

面向 FT-M7002 的 Sobel 边缘检测算法优化实现

范明亮¹, 郭子涵¹, 柴晓楠¹, 商建东²

(1. 郑州大学 信息工程学院, 郑州 450001; 2. 国家超级计算郑州中心, 郑州 450001)

摘要: 边缘检测是图像处理与计算机视觉领域中一种重要的图像分析方法, Sobel 算子常用于粗精度的边缘提取, 在图像边缘检测中被广泛应用。随着国产飞腾(FT)系列高性能数字信号处理器的发展, 图像处理领域对 FT 平台的需求日益提高, 同时急需实现面向 FT 平台的高性能图像处理算法。针对上述问题, 在 FT-M7002 平台上对 Sobel 边缘检测算法进行向量并行优化, 使用 FT-M7002 处理器内嵌 SIMD 指令, 挖掘 Sobel 边缘检测算法中的数据级并行性, 同时设计并实现一种字符型与整型数据间的并行化转换接口, 使用循环展开优化方法提升指令节拍数, 通过 DMA 矩阵转置解决数据访存不连续的问题。采用双缓冲技术实现数据传输与内核计算的并行, 从而隐藏数据传输与计算之间的时间间隙。对比分析多种卷积核大小及图片规模下原 Sobel 算法与优化算法的性能, 结果表明, 与原始算法相比, 该优化算法能取得 1.66~3.14 倍的加速比, 此外, 相较 TMS320C6678 处理器上的运行结果, 在 FT-M7002 平台上优化算法可达到 1.87~2.08 倍的加速效果。

关键词: 边缘检测; Sobel 算子; 高性能数字信号处理器; 向量并行; 循环展开

开放科学(资源服务)标志码(OSID):



中文引用格式: 范明亮, 郭子涵, 柴晓楠, 等. 面向 FT-M7002 的 Sobel 边缘检测算法优化实现[J]. 计算机工程, 2022, 48(6): 193-199.

英文引用格式: FAN M L, GUO Z H, CHAI X N, et al. Optimized realization of Sobel edge detection algorithm for FT-M7002[J]. Computer Engineering, 2022, 48(6): 193-199.

Optimized Realization of Sobel Edge Detection Algorithm for FT-M7002

FAN Mingliang¹, GUO Zihan¹, CHAI Xiaonan¹, SHANG Jiandong²

(1. School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China;

2. National Supercomputing Center in Zhengzhou, Zhengzhou 450001, China)

[Abstract] Edge detection is a robust image analysis method used in image processing and computer vision. The Sobel operator is widely used in edge detection and image processing. With the development of domestic FT series high-performance Digital Signal Processors (DSP), the demand for FT platforms in image processing is increasing. Moreover, it is urgent to implement high-performance image-processing algorithms for FT platforms. The vector parallel optimization of the Sobel edge detection algorithm was performed on the FT-M7002 platform to solve the above problem. Single Instruction Multiple Data (SIMD) instructions embedded in the FT-M7002 processor were used to mine the data-level parallelism in the Sobel edge detection algorithm. In addition, a parallel conversion interface between the character and integer data was designed and implemented. The loop unrolling optimization method was used to improve the number of instruction beats, and the problem of discontinuous data access and memory was solved using Direct Memory Access (DMA) matrix transposition. Double buffer technology was used to achieve parallel data transmission and kernel computing to eliminate the time gap between data transmission and computing. The performance of the original Sobel algorithm and the optimization algorithm under various convolution kernel sizes and picture sizes were compared and analyzed. The results showed that compared with the original algorithm, the optimization algorithm could achieve an acceleration ratio of 1.66~3.14 times. Compared with the operation results obtained using the TMS320C6678 processor, the optimization algorithm could achieve an acceleration effect of 1.87~2.08 times on the FT-M7002 platform.

[Key words] edge detection; Sobel operator; high-performance Digital Signal Processor (DSP); vector parallel; loop unrolling
DOI: 10.19678/j.issn.1000-3428.0061903

0 概述

随着大规模集成电路和信息技术的发展, 数字

信号处理器(Digital Signal Processor, DSP)受到广泛关注, DSP 具有高性能、低功耗等特点, 可以满足图像处理实时性以及密集型计算的需求, 因此, 在图像

