

JPEG2000 并行阵列式小波滤波器的 VLSI 结构设计

兰旭光, 郑南宁, 梅魁志, 刘跃虎

(西安交通大学人工智能与机器人研究所, 陕西西安 710049)

摘 要: 提出一种基于提升算法实现 JPEG2000 编码系统中的二维离散小波变换(Discrete Wavelet Transform)的并行阵列式的 VLSI 结构设计方法. 利用该方法所得结构由两个行处理器, 一个列处理器以及少量行缓存组成; 行列处理器内部是由并行阵列式的处理单元组成; 能使行和列滤波器同时进行滤波, 用优化的移位加操作替代乘法操作. 整个结构采用流水线的设计方法处理, 在保证同样的精度下, 大大减少了运算量和提高了硬件资源利用率, 几乎达到 100%, 加快了变换速度, 也减少了电路的规模. 该结构对于 $N \times N$ 大小的图像, 处理速度达到 $O(N^2/2)$ 个时钟周期. 二维离散小波滤波器结构已经过 FPGA 验证, 并可作为单独的 IP 核应用于正在开发的 JPEG2000 图像编解码芯片中.

关键词: 二维离散小波变换; VLSI; JPEG2000; 图像压缩; 并行阵列式结构; 提升方法

中图分类号: TN406 **文献标识码:** A **文章编号:** 0372-2112 (2004) 12 1806-04

Parallel Array VLSI Architecture Design of 2D DWT for JPEG2000

LAN Xu2guang, ZHENG Nan2ning, MEI Kui2zhi, LIU Yue2hu

(The Institute of Artificial Intelligence and Robotics, Xi. an Jiaotong University, Xi. an, Shaanxi 710049, China)

Abstract: A real2time parallel array architecture is proposed to perform the forward and inverse discrete wavelet transform (DWT) by using lifting scheme for the filters recommended in JPEG2000. It consists of two row processors and one column processor. The signals are processed in parallel way. Multiplications are substituted for shift2add operations. The whole architecture is optimized in the pipeline design way to speed up and achieve higher hardware utilization. It performs a decomposition in approximately $N^2/2$ clock cycles for an $N \times N$ image. A compact and independent IP core based on the architecture for JPEG2000 VLSI implementation has been demonstrated in FPGA.

Key words: 2D discrete wavelet transform (DWT); VLSI; JPEG2000; image processing; parallel array; lifting schemes

1 引言

小波在信号处理、图像压缩视频处理等领域已得到广泛的应用, JPEG2000 就是基于小波变换(DWT)压缩编码的最新静态图像压缩标准^[1]. 该标准采用二维离散小波变换 DWT 代替 JPEG 中的离散余弦变换(DCT, Discrete cosine transform), 性能优于基于 DCT 的编码系统; 可以实现按分辨率渐进和按图像质量渐进传输等. 在 JPEG2000 标准中采用 lifting^[2]算法来实现离散小波变换, 包括 5/3 和 9/7 滤波器. 这种算法的复杂度较标准卷积算法大约减少一半(对 D. 9/7 达到 64%)^[2].

在二维图像的小波变换一般的处理方法是直接法^[3], 先对图像的行(列)进行滤波变换, 把行(列)处理的中间结果存储, 然后进行列(行)滤波变换, 这需要存储大量的中间结果, 对于 $N \times N$ 的图像, 大约是 $N^2/2$ 存储空间, 消耗的时钟数是 $O(2N^2)$. 虽然控制逻辑简单, 但增加硬件开销, 减少硬件的利用率, 限制了芯片处理数据的速度. 另一种方法是并行法^[4], 用附加的处理器来代替大量的中间存储, 节省了存储空间, 这

种方法用两个并行的处理器来计算行和列方向的滤波, 仅需要 $N \times L$ (L 是滤波器的长度)大小的中间存储. 完成多级分解处理速度是 $O(N^2)$ 个时钟周期, 这不包含边界对称扩展. 这种方法附加了处理器, 增加了逻辑控制单元和时序的复杂度, 且硬件利用率较低. 文献[5]提出一种基于提升算法的 VLSI 实现结构, 它是由两个行处理器, 两个列处理器和两个存储模块组成, 但是这种结构的控制逻辑比较复杂, 且处理速度比较慢, 分解一级需要 $O(N^2)$ 个时钟周期.

