

week6实验记录

zxp

April 6, 2024

1 environment

实验1 cpu:Inter i5-12400f (2.5 GHz)‘

gpu:rtx4060ti

System:Ubuntu 22.04.1

Compiler:gcc 12.3

实验2 cpu:Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz (使用时申请了16个核心)

System:CentOS7

Compiler:9.5 和icpc

2 code

在GPU上实现了im2win

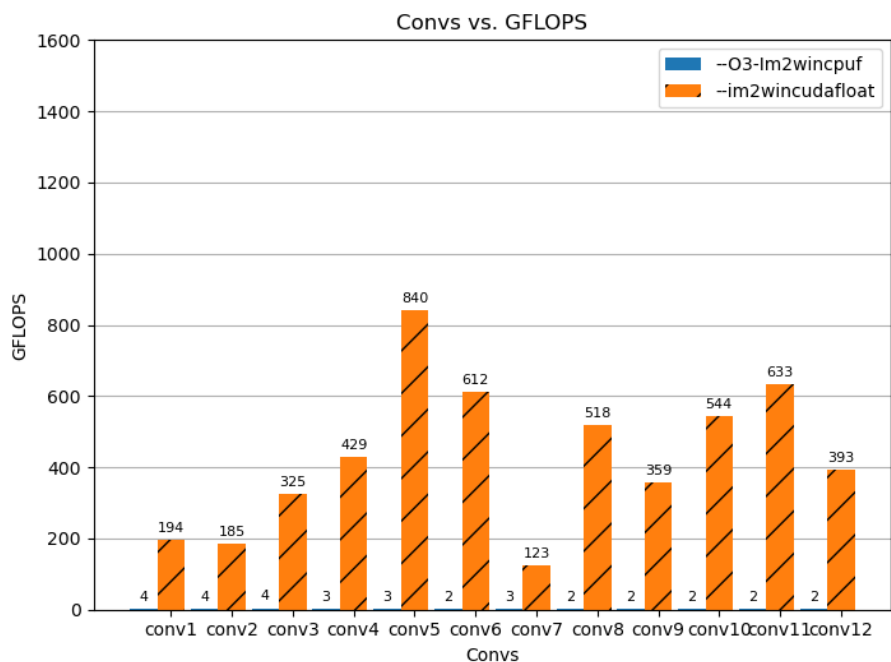


Figure 1: im2win

3 Experiment

在GPU上实现了im2win卷积，实现方式是一个核函数算一个output张量的元素，然后开启能一次计算完一个批次的那么多的线程，算一次得到一个output张量。

CUDA用起来十分麻烦，给人感觉就是一点反馈也没有，就只知道结果对不对，不好调试不知道问题出现在哪里。核函数可以用c语言的打印函数在控制台输出，但因为是多线程输出十分混乱（一开始的时候我还犯了很蠢的问题，用输出int的方式输出size_t，导致输出十分奇怪我一度以为GPU上的打印函数不对，后面把核函数的size_t全改int了，因为CUDA中获取线程id的变量就是int形的）。后面在每次输出是把对应的线程id也一起输出看起来才没这么混乱。前半周写的代码一直算不对，我一直以为是下标计算得有问题，后面通过把所有参与计算的元素输出才发现折磨了我很久的问题是数据搬运的时候数组长度弄错了导致GPU上的数据不全，GPU数组越界竟然没有报错。

3.1 Analysis

可以看出GPU非常快，非常适合做这种需要大量并行的计算，即使最朴素的写法都比CPU快很多很多。

Linux* OS

Supported Linux distributions with oneAPI 2024.1

For CPU:

- Red Hat Enterprise Linux* 8.x, 9.x
- Ubuntu 20.04, 22.04
- Fedora* 38, 39
- SuSE LINUX Enterprise Server* 15 SP3, SP4, SP5
- Debian* 11
- Amazon Linux 2022
- Rocky Linux 9
- WSL 2 (except oneCCL)

For GPU:

- Intel® Data Center GPU Max Series (PVC)
 - Red Hat Enterprise Linux* 9.2
 - SuSE LINUX Enterprise Server* 15 SP5
 - Ubuntu 22.04
- Intel® Data Center GPU Flex Series (ATS-M)
 - Red Hat Enterprise Linux* 8.6, 9.2
 - Ubuntu 22.04
- Intel® Arc™ graphics (dGPU)
 - Ubuntu 22.04
- Intel® Iris® Xe graphics (iGFX)
 - Ubuntu 22.04
- Level 0 and OpenCL graphics drivers required to be installed
 - To install the Linux General Purpose Intel GPUs (GPGPU) Driver, see [this article](#) and follow the installation directions for your device.
- GNU* **Bash** is required for local installation and for setting up the environment to use the toolkit.
- Linux kernel 4.11 or higher
- Linux Developer tools component installed, including gcc, g++ and related tools.
- The Intel® compilers are tested with a number of different Linux distributions, with different versions of gcc. The version of glibc you use must be consistent with the version of gcc in use, other distributions may or may not work and are not recommended. If you have questions and have Commercial Support, create a [support ticket](#). Otherwise use the [Intel Community Forums](#) when you need assistance.

Figure 2: mkl

4 Experiment2

如图，mkl变成onemkl后支持很多系统，但不支持这台服务器用的CentOS，于是使用服务器预装的mkl改成onemkl之前的版本mkl2021。这个版本英特尔的编译器还没更名成icpx，还叫icpc。icpc使用方式和icpx接近，只是链接MKL从-qmkl改成-mkl,在编译选项中添加-mkl=sequential会调用串行的MKL库，-mkl=parallel会调用并行的MKL库。控制使用AVX512还是AVX2只需要设置环境变量export MKL_ENABLE_INSTRUCTIONS=AVX2

4.1 Analysis

设置使用AVX2gflops下降了不少，虽然没下降一半。openblas用pthread实现并行还在写，openblas设置使用avx2还是avx512并没有mkl这么方便，在git上openblas的仓库里找到了解决这个问题的方式和设置使用并行还是串行一样需要编译不同版本，在编译时设置对应选项，还在尝试。

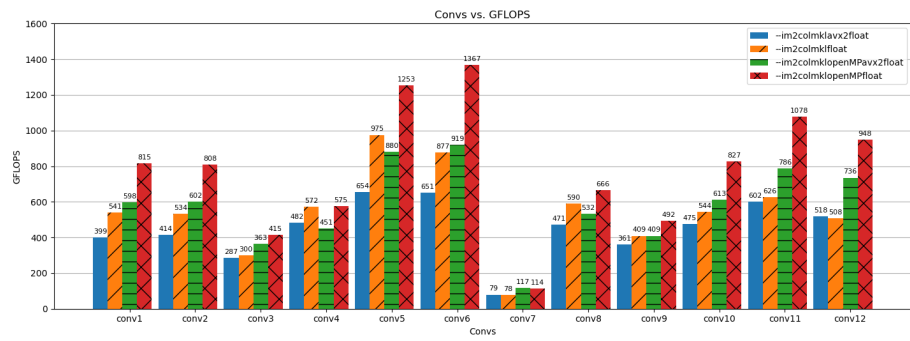


Figure 3: im2col