

中图分类号:TP391.41

文献标识码:A

文章编号:1007-9416(2023)08-0150-04

DOI:10.19695/j.cnki.cn12-1369.2023.08.48

JPEG_LS 图像无损压缩算法的 FPGA 实现

中国电子科技集团公司第五十八研究所 严丹丹 刘勇

为满足海量图像数据在存储和传输过程中的高速无损要求,图像无损压缩算法 JPEG_LS 的 FPGA 实现成为人们研究的热点。针对现有的用 C 语言实现的图像无损压缩算法 JPEG-LS 进行优化,使其适合用硬件来实现。该算法由游程编码、常规编码和 Golomb 编码组成。针对该算法,提出了使用 Verilog HDL 硬件描述语言来进行 RTL 级设计,并在 Xilinx ISE 平台上实现了编码器。最后对多幅灰度图像进行了测试,测试结果表明其能够对灰度图像进行无损压缩,并提高了压缩速度。

互联网兴起以来,多媒体数字技术也在不断更新,图像数据和视频数据也在巨量增长,不同用户和系统间互相交换数字图像、视频数据等这类文件成为必须,对海量视频图像数据的存储和传输提出了严苛的要求^[1]。图像视频数据通过合理压缩以满足用户传输和存储过程中的要求,从而降低资源成本。本文提出的基于 FPGA 的图像无损压缩算法 JPEG-LS,能够很好地解决图像数据存储和传输过程中的高速无损压缩问题。

本文所提 JPEG-LS 是由国际电信联盟 ITU 于 1998 年 6 月发布的 ITU-T.87 标准和国际标准化组织 ISO 于 1999 年 12 月发布的 ISO-14495-1 标准,是面向连续色调静止图像的无损及近无损压缩标准^[2]。JPEG-LS 压缩标准以 LOCO-I (Low Complexity Lossless Compression for Images, LOCO-I) 算法为核心,包括常规编码、游程编码、Golomb 编码^[3]。该算法在图片的无损压缩领域具有高保真度及低复杂度的特点,易于硬件实现^[4,5],与现有的 JPEG、JPEG2000 等图像压缩算法相比,JPEG-LS 无损压缩算法在硬件实现方面具有明显的优势^[5]。

本文主要分为以下 3 个部分:第 1 部分主要是 JPEG-LS 算法的基本原理;第 2 部分主要是 JPEG_LS 算法的 FPGA 实现方案;第 3 部分主要是 JPEG_LS 算法的测试结果。

收稿日期:2023-04-13

作者简介:严丹丹(1986—),女,江苏兴化人,硕士研究生,工程师,从事算法研究及科研项目管理相关工作。

1 JPEG_LS 的算法分析

JPEG_LS 图像无损压缩算法涉及了常规编码,游程编码、Golomb 编码等。本文研究图像无损压缩,将 NEAR 取值 0。JPEG_LS 追求更加低复杂度的无损,核心是 LOCO-1 算法,该算法需要建立误差模型。当前图像像素及其相邻像素编码模式的选择由上下文相邻像素作用完成,基于上下文误差的编码也是算法核心。其方式为:当像素连续相等或差值小于阈值 NEAR 时,选择游程编码模式;当像素与上下文像素不相等或差值大于 NEAR 时,选择常规编码模式^[6]。在预测编码阶段完成后,计算出当前像素的预测误差。根据信息特征,JPEG-LS 对预测误差采用 Golomb 编码^[7]。其中,游程编码过程中,通过编码计数器统计,当提取的像素灰度较为平滑,则将相同灰度值的图像像素实施游程长度编码。

JPEG_LS 无损压缩算法的编码简化框图如图 1 所示。该算法的复杂度低,不需要进行 DCT 和算数编码。原始图像数据以设定的扫描模式依次输入系统编码器,主要方法是,当前像素编码之前,扫描以往的数据,以给出像素值的条件概率 P,对当前的像素值及状态进行判别,以确定是平摊区域还是丰富区域,此时,像素值的平均码长分布为 $-\log 2^P$ 。而近无损图像压缩则是将原

