Canny Edge Detection Performance Comparison Between Parallel Programming and Serial Programming

NAME: 陳哲輝、黃羿翔、劉宗翰

Team 33

DATE: 2021/1/7

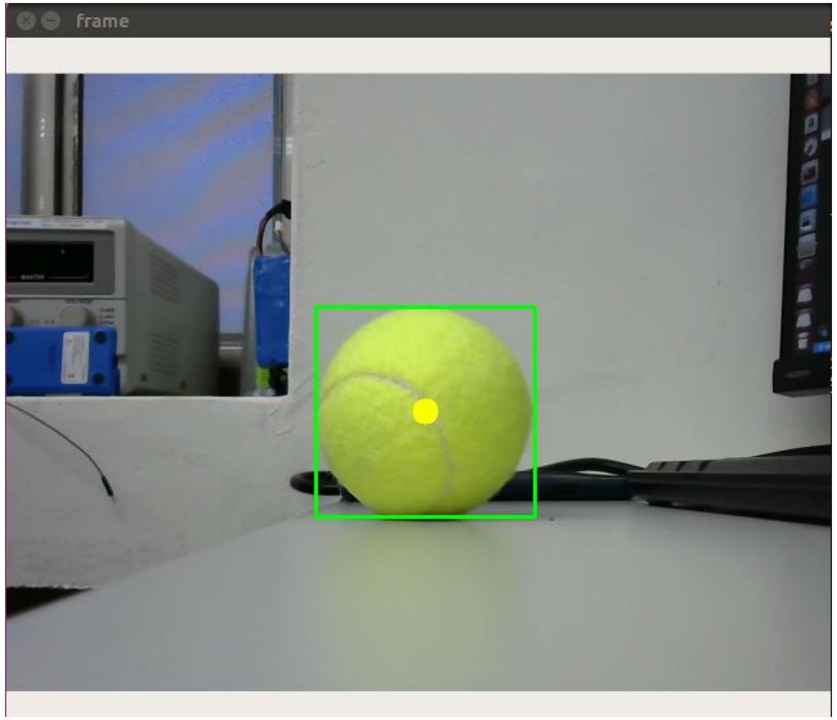
Parallel Programming, Fall 2020

OUTLINES

- Motivation
- Introduction
- Platform
- Pthreads
- OpenMP
- CUDA
- OpenCL
- Result & Conclusion

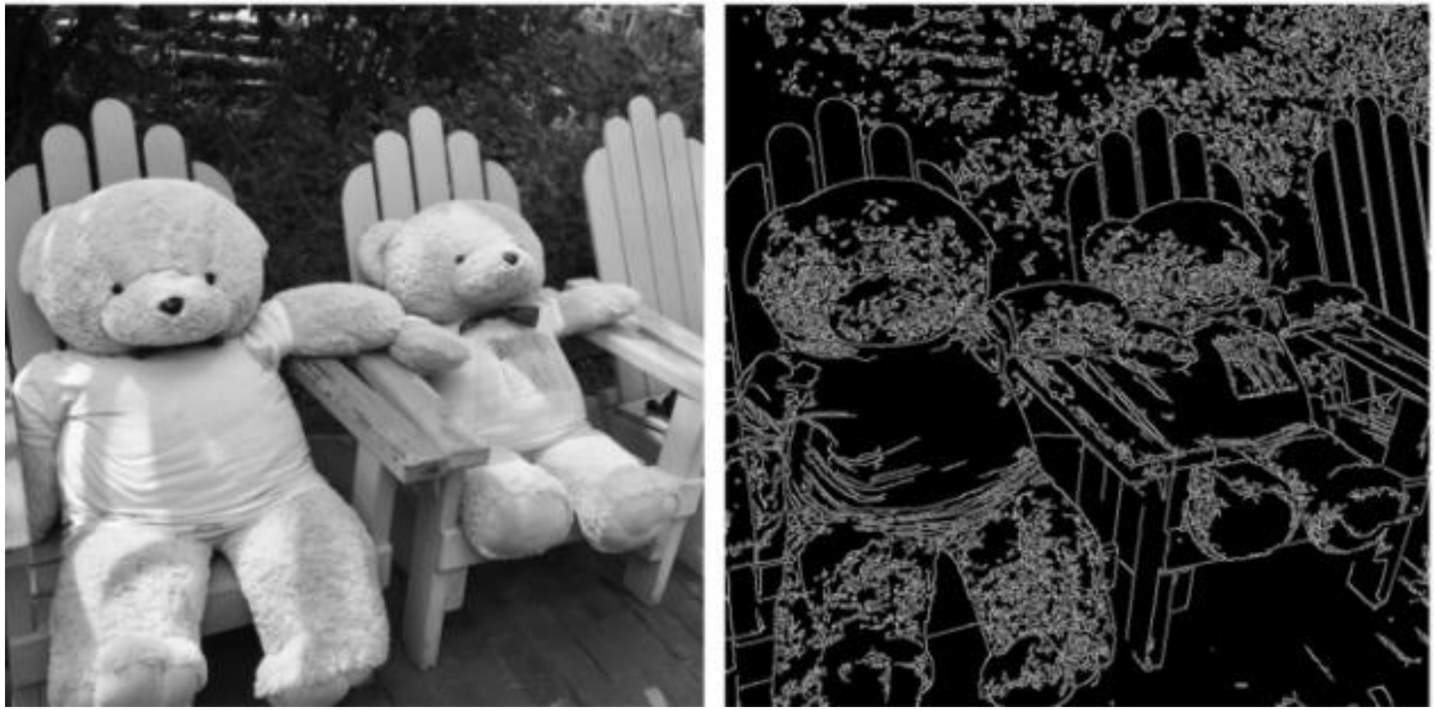
Motivation

以前有個專題需要尋找球體的邊緣，當初由於算力不足沒有辦法在高畫質的狀態下足夠快速地尋找球的邊緣(20hz 以上)