本文提出一种基于提升算法的并行、行列并行处理阵列式的二维离散小波滤波变换的 VLSI 实现结构. 其中, 二维 DWT 采用 JPEG2000 中 9/7 和 5/3 滤波变换, 但不限于此, 对于基于提升算法的二维离散小波变换均适用, 本文以 DAUBECHIES9/7 为例进行说明. 采用基于行的变换方式^[5]和最高分解级数为 5 级 Mallat 分解结构, 边界采用对称扩展处理. 我们用 $(L+1) \times N$ 条行缓存(Line Buffer)存储中间结果, 对 9/7 来说就是 $10 \times N$, 这部分存储器将在芯片内部用一读写的双端口同步 SRAM 实现. 在存储的同时读取这些中间

结果进行列滤波, 减少了硬件开销的同时, 实现了行与行、行与列并行计算; 用移位加代替乘法操作, 且在整个结构设计中, 采用流水线的设计方法处理, 增加了处理速度和硬件利用率. 该结构完成一级小波分解的所用的时钟周期数是 $O(N^2/2)$, 包含边界对称扩展, 且控制逻辑比文献[7]提出的结构简单. 小波变换后的系数经量化后送入 EBCOT 编码.

2 DAUBECHIES 9/7 提升算法实现

JPEG2000 标准中的一组双正交小波滤波器))) DAUBECHIES 9/7 双正交小波分解为提升方法(见图 1)实现如下^[1]:

step1: $Y(2n+1) = X_{ext}(2n+1) + A@(X_{ext}(2n) + X_{ext}(2n+2)),$
 $i_0 - 3F2n+1 < i_1+3$

step2: $Y(2n) = X_{ext}(2n) + B@ (Y(2n-1) + Y(2n+1)),$
 $i_0 - 2F2n < i_1+2$

step3: $Y(2n+1) = Y(2n+1) + C@ (Y(2n) + Y(2n+2)),$
 $i_0 - 1F2n+1 < i_1+1$

step4: $Y(2n) = Y(2n) + D@ (Y(2n-1) + Y(2n+1)),$
 $i_0F2n+1 < i_1$

step5: $Y(2n+1) = -K@Y(2n+1), \quad i_0F2n < i_1$

step6: $Y(2n) = Y(2n)/K, \quad i_0F2n < i_1$

其中 $i_0, (i_1-1)$, 分别表示输入一行(或列)数据的开始索引和最后索引.

从图 1 还可以看出, 小波逆变换正好和正变换相反, 实现起来相当方便, 这也是提升方法的突出优点.

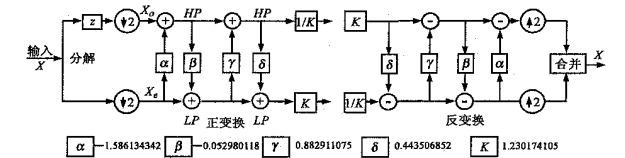


图 1 Daubechies 9/7 小波提升实现

3 把乘法操作优化为移位加操作

由于乘法器占用很大的硬件资源, 不利于芯片实现, 且鉴于小波滤波器的系数是固定的, 我们把乘法操作优化为移位寄存器和加法器操作, 现把滤波器小波系数量化如下:

$A = -1.586134342 = -1.1001011000001;$
 $B = -0.052980118 = -0.000011011001;$
 $C = 0.882911075 = 0.1110001;$
 $D = 0.443506852 = 0.0111000110001;$
 $K = 1.230174105 = 1.0011101011101;$
 $1/K = 0.812893066 = 0.1101000000011;$

因为 A 为负值, 所以先取 Num 的二进制补码, 然后相乘则有 $A@Num = Num + Num m1 + Num m4 + Num m6 + Num m7 + Num m13$ 即把乘法操作优化为移位加操作, 其它系数均作同样的处理. 在我们设计的 VLSI 结构中, 采用 24 位定点数进行运算, 后 13 位为小数.