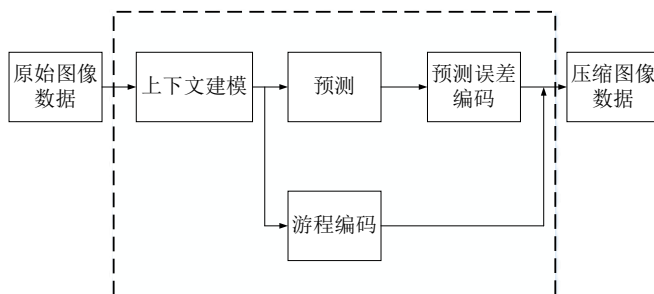


图 1 JPEG_LS 无损压缩编码简化框图

Fig.1 Simplified block diagram of JPEG_LS lossless compression coding

始值替换为重建值作为条件数据^[8]。在编码过程中, 图像将被大概率压缩。

1.1 上下文建模

通过当前待编码数据的附近与当前像素之间的相关性开展建模即为上下文建模, 是一个索引参数过程, 具有强依赖关系。例如, 若当前像素值 y , 其相邻的 4 个像素值记为 a 、 b 、 c 、 d , 以此 4 个像素样本进行寻址和更新, 得到三个梯度值, 即确认采用常规编码还是游程编码。待编码像素的上下文模型如图 2 所示。

| | | |
|-----|-----|-----|
| c | b | d |
| a | y | |

图 2 待编码像素的上下文模型

Fig.2 Context model of pixels to be encoded

其中, y 为当前采样像素点, a 、 b 、 c 、 d 为图中与 y 的位置关系所示的邻近点。 a 表示上一个处理的像素点, c 表示上一个处理像素点上方的像素点, b 表示当前处理像素点上方的像素点, d 表示下一个处理像素点上方的像素点。 a 、 b 、 c 、 d 所对应的像素重建值分别为 Ra 、 Rb 、 Rc 、 Rd 。

在对当前待编码像素周围的像素进行上下文建模后, 需要进行的操作主要有梯度值的计算、编码模式的选择以及当前像素的上下文参数 Q 的计算等。其中梯度值 Di ($i=1, 2, 3$) 的计算公式如式 (1)、式 (2)、式 (3) 所示:

$$D1=Rd-Rb \quad (1)$$

$$D2=Rb-Rc \quad (2)$$

$$D3=Rc-Ra \quad (3)$$

当梯度值 $D1$ 、 $D2$ 、 $D3$ 均小于等于 NEAR 或者全为 0 (对于无损压缩) 时, 即像素的灰度较为平滑时, 编码优先选择游程编码, 否则选择常规编码。

为了避免稀释上下文, 同时减少上下间数量, 对梯度值进行量化, 梯度值用 9 个区间进行区分, 将无限值的梯度值 D 变换为有限的 Qi ($i=1, 2, 3$), 便于运算和统计^[8]。JPEG_LS 算法规定使用 3 个非负常数阈值 $T1$ 、 $T2$ 、 $T3$ 来限定梯度值的量化范围, 使其量化为 $[-4, 4]$ 内的 9 个整数值, 这也就限制了后续合并梯度值所得的上下文参数的索引值 Q 的范围。

1.2 常规编码

常规编码部分主要包括 3 个部分: 获取当前像素预测值、计算预测误差、对预测误差的误差映射值进行 Golomb 编码^[9]。

当前像素值 x 的预测值记为 Px , 将由位于 a 、 b 、 c 的 3 个临近位置像素的重建值 Ra 、 Rb 、 Rc 得到。计算公式如式 (4) 所示:

$$P(x)=\begin{cases} \min(Ra,Rb)Rc \geq \max(Ra,Rb) \\ \max(Ra,Rb)Rc \leq \min(Ra,Rb) \\ Ra+Rb-Rc \quad \text{其他} \end{cases} \quad (4)$$