Introduction

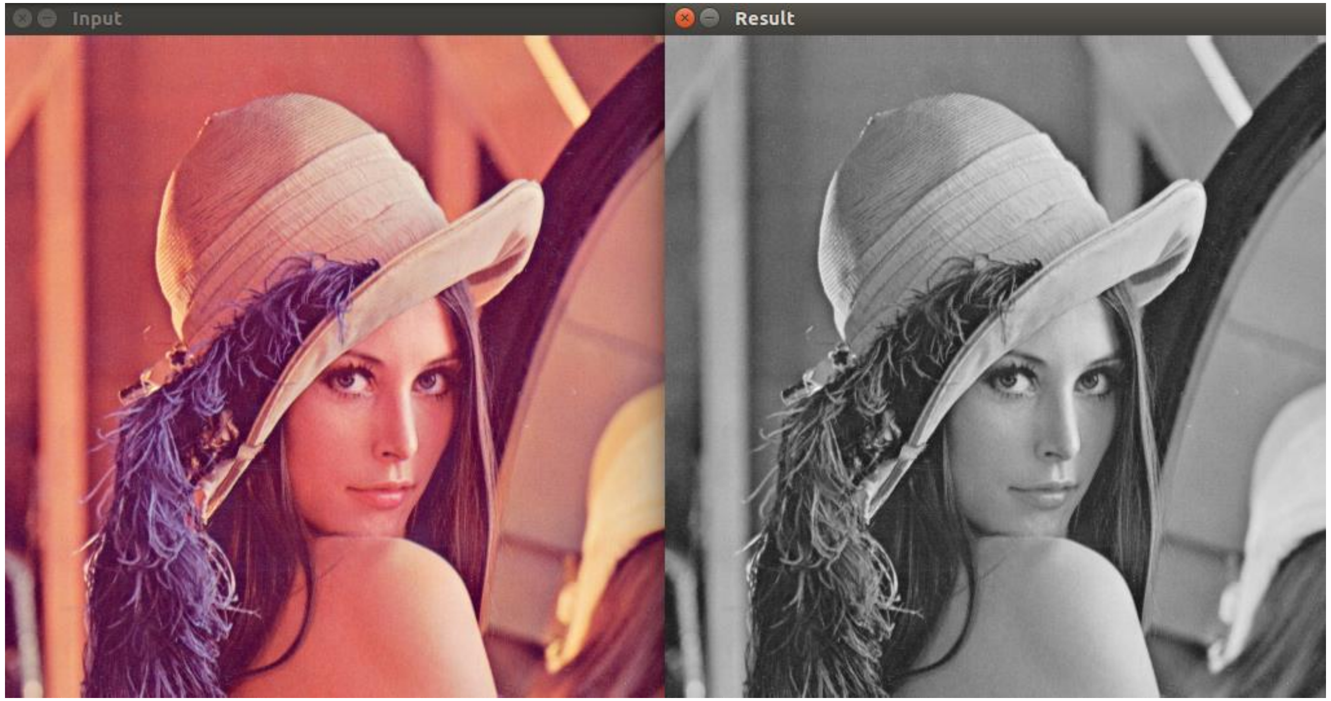
對於許多影像處理來說，尋找邊緣都是一個相當重要的步驟



Introduction

Canny 演算法算是一個不錯的邊緣探測演算法，主要分成以下幾個步驟

1. Noise reduction and convert image into grayscale



Introduction

2.Use Sobel filter to find the intensity gradient of the image

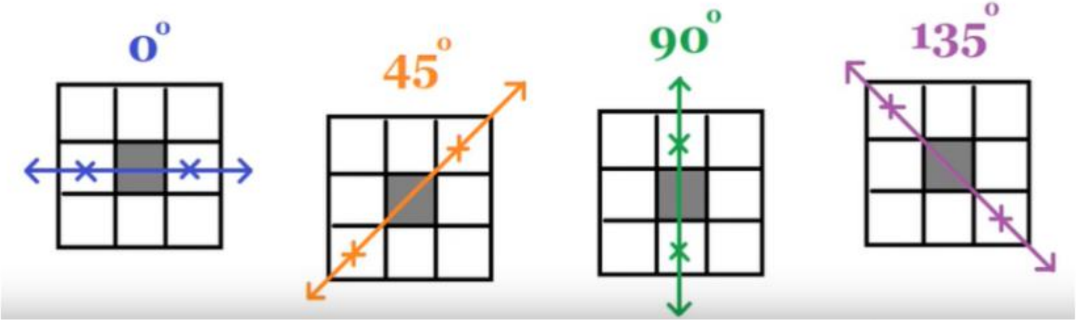
$$G = \sqrt{G_x^2 + G_y^2}$$

$$\theta = arc\tan\left(G_y / G_x\right)$$

Introduction

3.Non-maximum suppression

3-1. Find gradient direction



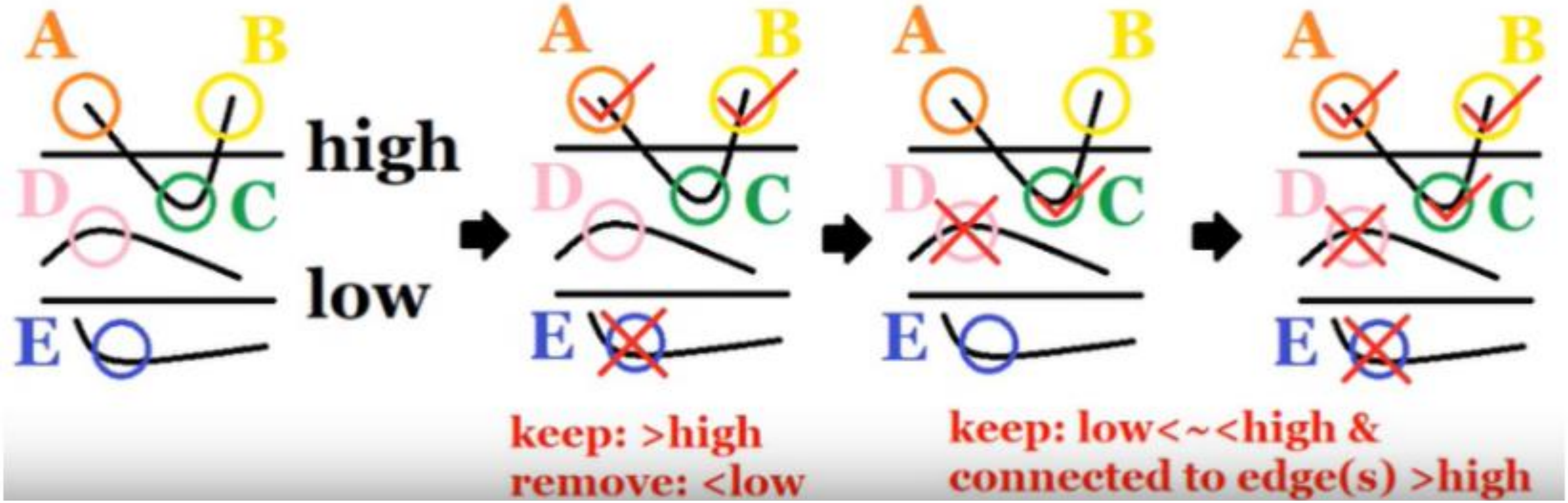
3-2. Non-maximum suppression

$$\begin{bmatrix} 0.5 & 0.9 & 1 \\ 0 & 0.3 & 0.7 \\ 0.9 & 0 & 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 0.5 & 0.9 & 1 \\ 0 & 0 & 0.7 \\ 0 & 0 & 0 \end{bmatrix}$$