4 二维离散小波变换实现并行的体系结构

图 2 表示二维离散小波变换的体系结构, 输入图像数据 (一个时钟输入两行) 先经过控制单元、行扩展模块和两个相同的行处理器, 再输入到行缓存存储控制器、和列处理器以及地址生成器, 最后输出小波分解系数到存储器. 控制单元是根据当前分解级数和所要求的小波分解级数来判断是否进行下一级小波分解, 并且控制变换输入的数据 (注意一个时钟输入两个数据), 是从原始图像读取数据还是从 LL 子带中读取数据. 行扩展

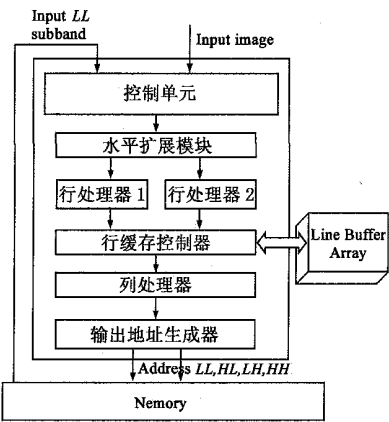


图 2 二维离散小波变换体系结构

模块是对输入进行行边界对称扩展, 两个行处理器是分别对扩展后的两行数据进行滤波处理. 行缓存控制器将一定数量的行变换结果存放在缓存中 (Line Buffer Array), 并实现从中读取数据送入列处理器, 以实现行列变换的并行化, 我们用 10 个行缓存. 列处理器是将从行缓存中读出数据进行列滤波处理, 地址生成器根据列滤波输出结果产生分配地址, 并把当前小波分解的系数写入外部存储器. 按此体系结构, 不但可以使行和列方向内的高通滤波和低通滤波实现并行化, 并且可以使行、列方向滤波变换实现并行化.

4.1 采用流水线的设计方法优化小波变换体系结构

在整个体系结构中, 我们用流水线的设计方法进行优化. 我们以行处理单元 PE. ROW1 为例进行说明, 表 1 给出了相应的访问存储器和运算操作时序, 其他模块作同样的流水线处理. 从表 1 和图 4 中可以看出, PE. ROW1 处理单元是三级流水线优化处理: 第一级把偶数点相加; 第二级相加结果与提升系数 (A) 相乘 (优化为移位加操作); 第三级是将乘积与奇数点相加. 实现此流水线操作仅需要延迟一个时钟.

表 1 PE. ROW1 的时序表

PE. ROW1			
Clock	Adder	Shift Adder	Adder
1	$X_{0,0}$		
2	$X_{0,2} + X_{0,0}$		
3	$X_{0,4} + X_{0,2}$	SA_1	
4	$X_{0,6} + X_{0,4}$	SA_2	$SA_1 + X_{0,1}$
5	$X_{0,8} + X_{0,6}$	SA_3	$SA_2 + X_{0,3}$
6	$X_{0,10} + X_{0,8}$	SA_4	$SA_3 + X_{0,5}$
7	$X_{0,12} + X_{0,10}$	SA_5	$SA_4 + X_{0,7}$
10	)	)	)

说明: $SA_1 = A(X_{0,2} + X_{0,0})$, 即始终对上一时钟的值进行运算; $X_{i,j}$ 表示输入信号的值

41.2 行处理器结构

行处理器由相同的行处理器 1、2 并行组成,如图 2。每个行处理器由 4 个串行的处理单元组成,如图 3 所示。8 个处理单元形成并行阵列式行处理器结构,这些处理单元都是三级流水线优化处理,见图 4。