基金项目: 国家重点研发计划子课题“全球对地观测成果管理及共享服务系统关键技术研究”(2018YFB0505000)。

作者简介: 范明亮(1997—), 男, 硕士研究生, 主研方向为高性能计算、图像处理; 郭子涵、柴晓楠, 硕士研究生; 商建东, 教授、博士、博士生导师。

收稿日期: 2021-06-11 **修回日期:** 2021-07-14 **E-mail:** fanmingliang97@163.com

处理领域得到大范围应用。目前,使用较多的高性能 DSP 主要有 TI 的 C66X 系列,其片上集成 8 个 DSP 核心,单核浮点运算能力可达 20 GFlops^[1-2]。近年来,我国高性能 DSP 的相关研究也取得了一定的成果,其中,由国防科技大学独立研制的 FT-M7002 性能较为突出。FT 处理器片上集成 2 个 DSP 核心和 1 个 CPU 核心,浮点运算能力达 200 GFlops^[3-4]。但是,在目前硬件发展较快的同时也显现出国产 DSP 平台缺乏相应软件支撑的问题。因此,对国产高性能 DSP 处理器相应的软件环境以及函数库进行适配与调优,充分发挥国产 DSP 的特殊硬件架构优势具有重大意义。

边缘检测是图像处理和计算机视觉中的重要研究课题,其目的是识别出图像中那些亮度发生明显变化的点。目前,图像边缘检测^[5]广泛应用于图像识别、医学影像处理、机器学习等^[6-8]领域。传统边缘检测算子包括 Roberts 算子、Sobel 算子、Canny 算子、Prewitt 算子等^[9]。Sobel 算子因其耗时短、具有一定的抗噪性、边缘检测效果好而得到广泛应用。近年来,很多学者基于硬件特性对 Sobel 算法进行改进,并取得了一定的成果。文献[10]基于 FPGA 提出一种多核矩阵处理器,其采用同构多核结构使数据并行传输和运算。文献[11]基于 CUDA 架构,根据 GPU 的并行结构和硬件特点优化数据存储结构。文献[12]基于 Zynq-7000 系列平台,将 ARM 处理器与 FPGA 组合起来并利用异构多核平台实现加速。但是,由于各平台体系结构存在差异性,导致上述优化算法并不适用于 FT 系列的高性能 DSP。

本文面向 FT-M7002 高性能处理器,基于该平台特性对 Sobel 边缘检测算法进行优化。具体地,分析 Sobel 算法特性,判断其是否可以充分利用 FT 平台架构进行优化,针对算法中密集型数据访存运算,使用向量化、循环展开等优化手段实现数据级并行性,以提高向量寄存器、向量运算部件等硬件的资源利用率。在此基础上,针对 DMA 产生的数据传输开销,采用双缓冲区结构使得数据传输与数据计算实现并行,并设计对比实验来测试优化算法的正确性及检测性能。

1 Sobel 算法原理与 FT-M7002 体系结构分析

1.1 Sobel 边缘检测算法

图像边缘即图像中灰度值变化剧烈的位置,梯度算子就是利用图像边缘灰度的突变来检测边缘^[13]。Sobel 算子包含 2 个 3×3 的矩阵,分别对水平和垂直方向上的像素进行滑动检测。假设一副图像中的某个 3×3 区域如图 1 所示,则 Sobel 算子水平和垂直梯度模板分别如图 2(a)、图 2(b)所示。

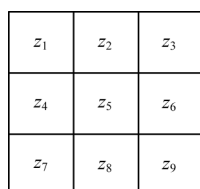


图 1 3×3 区域示意图

Fig.1 3×3 area schematic diagram

-1	0	1	1	2	1
-2	0	2	0	0	0
-1	0	1	-1	-2	-1

(a)水平

(b)垂直

图 2 Sobel 算子的水平和垂直梯度模板

Fig.2 Horizontal and vertical gradient templates of Sobel operator

2 个二维梯度模板可以拆分为 2 个一维卷积核,分别如下:

$$X \text{ 方向卷积模板为 } [-1 \ 0 \ 1] \times \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

$$Y \text{ 方向卷积模板为 } [1 \ 2 \ 1] \times \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

以 X 方向卷积模板为例,其中, $[-1 \ 0 \ 1]$ 卷积计算模板是为了计算 X 方向的差分,也就是得到可能的

边缘点,而 $\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$ 则用来平滑差分处理之后的像素,Y 方