Introduction

4. Double threshold

5. Edge Tracking by Hysteresis



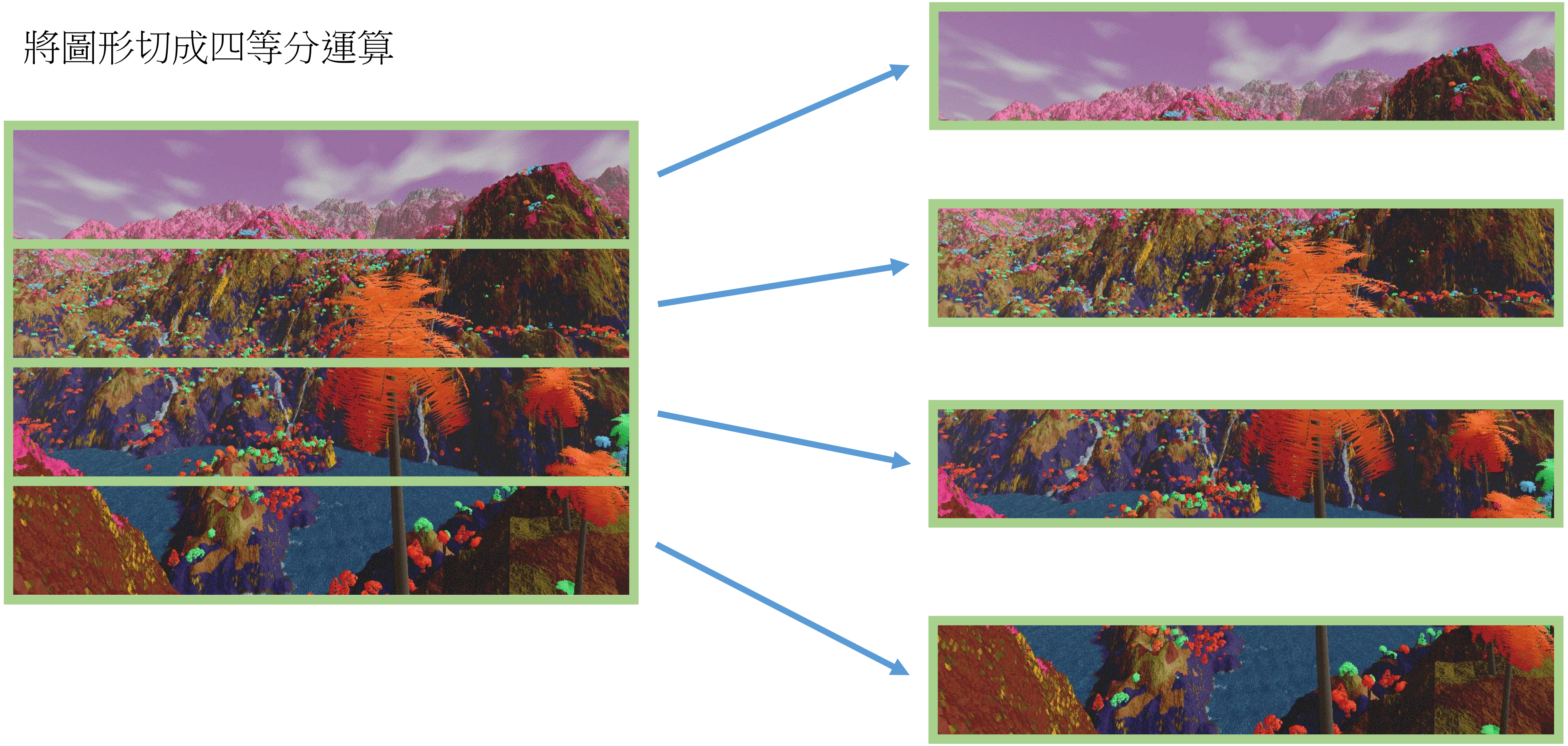
Platform

- Ubuntu 18.04
- CPU: i5-7500
- Memory 16G
- GPU: GTX 1060 6GB

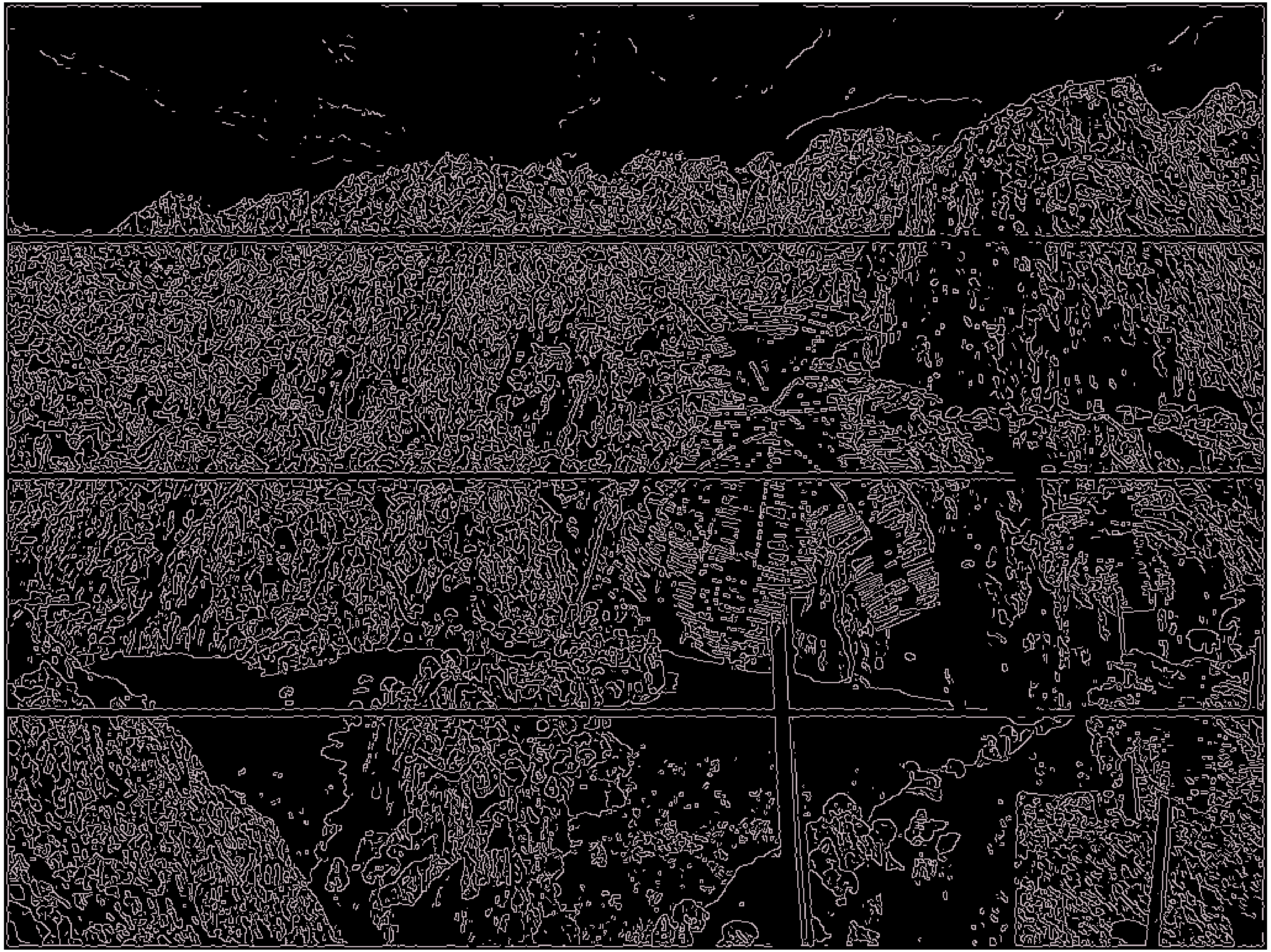
Pthreads

將圖形切成四等分運算

邊緣計算不如預期!