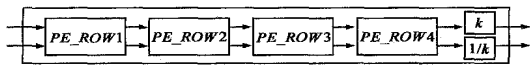


图 3 行处理器 1、2 的结构

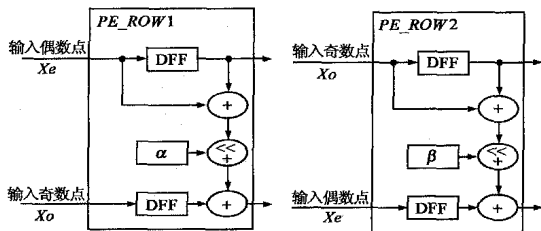


图 4 行处理单元 PE_ROW1(同 PE_ROW3)和 PE_ROW2 结构 (同 PE_ROW4)

与 PE_ROW1 相同的处理方式,依次进入 PE_ROW2, PE_ROW3, PE_ROW4 处理单元,其中 PE_ROW1 和 PE_ROW3 结构相同,PE_ROW2 和 PE_ROW4 结构相同,它们的提升系数分别是 A, B, C, D。随着数据的连续输入,行滤波器连续处理。这样可以充分利用硬件,使其利用率达到 100%。而且控制非常简单,只是利用几个延迟寄存器(DFF)即可实现滤波的连续处理。两个并行行滤波的结果存入 Line Buffer 阵列中,本实例,采用了 $(L+1)$ 即 10 条 Line Buffer,位宽度 24 位,长度为图像宽度,256。之所以用 10 条,是为了实现行滤波和列滤波并行进行,这样就大大节省了存储行变换中间结果的存储空间,使得硬件更为紧凑,不仅减少面积,同时提高了运算速度,也符合芯片设计思想。从结构中可以看到, D9/7 双正交小波滤波器的行滤波变换只需要 16 个延迟寄存器, 16 个加法器和 10 个移位寄存器, 10 个移位加就可以实现。由于移位寄存器可以用线与组合逻辑来实现,所以移位寄存器基本上没有增加硬件的开销。与传统方法相比,此结构极大地减小了硬件的开销和结构的复杂度(无乘法器)。

41.3 行缓存控制器

41.3.1 写控制 由于采用两个行处理器并行,所以一个时钟应写入两个行变换后数据。可以根据两个行处理器输出数据的使能线产生写控制使能。第一次写入第 1、2 行存中,第二次写入第 3、4 行存中,依次类推,第五次写入第 9、10 行存中,然后又从第 1、2 行存开始写,依次循环,实现连续写入。注意这里每条行存的大小都大于或等于输入图像宽度。

41.3.2 读控制 读控制使能可以根据写控制使能产生。对 D_9/7 小波,当开始写入第 5、6 条行存中时,就可以从行存中读取数据,一次读取 5 个数据即从第 1 条行存到第 5 条行存第一个开始,连续读取,垂直边界对称扩展后

为 9 个数据输出到列处理器中。当列方向边界对称扩展完毕后,一次从行存中读取 9 个数据,读取的行存依赖于读使能。这里读控制始终滞后写控制一个时钟且所用时钟都是系统时钟,这样就实现了写和读操作同时连续进行,不致产生干涉。如表 2 所示,当时钟 1 写入第 9、10 条行存时,时钟 2 就可以一个时钟读取 $Y_{0,0} \sim Y_{8,0}$ 9 个数据,这时已写入 $Y_{8,1} Y_{9,1}$,这样时钟 3 读操作就可以进行,读取 $Y_{0,1} \sim Y_{8,1}$ 9 个数据,依次循环,实现流水线读写操作。

而 5/3 小波只用 6 条行存,从写入第 3、4 条行存中就开始读,一次读取 3 个值,经扩展后为 5 个值,输入到列滤波处理器中,其他处理类似 9/7 小波。