在得到预测值 Px 之后, 就要对其进行预测值的修正, 通过新的预测值 Px , 计算预测误差 Errval。假设 x 像素的实际值用 Ix 表示, 则 Errval= $Ix - Px$ 。最后, 对 Errval 的误差映射值 MErrval 进行 Golomb 编码。而要对 MErrval 进行编码, 首先就要求出 Golomb 编码的变量 k 。Golomb 编码的编码参数满足 $m=2^k$ 。 k 值的计算通过算法 $for(k=1; k++; N[Q]<k<A[Q])$ 得到, 其中 $A[Q]$ 表示上下文参数 Q 所对应的预测误差绝对值的和; $N[Q]$ 表示上下文参数 Q 所对应的像素出现的次数; Q 的取值区间为 $[0, 364]$ 。常规编码的编码过程如图 3 所示。

1.3 游程编码

游程编码是一种无损压缩编码的, 当信源出现连续的符号差异在允许失真的范围内, 或者是相同值的符号, 可以采用此编码方法^[9]。例如, 指定 a 样本作为游程编码的起始, 设置初始样本值为 Ra , 下一个扫描值记为 Ix , 若 $|Ix - Ra| \leq \text{NEAR}$, 则继续执行游程编码, 否则游程编码过程中断。

游程编码有两个重要部分: 游程长度编码和游程终止像素编码。游程长度编码的对象是游程长度, 记为 RunCnt, 它经上下文模型扫描及模式判断后由计数器获得, 记录了连续进入游程模式的像素点的个数。游程终止像素编码与常规编码类似。游程编码的编码过程如图 4 所示。