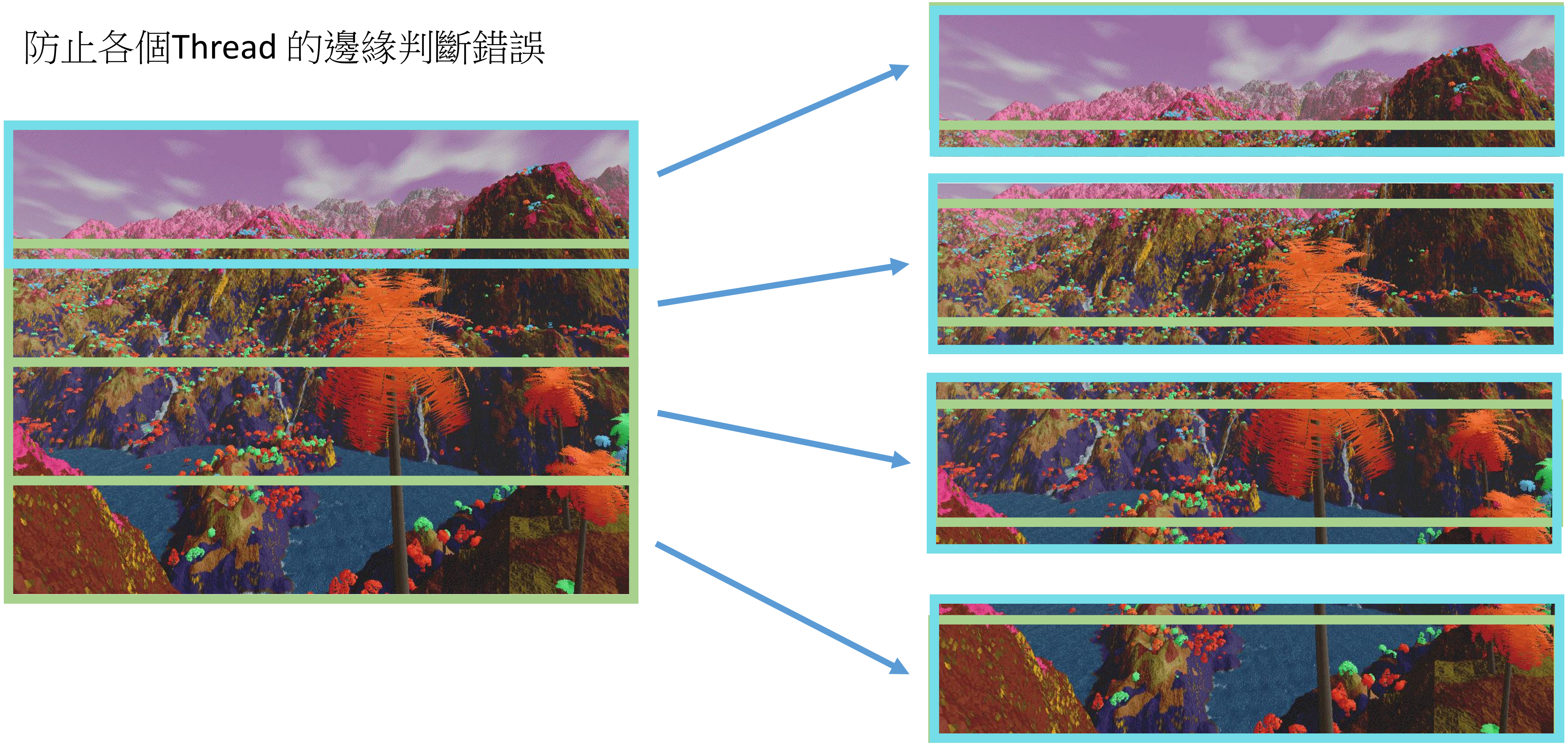


Pthreads



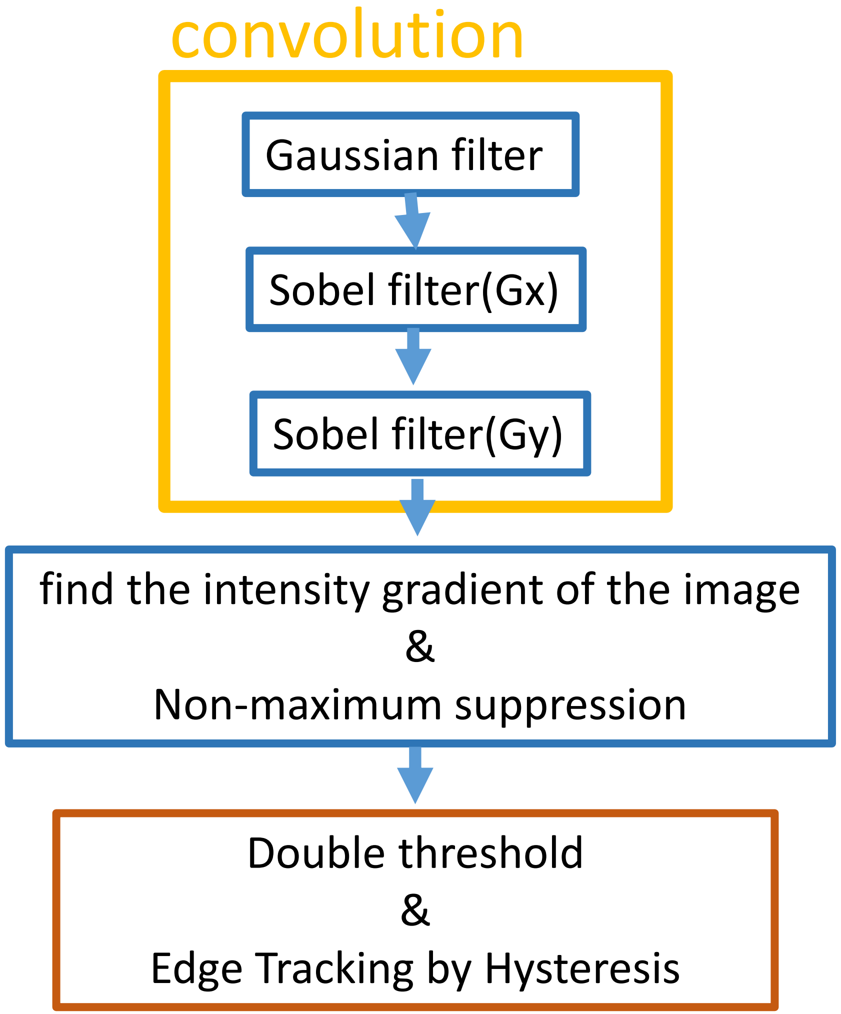
Pthreads

防止各個Thread 的邊緣判斷錯誤



Pthreads

OpenMP



convolution

Gaussian filter

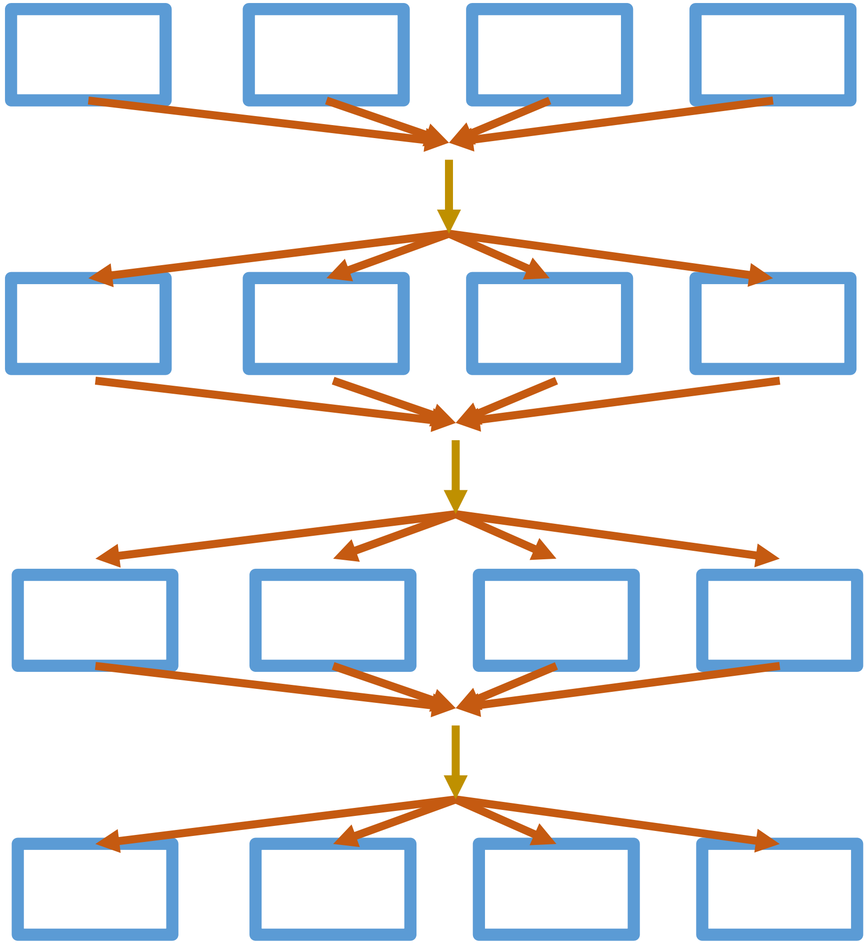
Sobel filter(Gx)

Sobel filter(Gy)

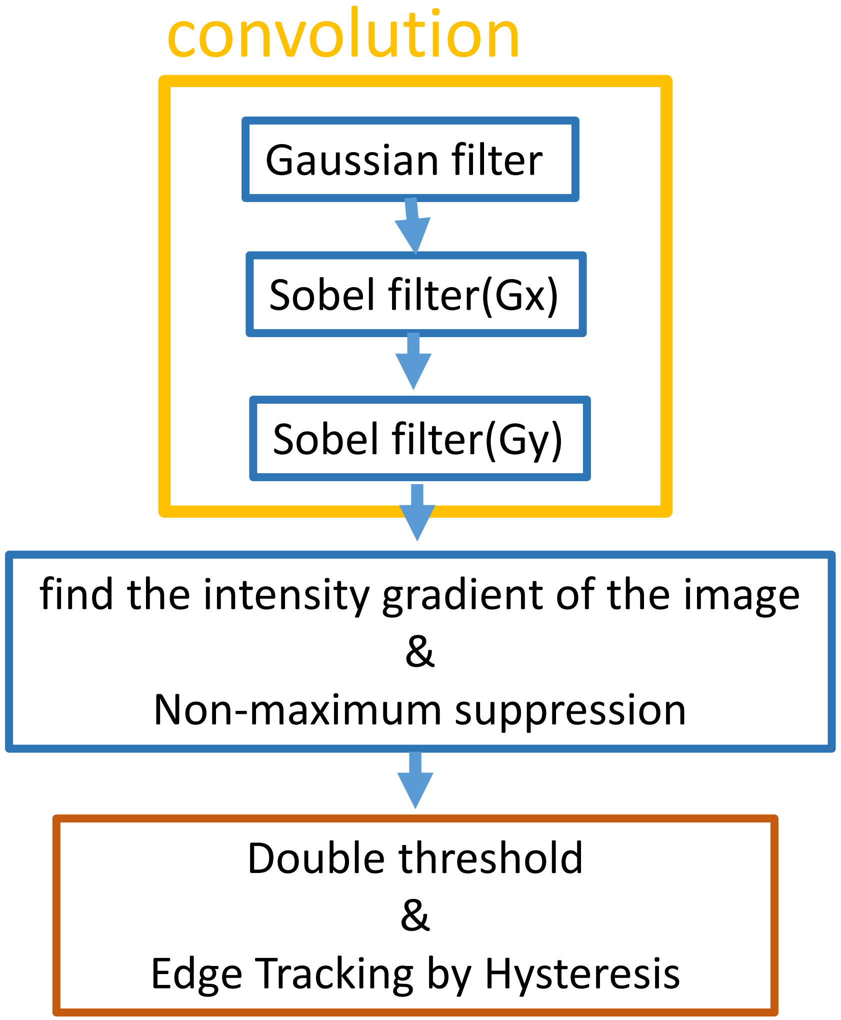
find the intensity gradient of the image
&
Non-maximum suppression