表 2 行存的读写控制

Line Buffer Array		列 1	列 2	列 3	)
WRITE	Line 1	$Y_{0,0}$	$Y_{0,1}$	$Y_{0,2}$	)
	Line 2	$Y_{1,0}$	$Y_{1,1}$	$Y_{1,2}$	)
	Line 3	$Y_{2,0}$	$Y_{2,1}$	$Y_{2,2}$	)
	)	)	)	)	)
READ	)	)	)	)	)
	Line 9	$Y_{8,0}$	$Y_{8,1}$	$Y_{8,2}$	)
	Line 10	$Y_{9,0}$	$Y_{9,1}$	$Y_{9,2}$	)
CLOCK		1	2	3	)

41.4 列处理器结构

根据列处理器的处理数据依次减少的特点,我们设计其并行阵列式结构。这种结构由四种处理单元组成,其中有 4 个并行的 PE_COLUMN1, 3 个并行的 PE_COLUMN2, 2 个并行的 PE_COLUMN3, 1 个 PE_COLUMN4,如图 5 所示。这四种处理单元之间仅仅是提升系数不同,分别是(A, B, C, D),硬件结构是相同的,其硬件资源利用率达到 100%。这些处理单元都经过流水线优化处理,如图 6 所示。也是三级流水线处理:第一级把偶数点(奇数点)相加;第二级其和与提升系数相乘;第三级是将乘积与奇数点(偶数点)相加。列处理器处理的数据是来自行滤波的结果,所以当列变换所需的数据数满足时,就可以从行缓存中读取数据,开始列滤波。9 个数据依次进入 10 个处理单元,最后一个时钟输出两个数据,分别是 IL、IH,或者是 HL、HH。地址生成器根据输出数据格式,产生相应的地址,写入外部存储器。从结构图中可以看出,列处理器需要 10 个延迟寄存器, 20 个加法器, 12 个移位寄存器和 12 个移位加操作,加速了硬件处理速度,也实现了行滤波和列滤波的并行化。

41.5 小波变换的多级分解

小波多级分解是利用有限状态机实现的。小波滤波器开

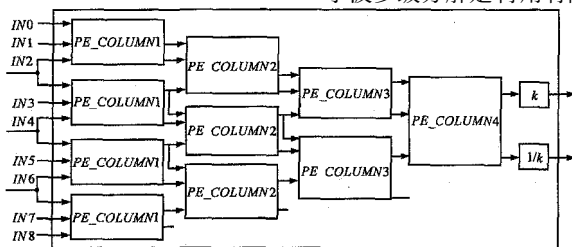


图 5 列处理器结构

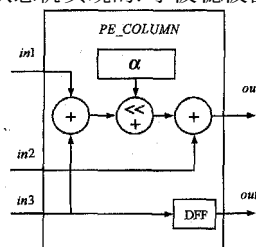


图 6 列处理器处理单元结构

始进入初始 IDLE 状态, 当前分解级数结束(level. finish 为高) 且小波总分解级数大于当前分解级数(level > current. level) 时即可进行下一级分解, 即从原图象数据或者 LL 子带中读取数据, 进入小波变换器, 依次进行变换处理. 设计时, 我们将小波变换最大分解级数定为 5 级, 这对于一般的图像已经足够. 这里, level 表示总的小波分解级数, current. level 表示当前分解级数, level. finish 表示当前分解级数结束. Idle 表示初始状态, 状态 level 1~ level5 表示分解级数为 1~ 5.

4.1.6 结合 EBCOT 编码^[6]

JPEG2000 编码系统首先经过小波变换, 经过量化后进入 EBCOT 编码. 由于小波 Mallat 分解成四个子带是同时完成的, 所以在小波并行计算过程中, 当在 LH, HL, HH 子带中形成一个码块大小的数据即可读取送入 EBCOT 编码, 即在小波变换的过程中就可以进行后续编码, 这大大提高了芯片处理速度, 增加了硬件资源的利用率.



