UNIVERSITY OF SOUSSE

جامعة سوسة

Socially Responsible University

University of Sousse

National Engineering School of Sousse

المدرسة الوطنية للمهندسين بسوسة

École Nationale d'Ingénieurs de Sousse

Comparison of Performance Between CPU and GPU on Point-to-Point Image Processing Operations

Conducted by:

MHAMED Mohamed

Field:

Industrial Electronics

I.Abstract

This report explores the comparison of performance between CPU and GPU in point-to-point image processing operations. Image processing, a specialized area within Digital Signal Processing, requires substantial computational power due to the nature of its complex algorithms, such as convolution and matrix operations. Traditional CPU-based sequential processing often fails to meet the high computational demands of modern image processing tasks. To address this, GPUs, with their parallel computing capabilities, have emerged as a more efficient solution. Using CUDA, a platform for parallel processing created by NVIDIA, this report evaluates various image processing algorithms executed on both CPU and GPU. The results indicate that the GPU significantly outperforms the CPU in most cases, particularly in tasks involving high-resolution images.

II.Introduction

Image processing is a subfield of Digital Signal Processing (DSP) that involves manipulating images to enhance or extract useful information. With the increasing demand for high-quality images in applications such as medical imaging, computer vision, and satellite imagery, traditional CPUs have struggled to keep up due to their inherent serial processing limitations.

Parallel processing, facilitated by GPUs, has proven to be an efficient alternative. Unlike CPUs, which process data sequentially, GPUs can perform multiple operations simultaneously due to their numerous cores. This makes them ideal for tasks such as image processing, where multiple pixels can be processed concurrently. CUDA, developed by NVIDIA, allows developers to harness the power of the GPU for computationally intensive tasks.

This report focuses on point-to-point image processing, which transforms each pixel individually without considering neighboring pixels. Operations such as brightening, darkening, and converting RGB images to grayscale are executed on both CPU and GPU to compare their performance.

III.Important Definitions

- **Digital Image Processing**: Refers to the manipulation of digital images through computational algorithms.

- **Parallel Processing**: The simultaneous execution of multiple computations, reducing the time required for tasks involving large datasets, such as image processing.

- **CUDA (Compute Unified Device Architecture)**: A parallel computing platform developed by NVIDIA, enabling developers to utilize the GPU's processing power.

- **OpenCV**: An open-source library of computer vision and machine learning algorithms, widely used for image processing tasks.

- **GPU (Graphics Processing Unit)**: A hardware component designed for rapid image rendering and parallel computations, making it ideal for tasks requiring intensive mathematical operations.

- **Grayscale**: A type of image where each pixel represents a shade of gray, often used in image processing to simplify the data being analyzed.

- **Thresholding**: A method of segmenting images by converting grayscale images to binary images based on a threshold value.

IV.Experiment and Results

ship_4k_rgba.png   ship_4k_rgba_gra y_cpu.png   ship_4k_rgba_gra y_gpu.png

The experiment involves converting an image to grayscale using both CPU and GPU processing techniques. The primary goal is to compare the performance of the two approaches. Below are the steps and results obtained from the experiment.
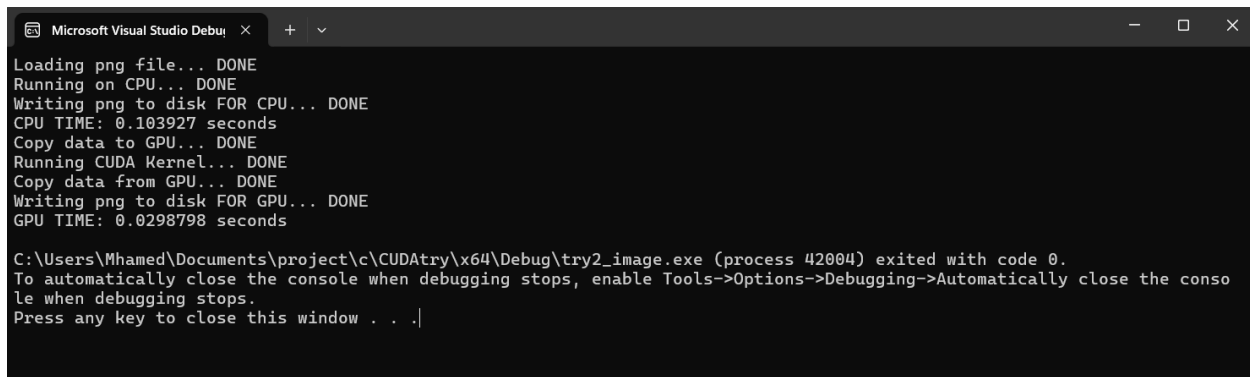
Experimental Setup

1. **Input Image**: The input image is a 4K resolution PNG file with RGBA channels, where each pixel has four components: red (r), green (g), blue (b), and alpha (a).

2. **CPU-Based Image Grayscale Conversion**: The CPU-based grayscale conversion iterates over each pixel, calculating the grayscale value using the following formula:

$$pixelValue = 0.2126 \times R + 0.7152 \times G + 0.0722 \times B$$

This pixel value is then applied uniformly to the red, green, and blue channels to produce a grayscale image.

3. **GPU-Based Image Grayscale Conversion**: A CUDA kernel performs a similar grayscale conversion, where each thread processes a single pixel in parallel. The threads are arranged in a grid of blocks, each block containing 32x32 threads, processing a chunk of the image at a time.

Results

- **CPU Time**: 0.103927 seconds

- **GPU Time**: 0.0298798 seconds

The GPU-based approach was approximately **3.48 times faster** than the CPU-based method in this experiment, showcasing the advantage of parallel processing for tasks like image manipulation.

V.Comparison

In terms of performance, GPU-based image processing significantly outperforms CPU-based methods due to its parallel architecture. For large image sizes, like the 4K image used in this experiment, the difference in execution time becomes highly noticeable.

- **CPU Processing**: While straightforward to implement, CPU image processing scales poorly with image size due to its sequential nature. Each pixel is processed one by one, which makes it time-consuming for high-resolution images.

- **GPU Processing**: CUDA kernels allow each pixel to be processed in parallel, greatly improving the execution time, especially for large datasets like 4K images. By leveraging thousands of GPU cores, the task can be done much faster than on a typical CPU with fewer cores.

VI.Conclusion

This experiment highlights the superior performance of GPUs over CPUs in point-to-point image processing operations. By leveraging parallel processing, the GPU can handle complex tasks more efficiently, particularly when dealing with large images. The results suggest that for real-time or large-scale image processing tasks, GPUs are the preferred choice due to their significant speed advantage. This experiment also demonstrates the utility of CUDA in maximizing the processing potential of GPUs, making it an essential tool for modern image processing applications.

Future work may involve extending the experiment to more complex image processing tasks such as convolution and filtering, which may reveal even greater performance disparities between CPUs and GPUs.