

# Análise de desempenho: Processamento de imagem

Artur Leal - albs

Felipe Souza - frs3

Romero Ramsés - rrcb

- **Introdução**
- **Objetivos e Métricas**
- **Métricas de Desempenho**
- **Parâmetros Utilizados**
- **Fatores Utilizados**
- **Projetando o Experimento**
- **Resultados**

- Aplicação desenvolvida consiste em selecionar uma imagem e aplicar um algoritmo para trocar os pixels coloridos por pixels preto e cinza.
- Para cada pixel no tensor que representa a imagem, substitui-se as componentes RGB por um cinza (gerado a partir das cores originais), mantendo o valor do alfa original



- Comparar o desempenho de três versões do algoritmo de processamento de imagem:
  - Uma versão sem concorrência
  - Uma versão utilizando uma thread para cada pixel da imagem
  - Uma versão utilizando concorrência mas limitando o número total de threads
- Como métrica de desempenho, será utilizado o tempo de execução do programa *(em milissegundos)*

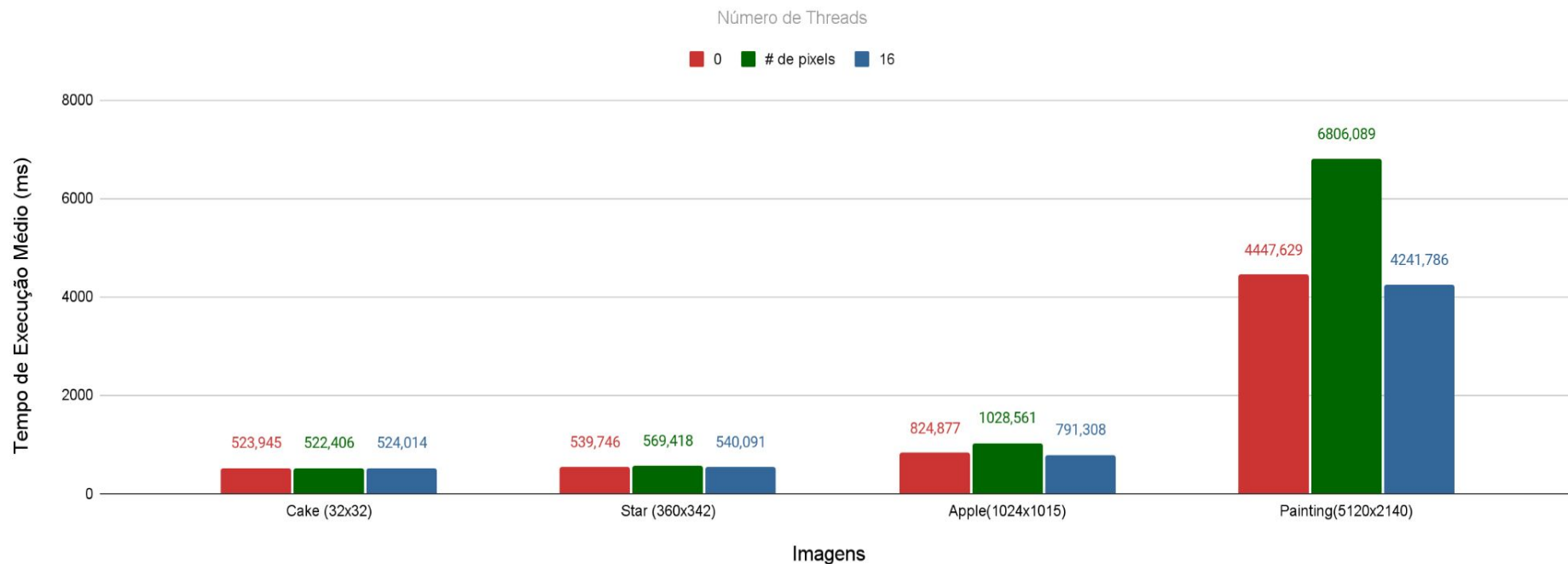
- **Parâmetros do Sistema:**
  - **CPU : AMD Ryzen 5700x (8C/16T) 4.00 GHZ**
  - **Memória: 32GB 3200Mhz DDR4**
  - **Sistema operacional: Windows 11 Home Versão 22H2**
  - **Wifi: Desligada**
  - **Configuração de Energia: Alto Desempenho (*maior estabilidade de clock*)**
- **Parâmetros da Carga de Trabalho:**
  - **Quantidade de GoRoutines**
  - **Tamanho da Imagem**

- **Fatores da Carga de Trabalho:**
  - **Quantidade de GoRoutines**
  - **Tamanho da Imagem**

<b>Quantidade de GoRoutines</b>	<b>1, 16, # de pixels</b>
<b>Tamanho da Imagem (<i>pixels</i>)</b>	<b>16 x 16, 360 x 342, 1024 x 1015, 5120 x 2160</b>

- **Técnica de Avaliação: Medição**
- **Um cliente executará o programa no terminal**
  - Usando um script auxiliar, cada experimento será executado 100 vezes, sendo computado o tempo de execução de cada uma. Ao final da centésima execução, tira-se a média aritmética dos 100 resultados e armazena esse valor como o resultado do algoritmo
  - Um experimento consiste em esse script auxiliar com cada um dos algoritmos, de forma a ter o resultado dos três algoritmos para poder compará-los
  - Serão realizados 4 experimentos, cada um utilizando uma imagem diferente
    - *Apple.png, Cake.png, Painting.png, Star.png*