向的横向和纵向卷积模板与 X 相同。根据上述 2 个梯度模板,分别将其与图像像素做卷积计算,公式如下:

$$g_x = (z_1 + 2 \times z_2 + z_3) - (z_7 + 2 \times z_8 + z_9) \quad (1)$$

$$g_y = (z_3 + 2 \times z_6 + z_9) - (z_1 + 2 \times z_4 + z_7) \quad (2)$$

由式(1)、式(2)可以得到该点横向和纵向的亮度差分近似值,从而检测出横向和纵向的边缘。梯度向量的幅值和方向分别如下:

$$g(x,y) = \sqrt{g_x^2 + g_y^2} \approx |g_x| + |g_y| \quad (3)$$

$$\alpha(x,y) = \arctan \left[\frac{g_y}{g_x} \right] \quad (4)$$

根据式(3)可以得到该像素梯度的近似值,最后选取合适的阈值,将计算得到的梯度近似值与阈值进行比较^[14],若梯度近似值大于阈值,则该点为图像的边缘点;否则,该点不是图像的边缘点^[15-17]。由于 Sobel 算子对像素的位置影响做了加权,可以降低边缘模糊程度,因此其检测效果优于传统的 Prewitt 算子、Roberts 算子。

1.2 FT-M7002 体系结构

FT-M7002 高性能 DSP 芯片使用三级存储结构,单个计算核拥有 32 KB 的一级数据缓存以及 512 KB 的向量存储空间(AM),所有计算核共享 2 MB 的全局共享 Cache,核外拥有 32 GB 的大容量 DDR 存储空间^[18-21]。FT-MT2 内核基于超长指令字(VLIW)结构,包含一个五流出标量处理单元(SPU)以及六流出向量处理单元(VPU),2 个处理单元以紧耦合方式工作^[22]。SPU 用于支持标量定点和浮点运算,主要负责数据的串行处理工作。VPU 内有 16 个同构运算簇(VPE),负责对运算量大的密集型数据进行并行计算。直接存储访问(DMA)部件为内核提供了高速数据传输通路,通过配置传输参数启动对特定存储资源的访问,DMA 部件能够实现标量存储空间、向量存储空间与核外 DDR 空间之间的数据交互。FT-MT2 内核结构如图 3 所示。

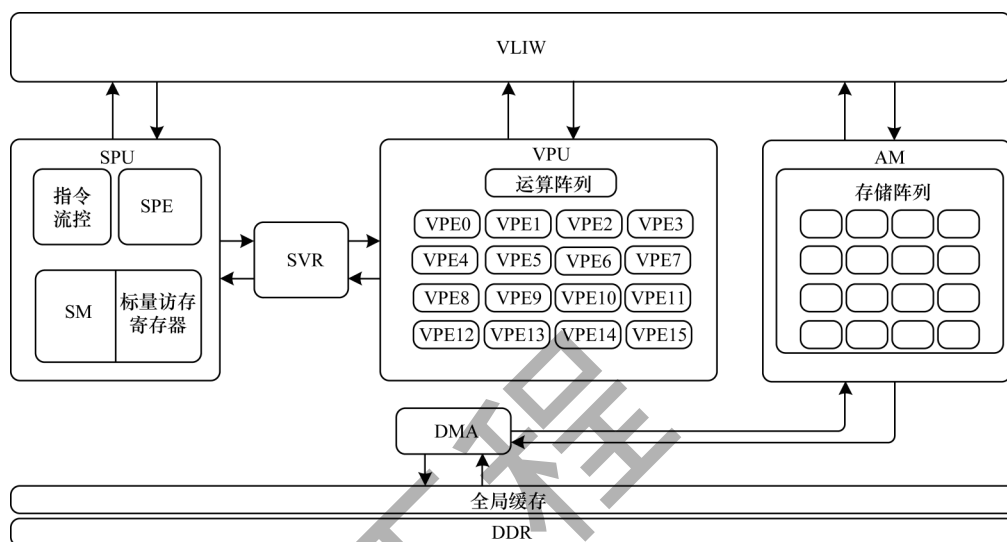


图3 FT-MT2内核结构

Fig.3 FT-MT2 kernel structure

VPE内部集成的4个运算部件可以支持标量定点和浮点运算,即该向量运算单元最多可以同时16路32位数据进行向量运算并且支持混洗/归约操作。FT-M7002特殊的体系结构为大宽度的向量运算提供了高效的数据存储及传输支持,能够在图像处理这种密集重复型计算中发挥优势。

2 基于FT-M7002的Sobel算法

2.1 算法实现

OpenCV中通过获取Sobel边缘检测所需的卷积核进行梯度计算。Sobel的实现是通过分离行卷积和列卷积来进行梯度计算,分别从X方向和Y方向对图片进行卷积操作,每个卷积方向又再次划分了横向卷积与纵向卷积,具体实现流程如图4所示。在进行X方向卷积计算时,先将图像像素点与横向一维卷积模板相乘,然后再将计算之后的卷积结果与纵向一维卷积模板相乘,Y方向与之相同。使用分离的行、列卷积,能够在很大程度上减少计算量,同时又促进了对内存的连续访问。但是,当处理较大的图片时,内存访问量仍然是影响该算法性能的主要因素。

