[Intel] Writing test by Wu Hao:

**Purpose:**

Writing the convolution, relu, pooling function in convolutional nerual networks(CNNs) using c/c++. The input data includes four dimensions: width, height, channels, batchsize, as well as kernels: kwidth, kheight, kchannels, kernel number. Computation efficiency should be calculated via GFLOPS.

Fused method and OpenMP optimization is plus.

**Time complexity and GFLOPS:**

Experimental information:

CPU: Intel® Core™ i7-8700 CUP@ 3.20GHz 6core

Visual Studio 2019 X86

Input data: 512\*512\*3\*24

Convolutional kernel: 3\*3\*3\*12

Type of data and weights: float

Padding 0 : 1

Stride: 1

Bias: 1.3

pooling number: 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method\Function | Convolution | Relu | Pooling | Fused |
| Naive | 10.02s | 0.16s | 0.52s |  |
| Im2col+gemm | 4.87s | 0.24s | 1.16s | 6.20s |
| Im2col+gemm+openmp | 0.71s | 0.03s | 0.17s | 0.80s |
| (Direct)SIMD | 4.33s | 0.05s | 0.03s | 4.96s |
| (Direct)SIMD+openmp | **0.60s** | **0.03s** | **0.01s** | **0.69s** |

The screenshots of the output results of each layer or function can be viewed on

<https://github.com/WuhaoCHN/IntelTest912update/tree/main/Test_results>.

Hardware computation peak: 307.2Gflops

According to:

<https://github.com/WuhaoCHN/IntelTest919update/blob/main/Test_results/SIMD%2BOpenmp2.png>

Computing efficiency for convolution function of SIMD+openmp: [(4+4)\*3\*3+4\*2+1]\*512\*512\*12\*24/0.65 = 9.41Gflops

by Wu Hao

2021/9/19