

# Homework9 OpenMP卷积加速测试

2101210578 胡成成

## Problem

**Sobel** 常用于图像的边缘检测，计算公式如下：

$$Gx = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \quad Gy = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A$$

B站

$$\mathbf{G} = \sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2}$$

其中 **A** 是二维图像，**G** 为检测到的梯度强度。

请用**OpenMP** 加速方法计算边缘检测（边缘强度），评估采用**OpenMP** 带来的加速比。

自选图像，要求图像像素数大于**2, 000, 000**。

## Answer

### 源代码

- C++ 代码

```
#include <opencv2/opencv.hpp>
#include <iostream>
#include <math.h>
#include <time.h>
#include "omp.h"

using namespace std;
using namespace cv;

int main() {
    Mat src = imread("t3.jpg");
    Mat src_gray;
    cvtColor(src, src_gray, COLOR_BGR2GRAY);

    Mat dst;
    int dst_gx, dst_gy;
    dst = src_gray.clone();

    int gx[9] = { -1, 0, 1, -2, 0, 2, -1, 0, 1 };
    int gy[9] = { -1, -2, -1, 0, 0, 0, 1, 2, 1 };
```

```

int m, n;
m = src_gray.rows;
n = src_gray.cols;
printf("m, n: %d, %d", m, n);

// Test basic calculation
clock_t start1 = clock();
for (int i = 0; i < m; i++) {
    for (int j = 0; j < n; j++) {
        if (i > 1 && i < m - 1 && j > 1 && j < n - 1) {
            dst_gx = src_gray.at<uchar>(i - 1, j - 1) * gx[0] +
src_gray.at<uchar>(i - 1, j) * gx[1] + src_gray.at<uchar>(i - 1, j + 1) * gx[2]
+ \
                src_gray.at<uchar>(i, j - 1) * gx[3] + src_gray.at<uchar>(i,
j) * gx[4] + src_gray.at<uchar>(i, j + 1) * gx[5] + \
                src_gray.at<uchar>(i + 1, j - 1) * gx[6] +
src_gray.at<uchar>(i + 1, j) * gx[7] + src_gray.at<uchar>(i + 1, j + 1) * gx[8];
            dst_gy = src_gray.at<uchar>(i - 1, j - 1) * gy[0] +
src_gray.at<uchar>(i - 1, j) * gy[1] + src_gray.at<uchar>(i - 1, j + 1) * gy[2]
+ \
                src_gray.at<uchar>(i, j - 1) * gy[3] + src_gray.at<uchar>(i,
j) * gy[4] + src_gray.at<uchar>(i, j + 1) * gy[5] + \
                src_gray.at<uchar>(i + 1, j - 1) * gy[6] +
src_gray.at<uchar>(i + 1, j) * gy[7] + src_gray.at<uchar>(i + 1, j + 1) * gy[8];
            dst.at<uchar>(i, j) = round(sqrt(dst_gx * dst_gx + dst_gy *
dst_gy));
            if (dst.at<uchar>(i, j) > 255) {
                dst.at<uchar>(i, j) = 255;
            }
            if (dst.at<uchar>(i, j) < 0) {
                dst.at<uchar>(i, j) = 0;
            }
        }
        else {
            dst.at<uchar>(i, j) = src_gray.at<uchar>(i, j);
        }
    }
}
clock_t stop1 = clock();

//Using OpenMP
clock_t start2 = clock();
#pragma omp parallel for private(dst_gx, dst_gy)
for (int i = 0; i < m; i++) {
    for (int j = 0; j < n; j++) {
        if (i > 1 && i < m - 1 && j > 1 && j < n - 1) {
            dst_gx = src_gray.at<uchar>(i - 1, j - 1) * gx[0] +
src_gray.at<uchar>(i - 1, j) * gx[1] + src_gray.at<uchar>(i - 1, j + 1) * gx[2]
+ \
                src_gray.at<uchar>(i, j - 1) * gx[3] + src_gray.at<uchar>(i,
j) * gx[4] + src_gray.at<uchar>(i, j + 1) * gx[5] + \
                src_gray.at<uchar>(i + 1, j - 1) * gx[6] +
src_gray.at<uchar>(i + 1, j) * gx[7] + src_gray.at<uchar>(i + 1, j + 1) * gx[8];
            dst_gy = src_gray.at<uchar>(i - 1, j - 1) * gy[0] +
src_gray.at<uchar>(i - 1, j) * gy[1] + src_gray.at<uchar>(i - 1, j + 1) * gy[2]
+ \