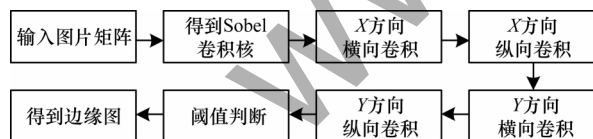


图4 Sobel算法实现流程

Fig.4 Sobel algorithm implementation procedure

2.2 算法优化

Sobel边缘检测的主要计算过程是X方向卷积与Y方向卷积,本文先对其核心计算进行热点分析,发现该部分约占整体运行时间的1/2。因此,本文主要针对该核心计算部分使用SIMD使其并行计算,充分利用FT内核中的向量寄存器资源,减少卷积计

算部分数据运算的访存次数,此外,利用DMA传输机制设置双缓冲结构使数据运算与传输并行,最大程度地提高算法并行性,并通过FT内部指令封装数据类型转换接口,从而扩展平台的兼容性。

2.2.1 向量并行优化

根据Sobel核心计算部分的特性可知,计算过程存在数据连续访问、迭代次数多、计算量大等特点,而FT处理器的内核向量计算特性可以很好地支持其使用SIMD并行化。

核心计算位于循环体内,因此,首先针对循环体进行循环展开,选择适当的展开因子以保证代码体积不会过大,同时提高软件流水并行性,再利用FT内嵌的向量指令对计算过程进行向量化改写。首先,使用数据传输指令M7002_datatrans_cv将内存中的图片数据传输到向量地址空间,通过vec_svbcast将这些数据从标量空间广播到向量空间;然后,利用vec_ld指令从向量空间连续取出16个数据,使用vec_add、vec_muli每次可以完成16个输入数据的乘加操作,计算完成后将结果保存到向量空间;最后,将结果传输回向量地址空间。以3×3核的行卷积为例,具体核心计算部分代码如下:

算法1 部分向量代码改写

```
M7002_vector int *src_cn = (M7002_vector int *)
0x040000000; //起始地址

M7002_vector int *dst_cn = (M7002_vector int *)
0x040040000; //目的地址

M7002_datatrans_cv(src_1, src_cn, (width + cn * 4) * 4);
//数据搬运

M7002_vector int x0, x1, x2, x3, x4, x5, x6;
M7002_vector int k0 = vec_svbcast(kx[0]); //向量广播
M7002_vector int k1 = vec_svbcast(kx[1]);
for (; i <= width - 16; i += 16)
{
```



```

x0 = vec_ld(i + cn, src_cn); //数据加载
x1 = vec_ld(i + cn*2, src_cn);
x2 = vec_ld(i + cn*3, src_cn);
x3 = vec_add(x0, x2); //向量加
x4 = vec_muli(x3, k1); //向量乘
x5 = vec_muli(x1, k0);
x6 = vec_add(x4, x5);
vec_st(x6, i, dst_cn); //存储
}
M7002_datatrans_cv(dst_cn, dst, i * 4);

```

2.2.2 DMA传输优化

在Sobel进行卷积的实际操作中,输入图片对应的矩阵规模一般大于向量存储空间,因此,输入数据需暂时置于核外DDR中,通过DMA将DDR内的数据传输到AM中。

作为数据传输的“中转站”,DMA的传输时间会影响整个算法的效率。为了提高传输速度,本文针对Sobel行卷积操作中的数据连续访问过程,采用双缓冲区传输结构,如图5所示,将AM空间分成AM0和AM1两部分。基于此模式,数据从核外DDR中传入AM0空间,在判断为有效数据之后开始计算,同时,DMA将下一块数据传给AM1。当AM1中的数据开始计算时,AM0输出上一次的计算结果并启动新数据传输,即在DMA进行传输循环检测前将数据的运算与传输并行化,减少一部分运算时间,从而达到性能优化的目的。

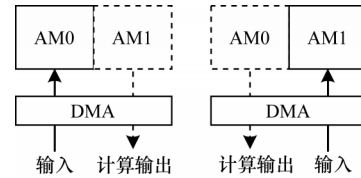


图5 双缓冲机制示意图

Fig.5 Schematic diagram of double buffer mechanism

与行卷积数据传输过程不同的是,列卷积存在不连续的数据访问,无法通过向量指令进行连续存取。本文通过设置DMA的目的索引值作为矩阵每行首元素的地址差,将行卷积矩阵中的每一行按列存储在矩阵中,最终实现矩阵转置。