Double threshold
&
Edge Tracking by Hysteresis

示意圖



CUDA & OpenCL



convolution

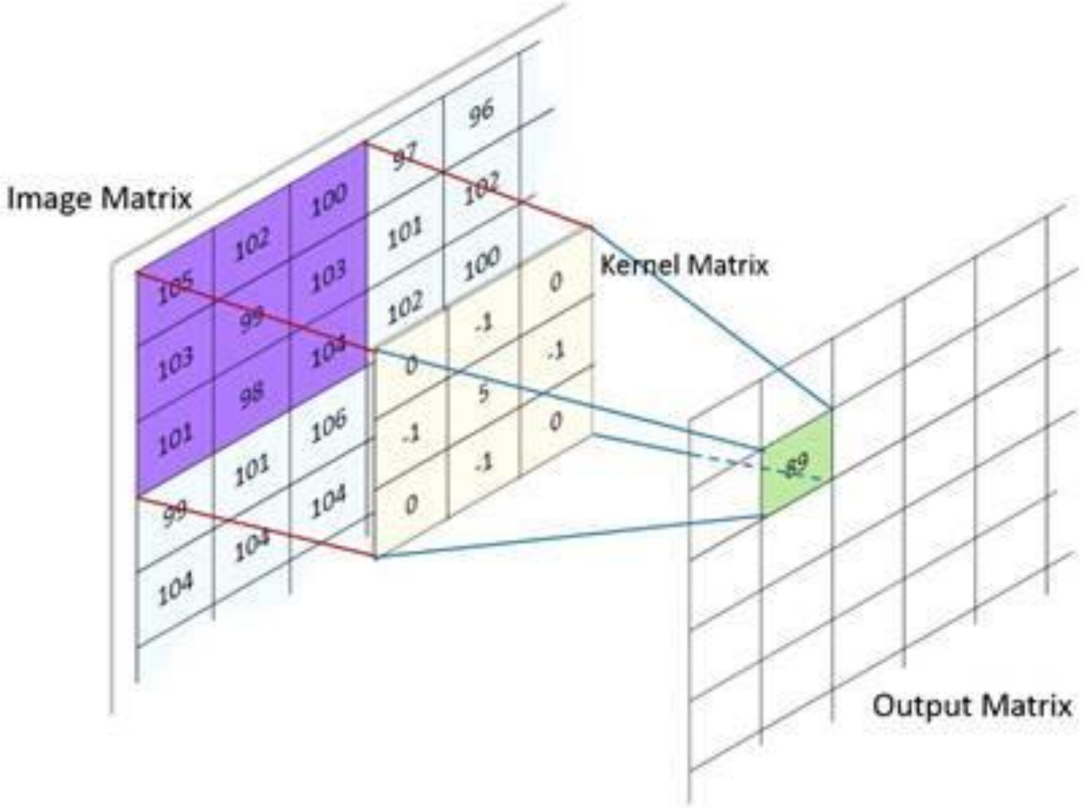
Gaussian filter

Sobel filter(Gx)

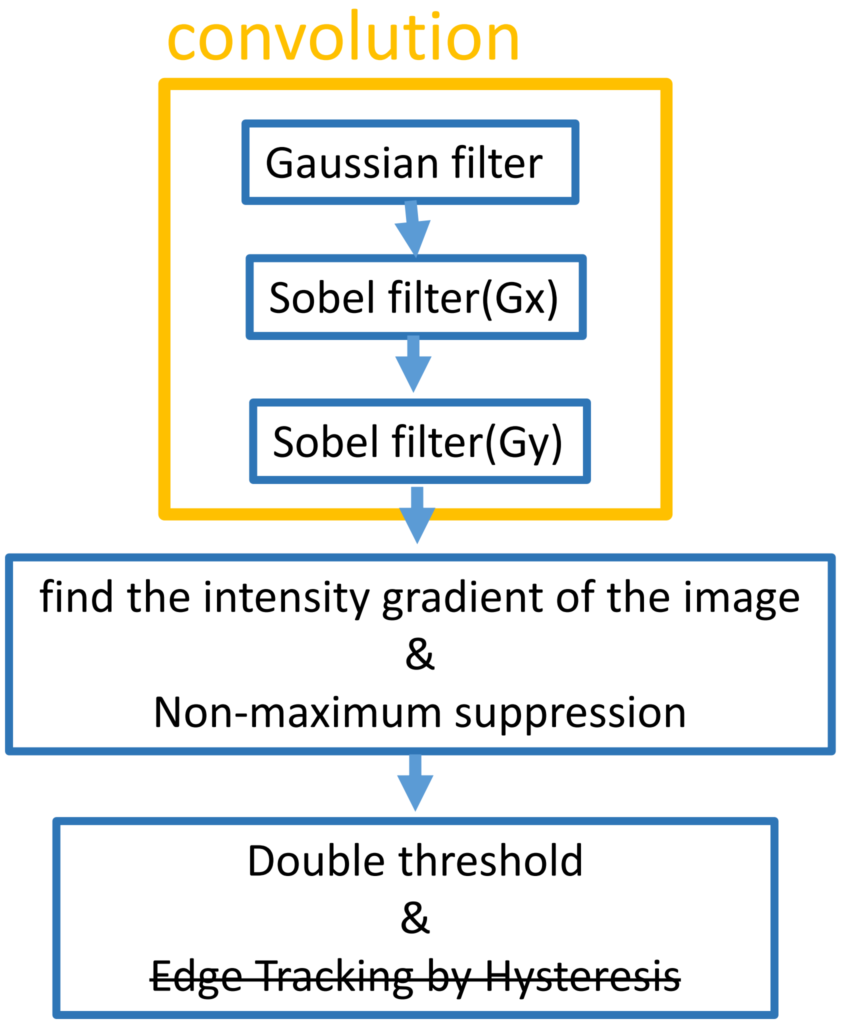
Sobel filter(Gy)

find the intensity gradient of the image
&
Non-maximum suppression

Double threshold
&
Edge Tracking by Hysteresis



CUDA & OpenCL



convolution

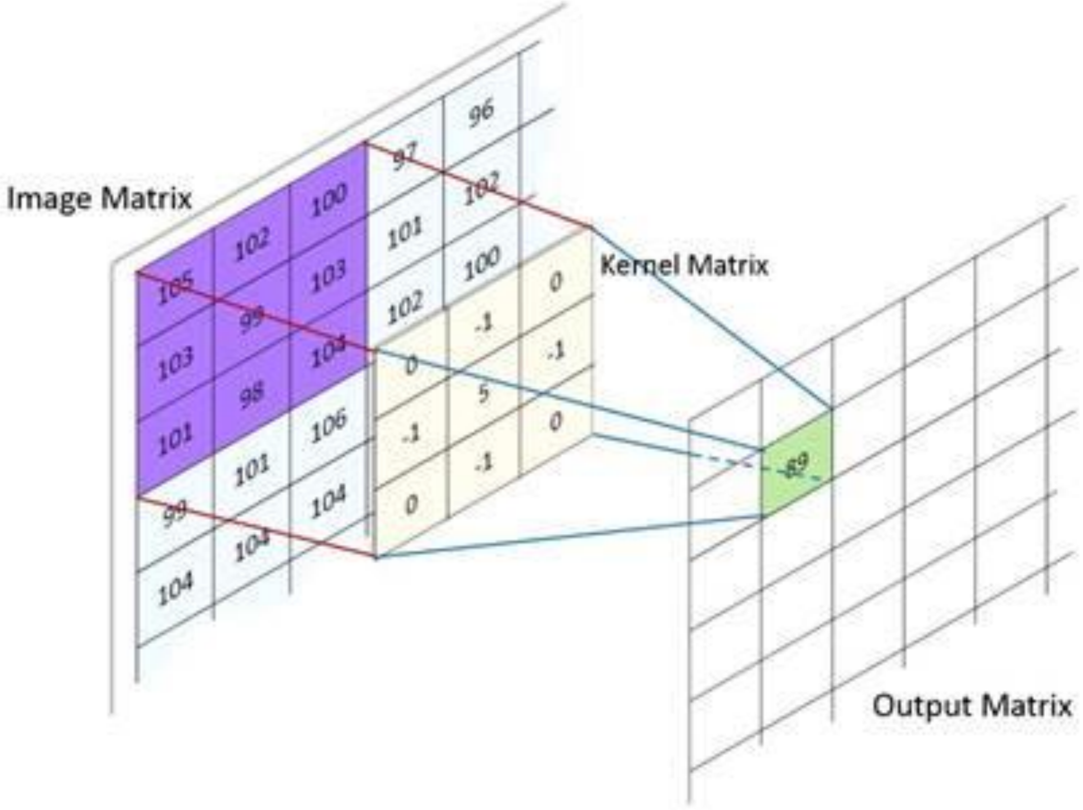
Gaussian filter

Sobel filter(Gx)