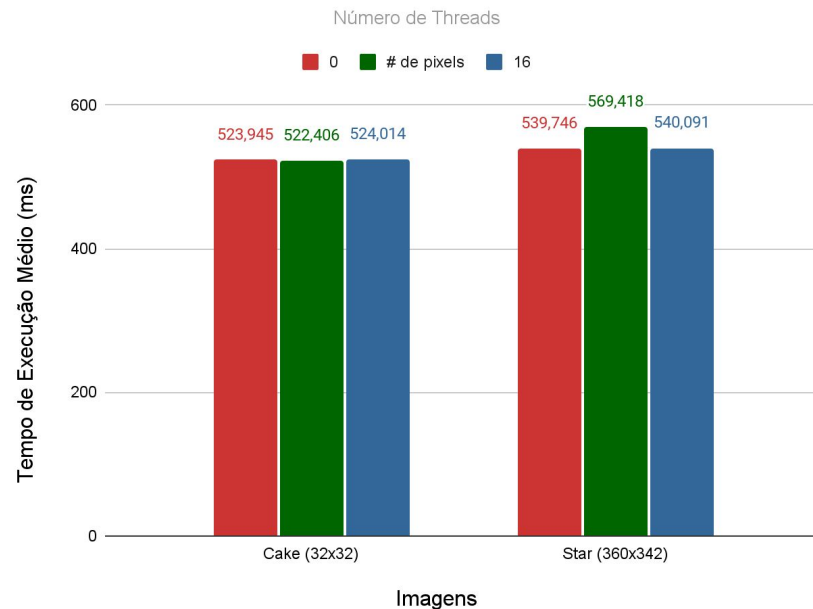
## Resultados Da Análise de Desempenho





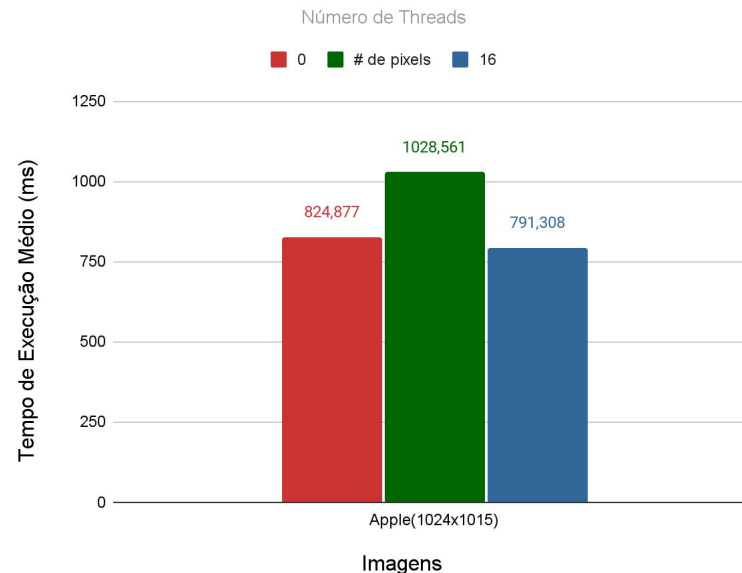
- Em imagens pequenas, não há uma diferença relevante de desempenho
- Em alguns dos testes, a versão sem threads performar melhor do que as versões com concorrência implementada

Resultados Da Análise de Desempenho ( Cake e Star)



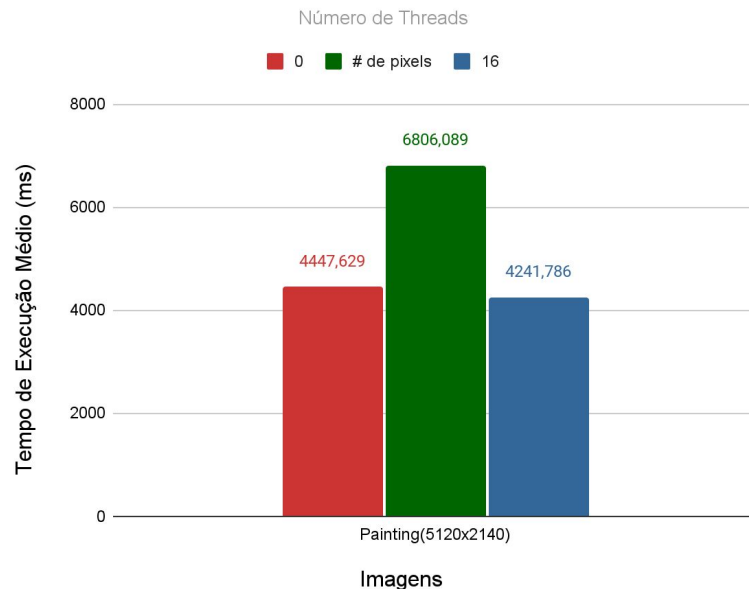
- Na imagem “Apple.png”, considerada de tamanho intermediário nos testes, percebe-se que a quantidade maior de pixels no primeiro caso de concorrência, uma vez que a grande quantidade de GoRoutines se torna ineficiente
- Por outro lado, a implementação de concorrência com o número de threads limitado à quantidade de threads do processador demonstrou performar melhor significativamente

Resultados Da Análise de Desempenho (Apple)



- Com uma imagem maior, fica ainda mais claro o que se foi observado em “Apple.png”
- Pode-se concluir que concorrência pode melhorar a performance da aplicação, mas uma implementação pouco eficiente pode ser pior do que não executar os processos em paralelo

Resultados Da Análise de Desempenho (Painting)



# Obrigado!