2.2.3 数据类型转换

在图像处理库中,默认图片矩阵类型为8u (unsigned char),但是FT向量处理器没有直接处理8u类型的指令,针对该问题,本文利用VPE内部混洗/规约等部件实现数据类型转换并封装成接口,从而实现 unsigned char 和 unsigned int 之间的相互转换,扩充FT平台所支持数据的类型。

FT-M7002向量寄存器可一次加载512位数据到向量寄存器中。由于每个VPU包含16个VPE,因此每个VPE中存放4个 unsigned char 类型数据,要转换成 unsigned int 类型,即将每个VPE中的4个数据分别放入4个VPE中。如图6所示,将每个原始VPE中的数据与0XFF按位相与,即可得到新的VPE中的数据存放,连续操作3次就可实现数据转换。

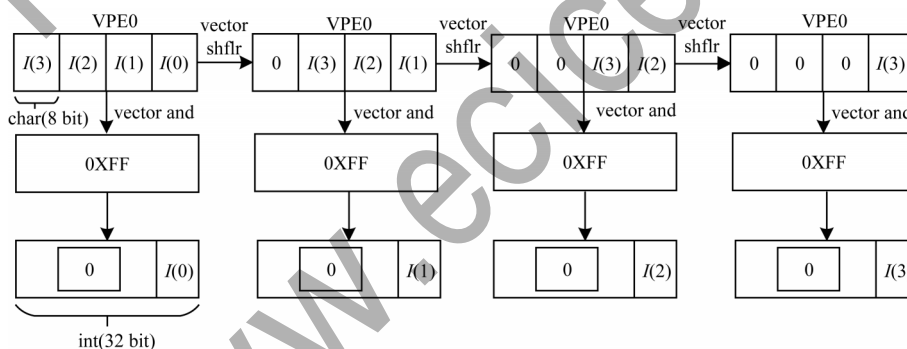


图6 char to int类型转换(VPE0实现)

Fig.6 char to int type conversion (implementation of VPE0)

图6只展示了VPE0的实现,可以对VPU中的VPE0~VPE15分别做上述操作。值得注意的是,上述操作得到的数据并不是按序存放的,因此,需要对这些数据进行一次混洗(shuffle)操作,从而得到正确的数据排列方式。混洗单元通过合理配置混洗模式,可以实现VPE中输入数据的按序存放。

3 实验结果与分析

本文以3×3、5×5、7×7卷积核为例,分析不同尺寸输入图片对算法加速比的影响,同时将本文优化算法与TMS320C6678中的Sobel算法进行性能对比,以分析2类处理器的单核计算能力。

3.1 实验环境

在飞腾平台的FT-M7002开发板上实现Sobel算

法优化,调用定时器的计时函数来测试优化后程序段的执行时间,将FT-M7002与TI平台的高性能数字信号处理器TMS320C6678进行对比分析。实验参数设置如表1所示。

表1 实验平台参数设置

参数	FT-M7002	TMS320C6678
主频/GHz	1.00	1.25
L1D/(KB·core ⁻¹)	32	32
AM/KB	512	—
编译器	M7002编译器	TI-C6000

3.2 正确性分析

Sobel边缘检测算法能在FT平台上正确运行是算法优化的基础。为了验证优化算法的正确性,将不同尺寸大小的图片输入优化算法并运行于FT平台,将其结果矩阵与OpenCV图像库中Sobel算法的结果矩阵进行对比,将2个结果图片矩阵按像素点相减取绝对值,小于浮点数后6位则认为结果正确。经验证,本文优化算法均能得到正确的处理结果。

3.3 性能分析

以Sobel卷积核 3×3 、 5×5 、 7×7 为例,分别以不同数据类型及尺寸的输入图片运行Sobel串行算法和优化算法,在FT-M7002处理器上的运行结果(时钟周期)如表2、表3所示。从中可以看出,对于不同数据类型的输入矩阵,本文优化算法可取得1.66~3.14倍的加速效果。加速效果整体趋势如图7、图8所示,其中,横坐标表示输入矩阵规模,纵坐标表示本文优化算法相较串行算法所取得的加速比。

表2 8u类型优化前后的性能比较

阶段	640×360像素			640×480像素			800×500像素			1 024×768像素		
	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7
优化前	16 236 241	25 156 358	36 743 541	21 655 812	33 145 483	48 719 305	27 630 329	43 180 339	63 598 876	54 739 200	87 467 430	125 680 104
优化后	9 295 663	14 083 641	20 285 646	12 779 244	18 935 806	27 817 701	15 944 165	24 320 287	36 065 471	32 415 762	49 238 693	70 985 214
加速比	1.74	1.78	1.80	1.70	1.75	1.75	1.73	1.77	1.78	1.69	1.77	1.77