Sobel filter(Gy)

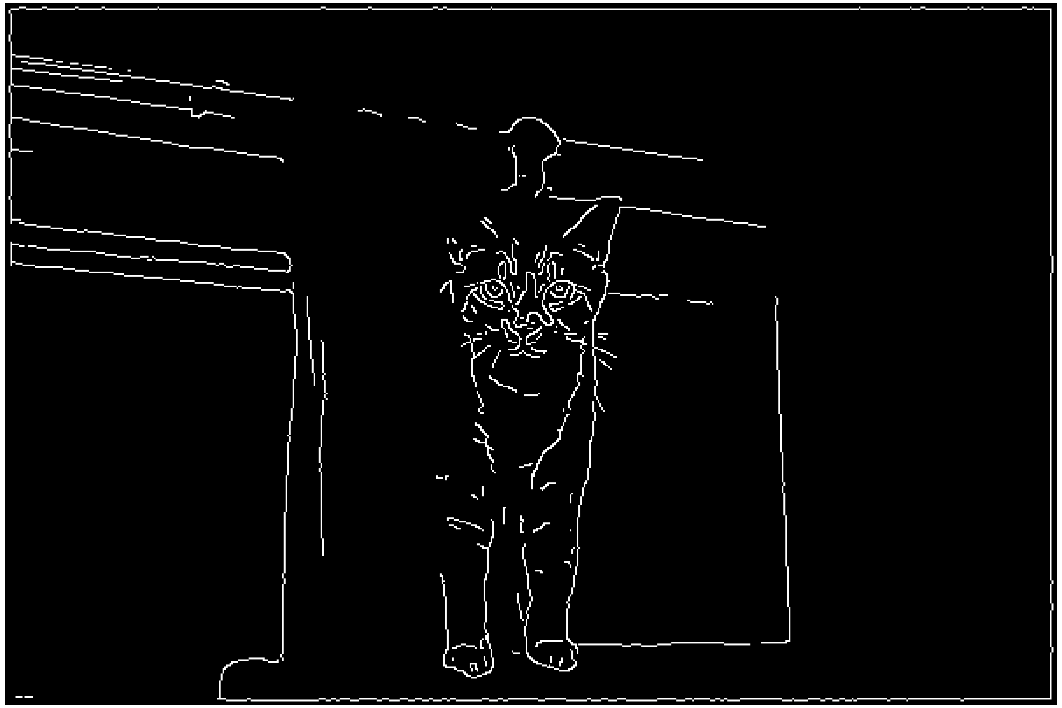
find the intensity gradient of the image
&
Non-maximum suppression