2 JPEG_LS 的 FPGA 实现

DSP、FPGA 和 ASIC 等均能促成 JPEG_LS 的硬件实现。综合考虑系统运行的速度、可靠性、灵活性、成

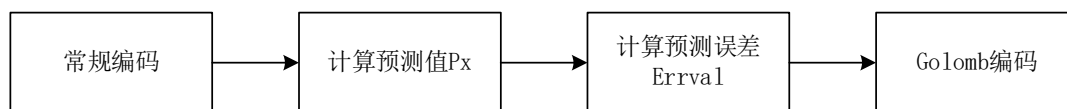


图 3 常规编码的编码流程图

Fig.3 Coding flow chart of conventional coding

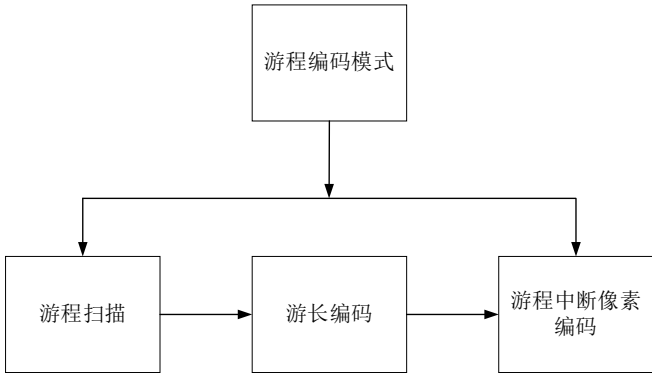


图 4 游程编码的编码流程图

Fig.4 Encoding flow chart of run coding

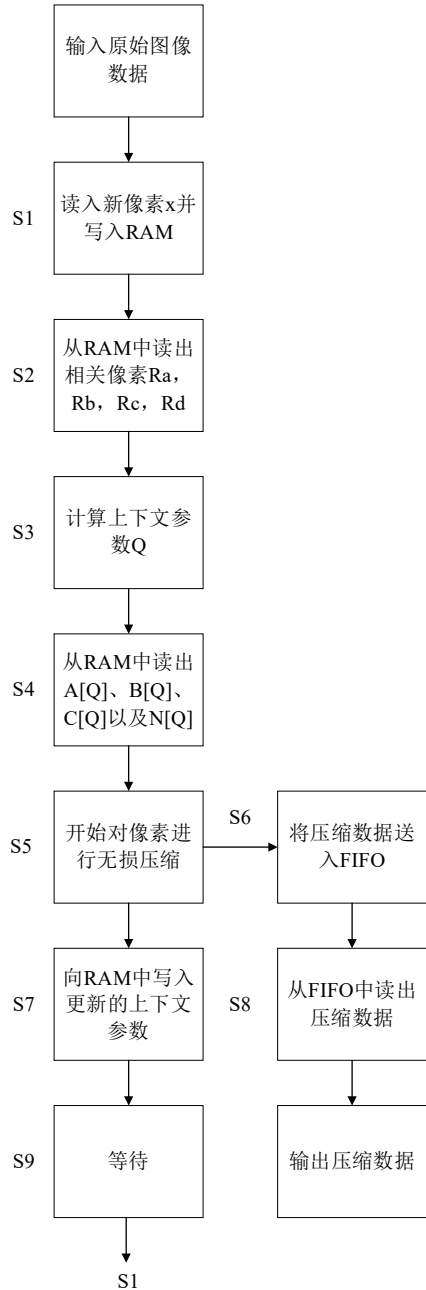


图 5 时序控制模块的状态转换图

Fig.5 State transition diagram of timing control module

本和知识产权等多种因素，本文以 FPGA 作为 JPEG-LS 算法的实现载体。基于 FPGA 实现 JPEG-LS 算法能够满足高质量图像的即时存储和快速传输，主要是通过海量图像数据地快速压缩来实现^[10,11]。

JPEG_LS 算法的硬件采用 Xilinx 公司的 FPGA 来实现，FPGA 开发平台是基于 Xilinx 公司高端 FPGA Virtex-6 系列中的 XC6VLX240T，它拥有 241,152 个逻辑单元，14,976kB 的内存等，足够容下整个 JPEG_LS 压缩算法硬件设计。开发软件采用 Xilinx 公司的 ISE。ISE 软件的特点是编译功能强大，表适合开发新器件和较大规模 FPGA 的开发，硬件描述语言则采用 Verilog HDL。

JPEG_LS 算法在基于 FPGA 的实现方案主要由两个模块组成：时序控制模块和压缩编码模块。时序控制模块的主要功能是分析时序问题，用来控制图像数据的输入输出。时序控制模块的状态共有 9 个状态，转换图如图 5 所示。首先从 RAM 中读取出所需像素值，然后计算出上下文参数的索引值 Q，再根据 Q 值从 RAM 中读出相应的上下文参数 A[Q]、B[Q]、C[Q] 以及 N[Q]，最后转入状态 5，启动压缩编码模块，完成像素压缩后，更新上下文参数并存入 RAM。压缩时所需要的像素及上下文数据可直接存放在 FPGA 自带的片内 RAM 中，因而无需片外的 SDRAM 实现存储需求^[12]。

压缩编码模块主要包括常规编码、游程编码以及 Golomb 编码等模块。JPEG_LS 编码器结构框图如图 6 所示。编码器编码的主要步骤如下：

- (1) 计算预测值 P_x ；
- (2) 计算预测误差 $Errval$ ；
- (3) 计算 Golomb 参数 k ；
- (4) 映射 $Errval$ 到 $MErrval$ ；
- (5) Golomb 编码；
- (6) 更新上下文相关参数值 A[Q]、B[Q]、C[Q] 以及 N[Q]。

3 实验结果

整个 JPEG_LS 图像无损压缩算法的设计使用 Verilog HDL 硬件描述语言编程，基于 ISE 平台上完成设计、综合、下载、验证等工作，其使用的 FPGA 资源约 5400 个逻辑单元。最后，使用 Modelsim 对几幅灰度图像进行了仿真。结果表明，压缩后的图像数据能够还原为原图像。在 40MHz 工作频率下，压缩一幅 256×256 的灰度图像大约耗时 81ms，可以保证应用中对其压缩速度的要求；压缩率约为 2:1，与使用软件压缩的压缩率相符合。因