表3 32f类型优化前后的性能比较

阶段	640×360像素			640×480像素			800×500像素			1 024×768像素		
	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7
优化前	18 433 828	27 385 114	39 998 071	24 610 004	36 494 627	52 165 627	31 856 361	48 226 313	68 478 785	62 668 168	94 232 948	138 360 266
优化后	11 145 001	11 102 468	13 320 043	14 741 788	14 896 865	17 890 032	19 222 606	19 150 093	23 004 412	37 417 169	37 908 492	45 408 828
加速比	1.66	2.46	3.00	1.66	2.44	2.91	1.66	2.51	2.97	1.67	2.48	3.04

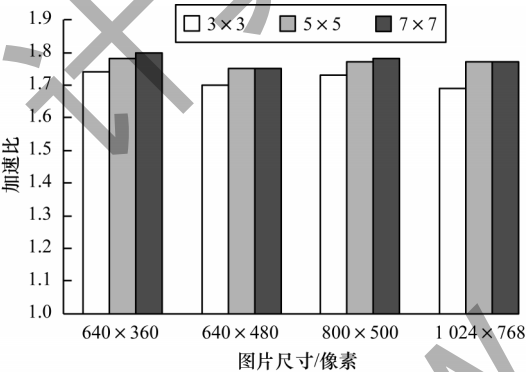


图7 8u类型Sobel优化算法相较串行算法的加速比
Fig.7 The speedup ratio of 8u type Sobel optimization algorithm compared with serial algorithm

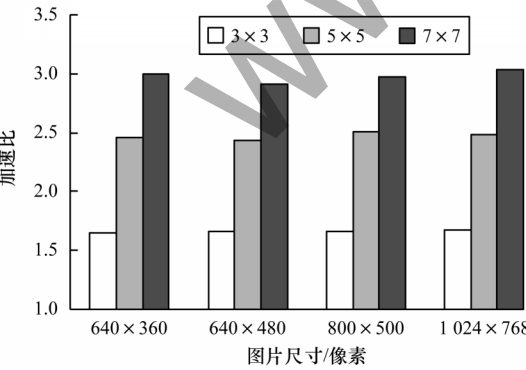


图8 32f类型Sobel优化算法相较串行算法的加速比
Fig.8 The speedup ratio of 32f type Sobel optimization algorithm compared with serial algorithm

3.3.1 unsigned char类型测试结果与分析

表2所示为8u类型不同矩阵规模分别在卷积核为 3×3 、 5×5 、 7×7 上运行时的时钟周期以及加速比,图7为其对应的加速效果柱状图。可以看出,整体加速比为1.75左右,从趋势上来看,在同一输入矩阵规模下,随着卷积核的增大,计算量也在增大,加速比也增大;在卷积核一定的情况下,加速比随着矩阵规模的增大而缓慢增大,最后趋于平稳,这是因为数据存放在DDR中,随着矩阵规模的增大,虽然使用了DMA双缓冲结构,但是传输仍然耗时从而使得加速效果不明显,另外还需要进行数据类型转换,输入矩阵规模越大,类型转换也就越耗时。

3.3.2 float类型测试结果与分析

表3所示为float类型不同矩阵规模分别在卷积核为 3×3 、 5×5 、 7×7 上运行时的时钟周期以及加速比,图8为其对应的加速效果柱状图。对比表2、表3可以看出, float类型的整体加速效果明显优于 unsigned char类型, 主要是因为FT向量处理器可以直接处理float类型的数据,不需要进行数据类型转换。对于不同的卷积核,核越大,加速效果越明显,这是因为随着核的不断增大,循环内每次计算的数据规模也就越大,因此,优化算法能更好地提升循环内数据的并行性。

3.3.3 平台对比分析

本次实验的对比测试所用图片为 unsigned char类型的三通道图片。在TMS320C6678中运行Sobel算法,在FT-M7002中运行本文优化算法,分别使用TI与FT

平台的计时函数记录运行时间,FT时钟周期换算为TI时钟周期时需要乘16。测试结果如表4所示,整体加速效果如图9所示。从中可以看出,对于不同大小的

图片矩阵,FT-M7002向量处理器比TI6678多核处理器快了1.87~2.08倍。因此,对于单核密集型运算,FT-M7002平台相较TI6678平台有一定的计算优势。