Double threshold
&
~~Edge Tracking by Hysteresis~~

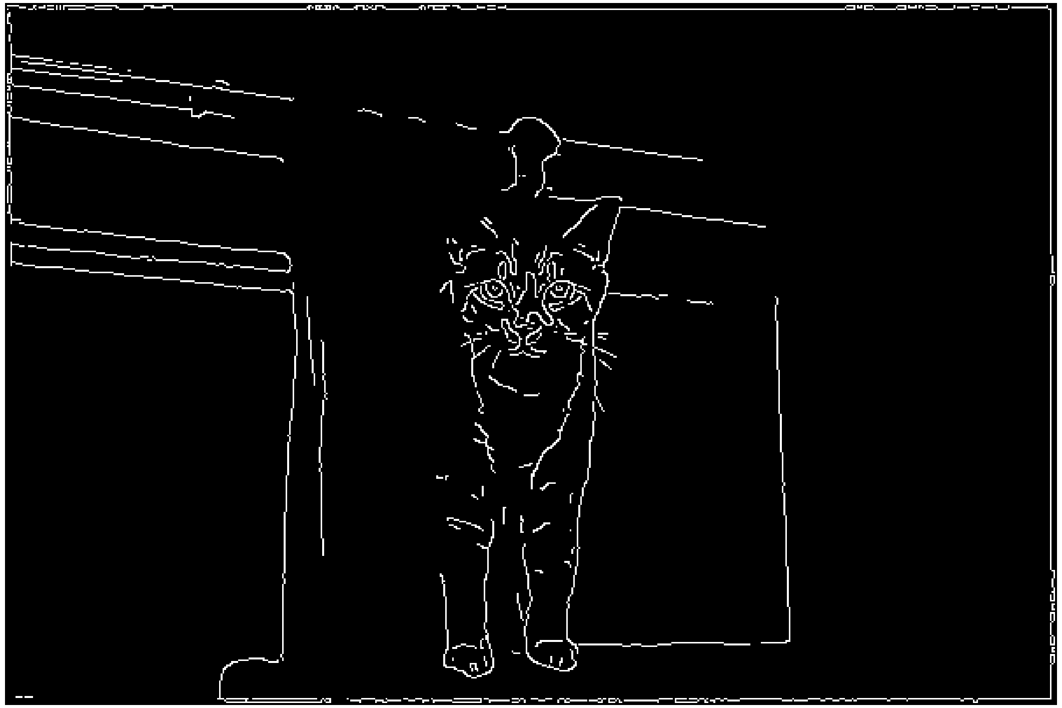


Result & Conclusion

simplified Edge Tracking by Hysteresis

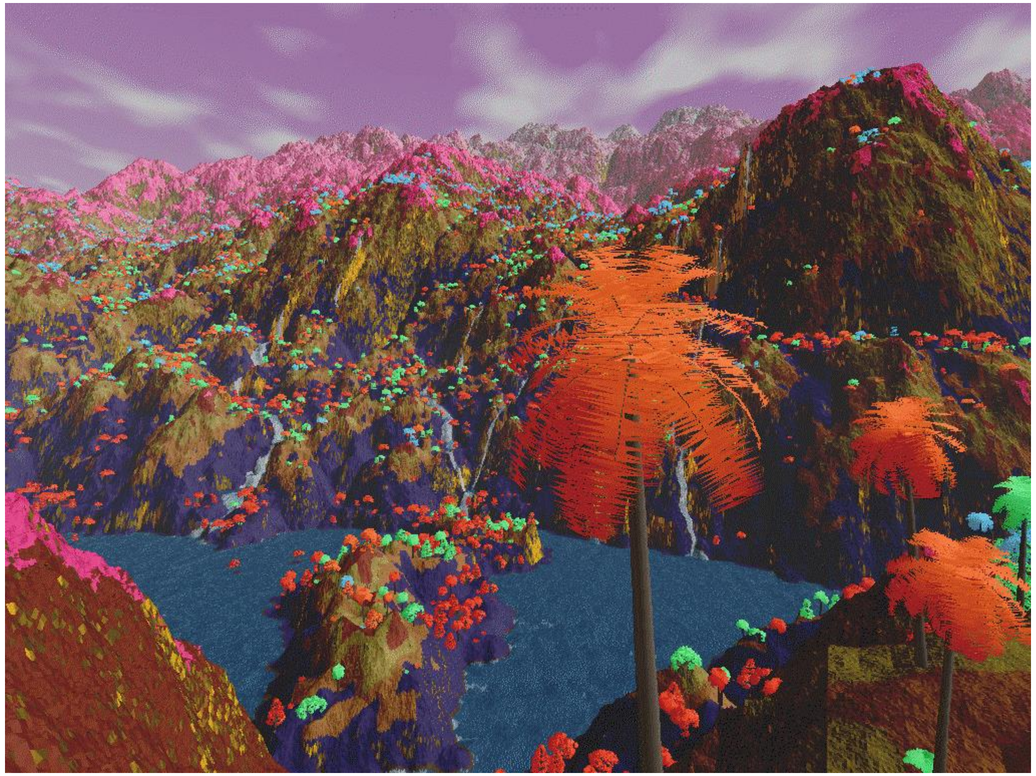


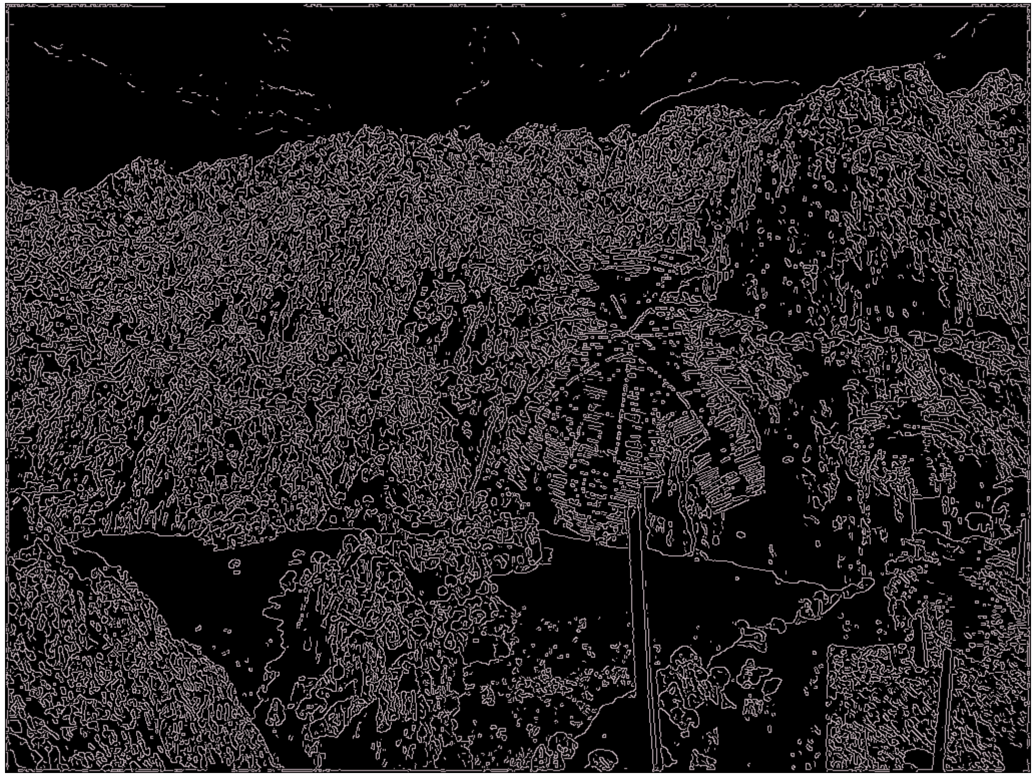
Edge Tracking by Hysteresis



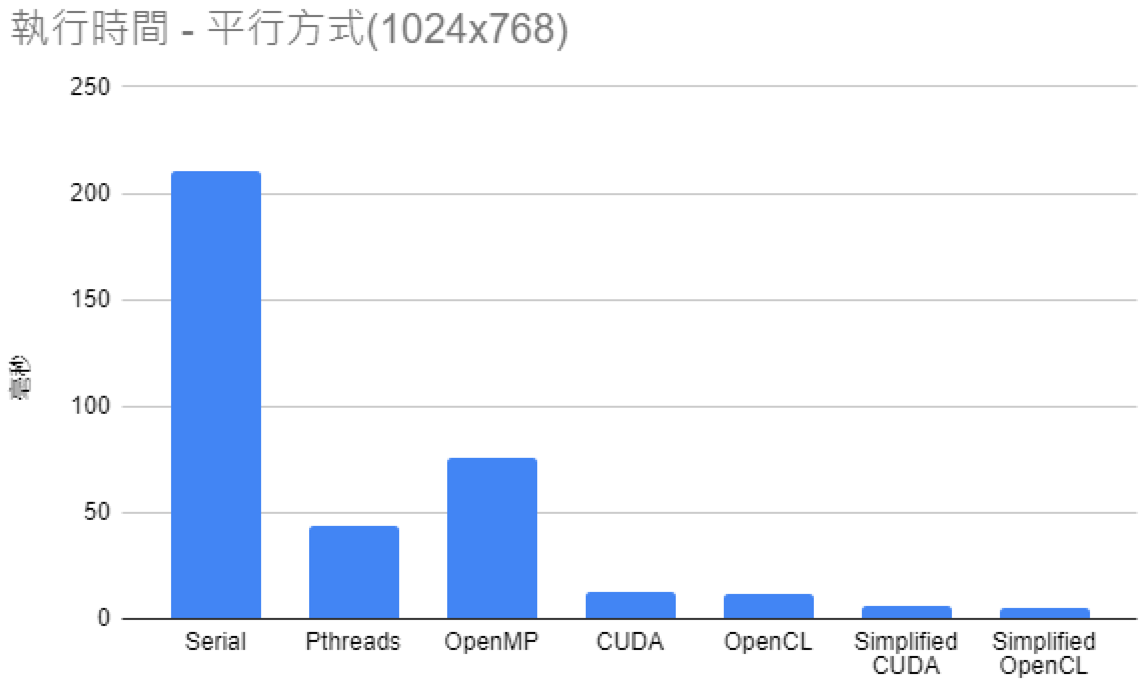
Result & Conclusion