```

```

        src_gray.at<uchar>(i, j - 1) * gy[3] + src_gray.at<uchar>(i,
j) * gy[4] + src_gray.at<uchar>(i, j + 1) * gy[5] + \
src_gray.at<uchar>(i + 1, j - 1) * gy[6] +
src_gray.at<uchar>(i + 1, j) * gy[7] + src_gray.at<uchar>(i + 1, j + 1) * gy[8];
dst.at<uchar>(i, j) = round(sqrt(dst_gx * dst_gx + dst_ty *
dst_ty));
if (dst.at<uchar>(i, j) > 255) {
    dst.at<uchar>(i, j) = 255;
}
if (dst.at<uchar>(i, j) < 0) {
    dst.at<uchar>(i, j) = 0;
}
else {
    dst.at<uchar>(i, j) = src_gray.at<uchar>(i, j);
}
}

clock_t stop2 = clock();

double div = double(stop1 - start1) / double(stop2 - start2);

printf("Basic calc: %d\n", stop1 - start1);
printf("Using OpenMP: %d\n", stop2 - start2);
printf("Div: %f\n", div);

imshow("dst", dst);

//imwrite("./test_dst.jpg", dst);
waitKey(0);

return 0;
}

```

- 测试原始图像



- 运行结果：没有开并行计算需要4864ms，开了只需要615ms，加速比：7.908943

```
Microsoft Visual Studio 调试控制台
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load C:\Windows\SYSTEM32\opencv_core_parallel_tbb454_64d.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load opencv_core_parallel_tbb454_64d.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load C:\Windows\SYSTEM32\opencv_core_parallel_openmp454_64d.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load opencv_core_parallel_openmp454_64d.dll => FAILED
m, n: 4928, 3264
Basic Calc: 4864
Using OpenMP: 615
Div: 7.908943
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\highgui\src\registry.impl.hpp (114) cv::highgui
backend::UIBackendRegistry::UIBackendRegistry UI: Enabled backends(4, sorted by priority): GTK(1000); GTK3(990); GTK2(9
80); WIN32(970) + BUILTIN(WIN32UI)
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load C:\Windows\SYSTEM32\opencv_highgui_gtk454_64.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load opencv_highgui_gtk454_64.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load C:\Windows\SYSTEM32\opencv_highgui_gtk3454_64.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load opencv_highgui_gtk3454_64.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load C:\Windows\SYSTEM32\opencv_highgui_gtk2454_64.dll => FAILED
[ INFO:0] global c:\build\master_wipack-build-win64-vc15\opencv\modules\core\src\utils\plugin_loader.impl.hpp (67) cv:::
plugin::impl::DynamicLib::libraryLoad load opencv_highgui_gtk2454_64.dll => FAILED
[ INFO:0] global C:\build\master_wipack-build-win64-vc15\opencv\modules\highgui\src\backend.cpp (90) cv::highgui_backen
d::createUIBackend UI: using backend: WIN32 (priority=970)
[ INFO:0] global C:\build\master_wipack-build-win64-vc15\opencv\modules\highgui\src>window_w32.cpp (3013) cv::impl::Win
```

- Sobel算子计算后的图像



## 遇到问题

- 加速比几乎为1的时候需要在visual studio中属性里打开支持OpenMP