表4 FT与TI平台的比较

Table 4 Comparison between FT and TI platforms

平台	640×360像素			640×480像素			800×500像素			1 024×768像素		
	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7
TI	304 028 343	463 771 941	627 036 975	403 527 838	615 710 464	832 786 810	521 018 033	809 257 007	1 089 522 833	1 028 213 741	1 603 484 225	2 225 269 367
FT	9 295 663	14 083 641	20 285 646	12 779 244	18 935 806	27 817 701	15 944 165	24 320 287	36 065 471	32 415 762	49 238 693	70 985 214
加速比	2.04	2.05	1.93	1.97	2.03	1.87	2.04	2.08	1.88	1.98	2.03	1.96

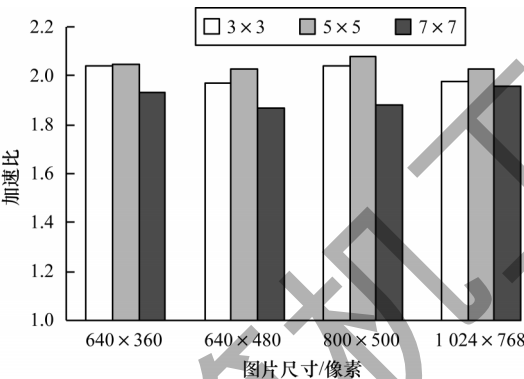


图9 FT-M7002平台较TI6678平台的加速比
Fig.9 The speedup ratio of FT-M7002 platform compared with TI6678 platform

4 结束语

本文分析FT-M7002硬件平台架构和Sobel边缘检测算法,基于FT内核结构特点对Sobel算法进行优化。为了充分发挥FT-M7002平台的高性能计算特性,使用FT内嵌指令对算法进行向量并行优化,利用循环展开的优化手段充分发掘数据级并行性。设计DMA双缓冲区传输结构同步数据的传输与运算,对列卷积进行矩阵转置以解决其不能连续访存的不足。针对FT平台向量部件不支持unsigned char类型访存操作的问题,设计相应的数据转换接口。对多种输入图片矩阵规模和卷积核大小进行对比测试,结果表明,在保证正确性的基础上,该优化算法相较串行算法获得了1.66~3.14倍的加速比,相较TMS320C6678处理器也取得了1.87~2.08倍的加速效果,充分验证了本文所使用优化方法的有效性以及FT-M7002平台的高性能计算优势。但是,本文目前仅针对单核进行了优化研究,下一步将探索多核优化问题,并在FT-M7002平台上实现OpenCV图像算法库的整体优化。

参考文献

[1] 周佩,周维超,王凯凯. TMS320C6678多核DSP并行访问存储器性能的研究[J]. 微型机与应用,2014,33(13):20-24.
ZHOU P, ZHOU W C, WANG K K. Research on the performance of TMS320C6678 multicore DSP parallel memory access[J]. Microcomputer & Its Applications, 2014,33(13):20-24. (in Chinese)
[2] 赵一需. DSP的发展与应用[J]. 电子技术与软件工程,

2018(6):92.
ZHAO Y P. Development and application of DSP[J]. Electronic Technology & Software Engineering, 2018(6): 92. (in Chinese)
[3] 张军阳,郭阳,扈啸. 二维矩阵卷积的并行计算方法[J]. 浙江大学学报(工学版),2018,52(3):515-523.
ZHANG J Y, GUO Y, HU X. Parallel computing method for two-dimensional matrix convolution [J]. Journal of Zhejiang University(Engineering Science), 2018,52(3): 515-523. (in Chinese)
[4] 李勇,陈书明,陈胜刚. 一种基于YHFT-Matrix DSP的去块效应滤波算法的向量化实现[J]. 计算机工程与科学, 2014,36(2):206-210.
LI Y, CHEN S M, CHEN S G. Vectorizable implementation of adaptive deblocking filter on YHFT-Matrix DSP[J]. Computer Engineering and Science, 2014,36(2):206-210. (in Chinese)
[5] LIU F, GUAN G X, ZHAO H M, et al. Fast processing of UAV remote sensing image based on DSP hardware simulation platform[J]. Journal of Physics Conference Series, 2020,1486(7):072072.
[6] OKAMOTO T, KOIDE T, YOSHIDA S, et al. Implementation of computer-aided diagnosis system on customizable DSP core for colorectal endoscopic images with CNN features and SVM[C]//Proceedings of 2018 IEEE Region Conference. Washington D. C. , USA: IEEE Press, 2018:1663-1666.
[7] AKASAPU A, SAILAJA V, PRASAD G. Implementation of Sobel filter using CUDA[J]. IOP Conference Series: Materials Science and Engineering, 2021,1045(1):012016.
[8] 罗紫阳,李雪梅,陈鹏. 基于FPGA的Sobel算子边缘检测算法的实现[J]. 电子制作,2020(21):6-7.
LUO Z Y, LI X M, CHEN P. Implementation of Sobel operator edge detection algorithm based on FPGA[J]. Electronic Manufacturing, 2020(21):6-7. (in Chinese)
[9] YANG S H, HSIAO S J. H. 266/VVC fast intra prediction using Sobel edge features[J]. Electronics Letters, 2021, 57(1):11-13.
[10] 陈鹏. 图像Sobel处理的FPGA多核矩阵实现[D]. 沈阳: 沈阳理工大学,2013.
CHEN P. FPGA multicore matrix's achieve of the image Sobel processing [D]. Shenyang: Shenyang Ligong University, 2013. (in Chinese)
[11] 左颖睿,张启衡,徐勇,等. 基于GPU的快速Sobel边缘检测算法[J]. 光电工程,2009,36(1):8-12.
ZUO H R, ZHANG Q H, XU Y, et al. Fast sobel edge detection algorithm based on GPU[J]. Opto-Electronic Engineering, 2009,36(1):8-12. (in Chinese)