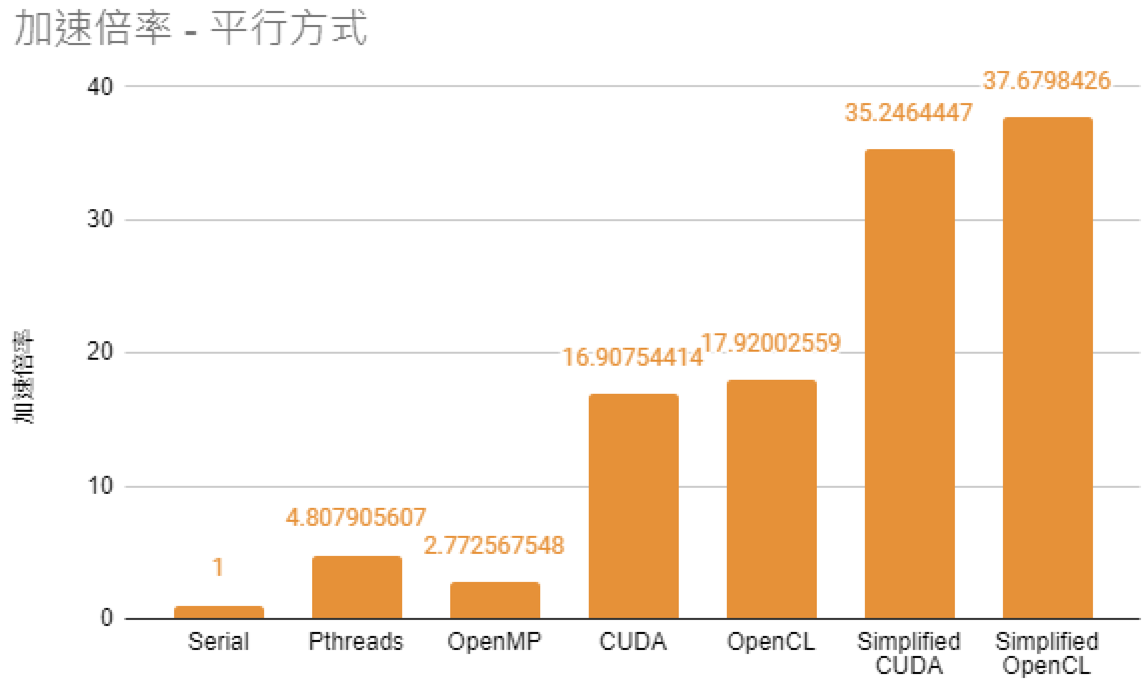
Pixel: 1024*768





Result & Conclusion



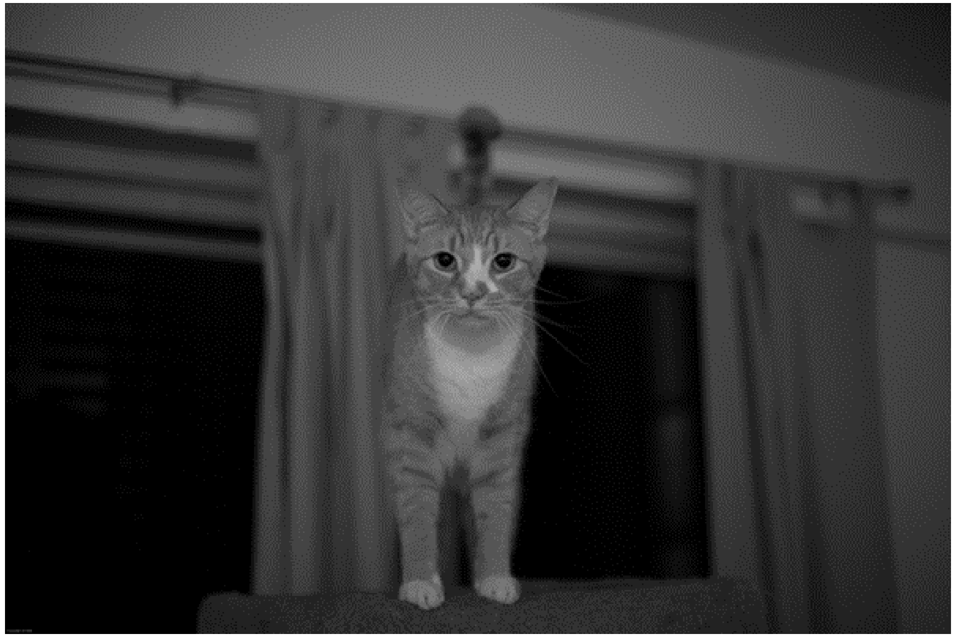


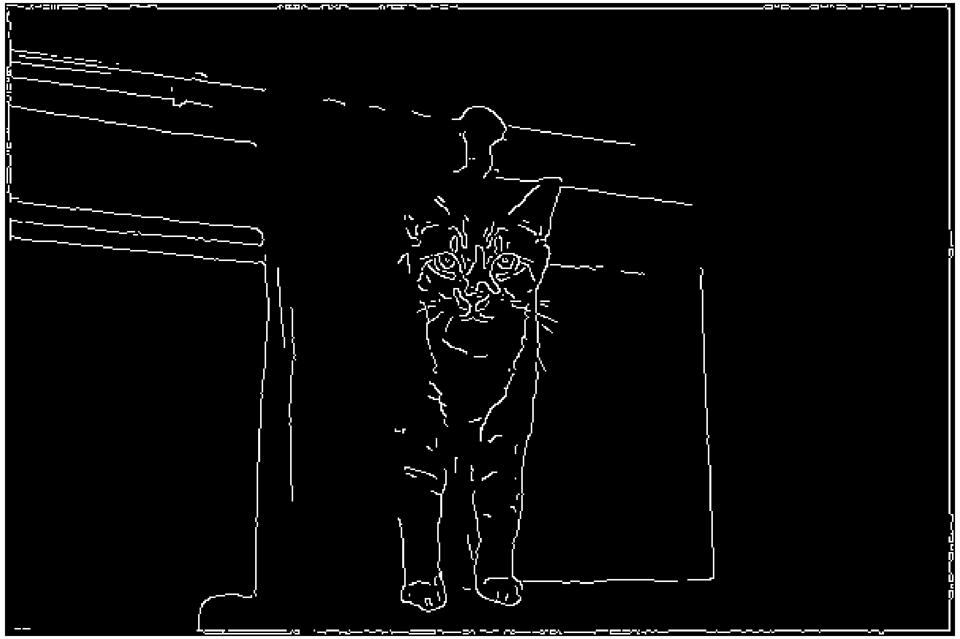
4.76 Hz

178.86 Hz

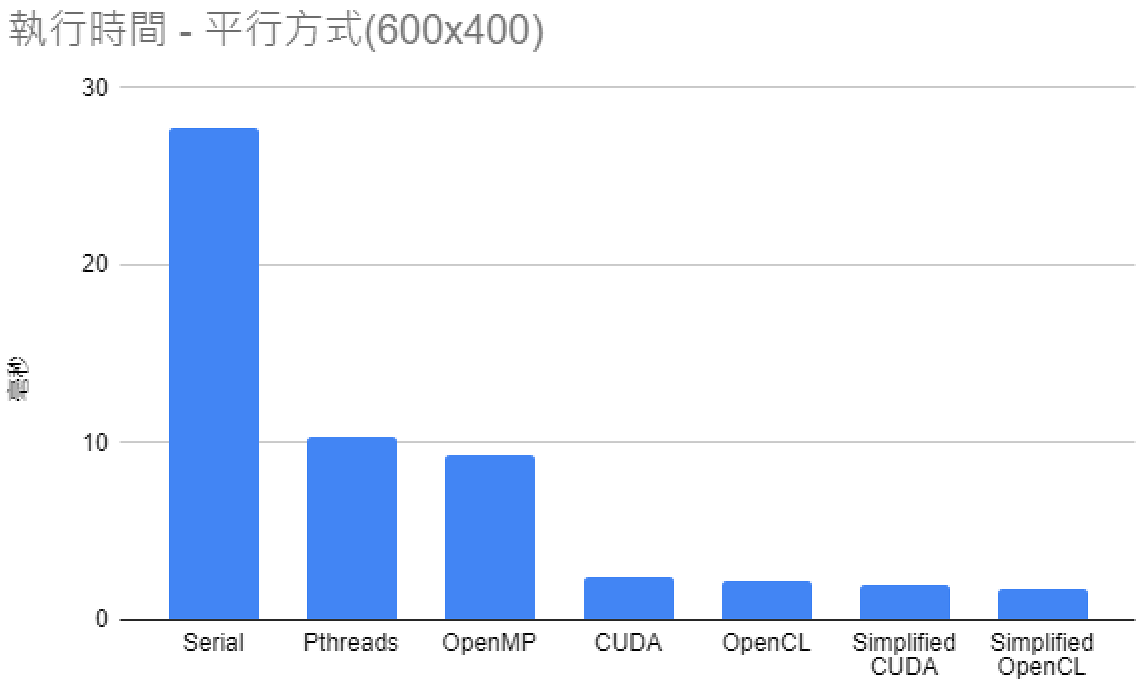
Result & Conclusion

Pixel: 600X400

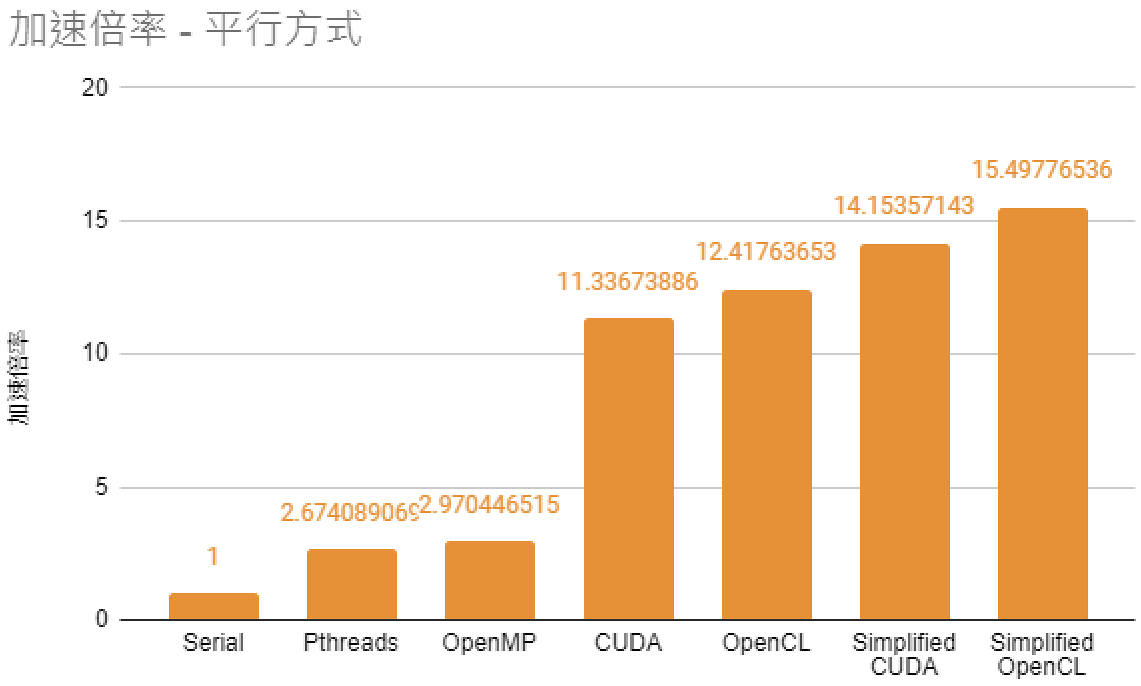




Result & Conclusion



執行時間 - 平行方式(600x400)



加速倍率 - 平行方式