

# Otimizando o tempo de execução no processamento de imagens

Henrique Miyamoto e Thiago Benites

## I. CONTEXTUALIZAÇÃO

Apresentamos uma comparação de desempenho entre diferentes métodos para aplicação de brilho em uma imagem. A mesma funcionalidade foi implementada de quatro maneiras diferentes: usando múltiplas *threads*, usando multiprocessos, em uma única linha de execução, varrendo a matriz por linhas e por colunas.

*Threads*, assim como processos, são mecanismos que permitem que um programa execute ações de forma que elas pareçam estar sendo executadas ao mesmo tempo. A diferença entre eles é que *threads* possuem área de memória compartilhada, o que não ocorre com processos [1].

O objetivo desse trabalho é comparar e discutir os desempenhos de cada método a partir do tempo de execução de sistema de cada um deles.

A tabela I apresenta as sintaxes para os diferentes métodos de aplicação de brilho usados em nosso programa.

TABLE I  
SINTAXE DA LINGUAGEM DE PROGRAMAÇÃO PARA DIFERENTES MÉTODOS

Método de implementação	Sintaxe
<i>Multithreads</i>	destino.jpg = origem.jpg *float THR
Multiprocessos	destino.jpg = origem.jpg *float PRC
Varredura por linhas	destino.jpg = origem.jpg *float LIN
Varredura por colunas	destino.jpg = origem.jpg *float COL

## II. DEMONSTRAÇÃO

Para medição dos tempos de execução foi usada a função `gettimeofday` da biblioteca `time.h`. As tabelas a seguir apresentam os tempos de execução dos diferentes métodos, para imagens de diferentes tamanhos: pequena 32x32 pixels (`dora.jpg`), média 640x640 (`ssa.jpg`) e grande 2592x1944 (`demo.jpg`).

TABLE II  
TEMPOS DE EXECUÇÃO DOS MÉTODOS PARA IMAGEM PEQUENA.

Implementação	Tamanho da imagem	Tempo de sistema (ms)
Multithreads	32x32	3,7660
Multiprocessos	32x32	53,8670
Varredura por colunas	32x32	0,092000
Varredura por linhas	32x32	0,113000

TABLE III

TEMPOS DE EXECUÇÃO DOS MÉTODOS PARA IMAGEM MÉDIA.

Implementação	Tamanho da imagem	Tempo de sistema (ms)
Multithreads	640x640	31,2480
Multiprocessos	640x640	825,2130
Varredura por colunas	640x640	9,732000
Varredura por linhas	640x640	9,556000

TABLE IV

TEMPOS DE EXECUÇÃO DOS MÉTODOS PARA IMAGEM GRANDE.

Implementação	Tamanho da imagem	Tempo de sistema (ms)
Multithreads	2592x1944	101,4530
Multiprocessos	2592x1944	7188,9360
Varredura por colunas	2592x1944	169,008000
Varredura por linhas	2592x1944	147,995000

## III. ANÁLISE

Observa-se que a execução da função de brilho através de *multithreading* não apresenta desempenho estritamente superior ao com varredura simples. Para imagens pequenas, o processamento simples é mais rápido que com *multithread*. Por outro lado, quando as imagens são grandes, usar *multithreads* aumenta consideravelmente a velocidade de execução da função. Em ambos os casos, aumentar a quantidade de pixels tratados em cada *thread* melhora o desempenho.

## REFERÊNCIAS

- [1] MITCHELL, Mark; OLDHAM, Jeffrey e SAMUEL, Alex. *Advanced Linux Programming*. Indianopolis: New Riders, 2001.

\* Os códigos do projeto estão disponíveis em <https://github.com/miyamotohk/linguagem-processamento-imagem>.