- [12] 王威. 基于ARM-FPGA异构多核平台上图像处理算法的加速研究[D]. 西安:西安电子科技大学,2019.
WANG W. Accelerated research on image processing algorithm based on ARM-FPGA multi-core platform[D]. Xi'an: Xidian University, 2019. (in Chinese)
- [13] HADI SAPUTRA V, HERWINDIATI D E, SUTRISNO T. Car shape clustering using Sobel edge detection with divisive average linkage and single linkage algorithm (case: bus, Sedan, citycar, mpv, and truck)[J]. IOP Conference Series: Materials Science and Engineering, 2020, 1007(1): 012136.
- [14] TIAN R, SUN G L, LIU X C, et al. Sobel edge detection based on weighted nuclear norm minimization image denoising[J]. Electronics, 2021, 10(6): 655.
- [15] 李丹阳. 基于Soble算子的图像边缘检测优化设计[J]. 数字技术与应用, 2017(11): 137-138.
LI D Y. A optimization design of edge detection based on Soble algorithm[J]. Digital Technology and Application, 2017(11): 137-138. (in Chinese)
- [16] 缪成根, 刘琛. 基于改进索贝尔算子的灰度图像边缘检测[J]. 物联网技术, 2020, 10(11): 37-38, 41.
MIAO C G, LIU C. Gray image edge detection based on improved Sobel operator [J]. Internet of Things Technologies, 2020, 10(11): 37-38, 41. (in Chinese)
- [17] RAVIVARMA G, GAVASKAR K, MALATHI D, et al. Implementation of Sobel operator based image edge detection on FPGA[J]. Materials Today: Proceedings, 2021, 45: 2401-2407.
- [18] WANTO A, RIZKI S D, ANDINI S, et al. Combination of Sobel+Prewitt edge detection method with Roberts+Canny on passion flower image identification [J]. Journal of Physics: Conference Series, 2021, 1933(1): 012037.
- [19] 孙广辉, 扈啸, 王蕊. 基于FT-M7002的OpenCV移植与优化[C]//第二十二届计算机工程与工艺年会暨第八届微处理器技术论坛论文集. [出版地不详]: 中国计算机学会计算机工程与工艺专业委员会, 2018: 9.
SUN G H, HU X, WANG R. Transplantation and optimization of OpenCV based on FT-M7002 [C]//Proceedings of the 22nd Annual Conference on Computer Engineering and Technology and the 8th Microprocessor Technology Forum. [S. l.]: Professional Committee of Computer Engineering and Technology of China Computer Society, 2018: 9 (in Chinese)
- [20] 沈洁, 龙标, 姜浩, 等. 飞腾处理器上向量三角函数的设计实现与优化[J]. 计算机研究与发展, 2020, 57(12): 2610-2620.
SHEN J, LONG B, JIANG H, et al. Implementation and optimization of vector trigonometric functions on Phytium processors[J]. Journal of Computer Research and Development, 2020, 57(12): 2610-2620. (in Chinese)
- [21] 郭恒亮, 柴晓楠, 韩林, 等. Canny边缘检测算法在飞腾平台上的实现与优化[J]. 计算机工程, 2021, 47(7): 37-43.
GUO H L, CHAI X N, HAN L, et al. Implementation and optimization of Canny edge detection algorithm on FT platform[J]. Computer Engineering, 2021, 47(7): 37-43. (in Chinese)
- [22] 孙广辉. 基于FT-M7002的OpenCV移植与优化[D]. 西安: 西安电子科技大学, 2019.
SUN G H. OpenCV transplantation and optimization based on FT-M7002[D]. Xi'an: Xidian University, 2019. (in Chinese)