da arquitetura de processamento computacional usaremos, de acordo com a Taxionomia de Flynn[2], é uma etapa importante, pois, sabendo essa classificação entenderemos melhor como deverá acontecer a concorrência dos processos e o controle de fluxo da memória. Além disso, a Lei de Amdahl[3], também, deverá ser alvo de estudo, uma vez que ela nos permite identificar as barreiras pra atingir um melhor desempenho em programas paralelos. A pesquisa, quanto aos

objetivos, conduz a uma experimentação, desta forma caracterizando-se como pesquisa experimental, assim, será necessário um estudo profundo das especificações da plataforma OpenCL, tanto da linguagem, API, runtime e modelos de programação. Assim como, das arquiteturas das GPUs Nvidia e ATI. Usaremos dois computadores, com as mesmas configurações de processador, memória, etc, para que tenhamos ambientes semelhantes para os testes. Um dos computadores terá uma placa gráfica da Nvidia e o outro será equipado com uma placa gráfica ATI/AMD. Esses computadores pertencem ao Laboratório de Engenharia de Computação da sala B9. Para as implementações dos filtros, usaremos o Microsoft Visual Studio 2010 professional com a licença para estudante para realizar as implementações dos filtros digitais em OpenCL no sistema operacional Microsoft Windows. Os experimentos contarão com um determinado número de amostras, que buscam garantir uma precisão mais alta nos resultados finais. Por fim, faremos uma análise e interpretação dos resultados para avaliar o desempenho dos filtros digitais, e determinar em qual GPU eles possuem o melhor desempenho.

## **5. Cronograma**

A seguir, o cronograma para realização das tarefas.

## **Referências**

- [Karimi et al. 2010] Karimi, K., Dickson, N. G., and Hamze, F. (2010). A performance comparison of cuda and opencl. *CoRR*, abs/1005.2581.