图 7 小波 5 级分解图(硬件仿真时, 将原图 1024 @ 1024 @ 8 分解成 256 @ 256 @ 8 图像片)

4.1.7 性能分析

本文所提出的结构已成功在 FPGA 验证. 输入图像下载到 SDRAM 中, 经过前处理, 数据输入到 ALTERA 公司的芯片 EP1S25. 采用 0125 TSMC 工艺库, 经 SYNOPSIS 综合, 得到小波核行列处理器的逻辑面积约为 61000 个等效门(标准 2 输入与非门), 关键路径延迟 51.6ns, 时钟设置为 6ns, 时钟不确定性设为 0.13ns. 数据是 24 位定点数, 仿真结果和软件仿真完全相同, 表明了所提出并行阵列式流水线的二维离散小波变换结构的正确性, 滤波处理并行性; 其硬件资源利用率, 几乎达到 100%. 仿真采样时钟频率是 45MHz, 对一幅 1024 @ 1024 @ 8 比特的图像, 用 9/7 小波分解 5 级, 见图 7, 仅需要耗时 181.18ms.

5 结论

本文以 JPEG2000 图像压缩标准中的二维离散小波变换的 VLSI 结构设计和实现为研究对象, 提出基于行缓存、并行阵列式流水线的设计方法实现二维离散小波变换的并行结

构. 此结构不仅适用于基于提升算法的 9/7 和 5/3 小波滤波器, 而且还对基于提升方法实现的二维离散小波滤波器具有适用性. 由于对结构和算法进行了优化, 采用了移位加操作, 因此, VLSI 实现时可以大大减少了硬件资源的开销, 增加了硬件资源的利用率, 几乎达到 100%, 加快了处理速度, 对于分解一级可实时处理. 如果芯片采样时钟频率是 100MHz, 用 9/7 小波 5 级小波分解, 可以每秒处理 24.4 帧大小为 2048 @ 2560 @ 8 比特的图像.

在进行小波变换的过程中即可从 LH, HL, HH 子带中读取码块, 进入 EBCOT 编码, 这样使得整个压缩处理过程中的闲置时间大大减少. 除此之外, 本设计具有强扩展性, 可以用于许多需要实时图像处理的领域, 如数码相机等.

参考文献:

- [1] ISO/IEC JTC 1/SC 29/WG 1 N1646R, 16, March 2000. JPEG2000 part 1 final committee draft version 1.0[S].
- [2] I Daubechies, W Sweldens. Factoring wavelet transforms into lifting schemes[J]. The J Fourier Analysis and Applications, 1998, 4: 247-269.
- [3] M Vishwanath, R M Owens, M J Irwin. VLSI architecture for the discrete wavelet transform[J]. IEEE trans On circuit and systems, 1995, 42(5): 305-316.
- [4] Chokri Souani, Mohamed Abid. VLSI design of 12D DWT architecture with parallel filters[J]. INTEGRATION, the VLSI journal 2000, (29): 181-207.
- [5] Kishore Andra, Chaitali Chakrabarti. A VLSI architecture for lifting2 based forward and inverse wavelet transform[J]. IEEE Trans On Signal Processing, 2002, 50(4): 966-977.
- [6] D Taubman. High performance scalable image compression with EBCOT[J]. IEEE Trans On Image Processing, 2000, 9(7): 1158-1170.

作者简介:



兰旭光 男, 1976 年 12 月生于山东海阳, 现为西安交通大学人工智能与机器人研究所的博士研究生, 主要研究方向为模式识别与智能系统, 小波变换, 视频压缩, 数字集成电路设计和信号处理.

郑南宁 男, 1952 年生于江苏南京, 中国工程院院士, 西安交通大学教授, 长期从事计算机视觉与模式识别、数字视频与信号处理等方面的研究工作, 发表学术论文 100 余篇.