

# Implementação de algoritmos de processamento de imagens digitais utilizando o padrão OpenCL

Daniela Bonfatti. danielabnf.tva@gmail.com

Tiago Alexandre Dócusse. tad@ifsp.edu.br

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo Câmpus Barretos

Palavras Chave: *Processamento de Imagens, Programação Paralela, Computação Heterogênea*

## Introdução

A computação heterogênea consiste em utilizar diversas arquiteturas em um mesmo ambiente computacional, visando melhorar o desempenho das aplicações que serão executadas neste ambiente [1]. O *Open Computing Language* (OpenCL) é um padrão de programação de código aberto e livre de *royalties* com foco em sistemas heterogêneos, que possibilita a implantação de algoritmos de forma a serem executados em sistemas heterogêneos. Como operações de processamento de imagens digitais são altamente paralelizáveis [2], apresentamos neste trabalho como a utilização do OpenCL pode aumentar o desempenho destas operações.

## Objetivos

O objetivo deste trabalho é realizar a implementação de operações de processamento de imagens utilizando OpenCL e comparar o desempenho de duas execuções: uma realizada apenas no processador do computador e outra com o auxílio da placa de vídeo.

## Material e Métodos

Foi utilizada a linguagem de programação C para implementar a operação de binarização em uma imagem. Nesta operação, define-se um limiar e, para todos os *pixels* da imagem avaliada que possuem valor inferior ao limiar, atribui-se o valor 0 na imagem resultante, enquanto os *pixels* que não satisfazem essa condição possuem o valor 255 atribuído na imagem resultante. Para executar no processador do computador, o processamento da imagem foi dividido em vários *threads*, cada um trabalhando sobre uma região da imagem. Assim, a imagem é dividida em diferentes partes iguais, de acordo com a quantidade de *threads* utilizada, e cada *thread* realiza a operação em uma parte da imagem. Já ao executar essa operação na placa de vídeo, o próprio ambiente de execução do padrão OpenCL atribui, para cada unidade de computação da placa de vídeo, quais *pixels* da imagem deverão ser processados, sendo esse processamento realizado de forma paralela.

## Resultados e Discussão

Os resultados aqui discutidos foram obtidos ao utilizar um computador com processador AMD FX-8350 com 8 núcleos reais juntamente com uma placa de vídeo Nvidia GTX 970 com 1664 unidades de computação. A execução da função de binarização no processador central, para 128 imagens de tamanho 8192 x 8192 *pixels* utilizando 1, 4, 16 e 64 *threads* respectivamente, resultou nos seguintes tempos de processamento: 31s, 9s, 8s e 8.3s. Percebe-se o ganho de desempenho considerável ao utilizar mais de um *thread* de processamento, porém, ocorre a estabilização do tempo gasto de processamento, a partir do ponto em que se atinge o limite de núcleos físicos do processador. Ao executar o código utilizando o padrão OpenCL, para a mesma entrada utilizada anteriormente, verificou-se que o tempo de execução da operação foi de 0.4s. Assim, percebe-se um ganho de desempenho de 20 vezes do código executado utilizando o padrão OpenCL quando comparado ao código executado apenas no processador central do computador, devido à alta capacidade de paralelização da operação de binarização, juntamente com a maior quantidade de unidades de computação da placa de vídeo quando comparada ao número de núcleos do computador central do processador. Como este trabalho ainda está em andamento, espera-se implementar outras operações de processamento de imagens digitais para verificar se esse ganho de desempenho também se repete.

## Conclusões

O desempenho da operação de processamento de imagens digitais implementada utilizando o padrão OpenCL mostrou-se altamente superior à sua execução utilizando apenas o processador central do computador, possibilitando um menor tempo de espera para o término de sua execução.

## Agradecimentos

Os autores gostariam de agradecer ao Câmpus Barretos do IFSP pela bolsa concedida para o desenvolvimento deste trabalho.

## Bibliografia

1. STRINGHINI, D. et al. Introdução à computação heterogênea. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, XXXII, 2012, Curitiba. **Anais...** Curitiba: Sociedade Brasileira de Computação, 2012.
2. GONZALEZ, R. C., WOODS, R. E. **Processamento Digital de Imagens**. 3. ed. São Paulo: Pearson Pratices Hall, 2010. 624 p.