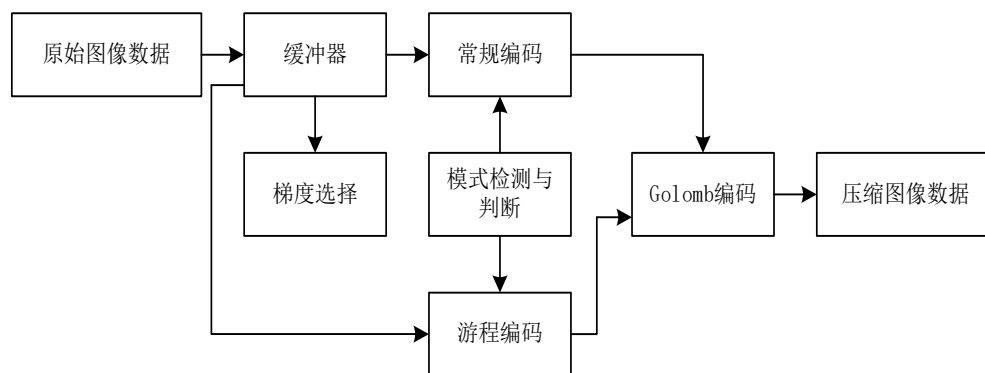


图 6 JPEG_LS 编码器结构框图

Fig.6 JPEG_LS encoder structure block diagram

此, 本文所设计的基于 FPGA 的 JPEG_LS 图像无损压缩算法能够满足海量图像数据在存储和传输过程中的高速无损要求, 为设计硬件实现的图像无损压缩提供了一个较好的解决方案。

引用

- [1] 景妮琴.基于DSP的JPEG图像压缩技术的探讨[J].电源技术应用,2012(9):110-111.
- [2] 王舒瑶.基于JPEG-LS标准的图像编码器硬件系统设计[D].西安:西安电子科技大学,2013.
- [3] Jtc1/Sc29/Wg1 I.Lossless and Near-lossless Coding of Continuous Tone Still Images(JPEG-LS)[J].FCD 14495,1997.
- [4] 范文晶,王召利,王惠娟,等.基于FPGA的无损图像压缩算法实现[J].电子科技,2016,29(11):126-128+132.
- [5] 王海荣.一种易于硬件实现的JPEG-LS无损图像压缩算法[J].科技视界,2014(17):22-24.
- [6] 曹健.一种无损JPEG压缩电路的设计[D].南京:东南大学,2016.
- [7] DING J J,CHEN H H,WEI W Y.Adaptive Golomb Code for

Joint Geometrically Distributed Data and Its Application in Image Coding[J].IEEE Transactions on Circuits & Systems for Video Technology,2013,23(4):661-670.

[8] MERT Y M.Low Complexity Scheme with JPEG-LS for Near-Lossless,Multicomponent and Selective Compression[C]//Conference on Real-time Image and Video Processing,2017:1-11.

[9] 唐焱,曹剑中,李变侠,等.基于FPGA的JPEG-LS无损压缩算法的实现[J].弹箭与制导学报,2006(SA):1219-1222.

[10] NAZAR F,MURUGAN S.Implementation of JPEG-LS Compression Algorithm for Real Time Applications[C]//International Conference on Electrical,Electronics,and Optimization Techniques. IEEE,2016:2772-2774.

[11] DENG L H,HUANG Z H.The FPGA Design of JPEG-LS Image Lossless Decompression IP Core[C]//Chinese Automation Congress.IEEE,2016:2199-2203.

[12] 张天序,邹胜,曾永慧.基于FPGA的图像无损压缩算法的实现[J].系统工程与电子技术,2004(10):1340-1343.