

# 矢量量化中码书旋转压缩的研究

王冬芳, 廖裕民, 余宁梅

WANG Dong-fang, LIAO Yu-min, YU Ning-mei

西安理工大学 电子工程系, 西安 710048

Department of Electronics Engineering, Xi'an University of Technology, Xi'an 710048, China

E-mail: wangdong@xaut.edu.cn

**WANG Dong-fang, LIAO Yu-min, YU Ning-mei. Research on compressed codebook by rotating codewords in vector quantization algorithm. Computer Engineering and Applications, 2008, 44(24): 53-55.**

**Abstract:** A lot of correlations exist in original codebook by rotating each codeword to different directions, so a theory that an original codebook can be compressed by rotating codewords to different directions in Vector Quantization (VQ) algorithm is proposed. In this theory, all codewords in codebook are rotated to four hierarchical directions to check their correlations. If correlations between two codewords with four directions exist, one codeword will be deleted from this codebook. So in compressed codebook, each codeword has four directions, it can be instead of four codewords. While encoding, the compressed codebook will be recovered by rotating codewords. An algorithm how to compress codebook by rotating codewords is also proposed and verified in this paper, a codebook can be compressed from 1 024×16×8 bit to 256×16×8 bit using this algorithm. Simulation results show that this codebook can decrease 75% memory room and I/O bandwidth but only lost 0.28 dB PSNR compared with the original codebook.

**Key words:** vector quantization; codebook; rotating compression

**摘要:**普通码书中的码字之间在不同的方向上具有很大的相关性,存在大量的数据冗余。提出了将码书中的码字旋转压缩的理论。该理论是将各个码字按四个方向垂直旋转后进行相似性检查。如果旋转后的码字其中一个方向上与前面的码字存在相似,则将该码字删除,从而达到压缩的目的。编码时将压缩后的码书旋转恢复后进行编码,从而大幅降低了需要存储的码字数量。同时给出了一种将现有 1 024 阶 16 维码书旋转压缩成 256 阶 16 维的方法,并对该方法得到的码书性能进行了仿真验证。实验结果表明使用压缩后的码书在硬件实现时与普通的矢量量化码书相比减少了 75% 的存储空间和输入带宽,而 PSNR 平均只降低 0.28 dB。

**关键词:** 矢量量化; 码书; 旋转压缩

**DOI:** 10.3778/j.issn.1002-8331.2008.24.014 **文章编号:** 1002-8331(2008)24-0053-03 **文献标识码:** A **中图分类号:** TP391

## 1 引言

矢量量化(Vector Quantization, VQ)<sup>[1]</sup>是 20 世纪 70 年代后期发展起来的一种数据压缩技术,其压缩率较高,编、解码的算法简单,易于硬件实现,因此在数字视频与音频压缩以及图像压缩方面得到广泛应用。

矢量量化有两大关键技术:高性能的码书和快速的搜索算法。其中码书起着举足轻重的作用,好的码书在解码时能够恢复失真度小的图像。如何生成性能优越的码书一直是图像压缩领域的研究热点。对矢量量化算法中码书的处理主要从时域和频域两大方面展开,时域方面有许多经典的算法,如 LBG<sup>[2]</sup>算法;频域方面主要用离散余弦变换以及离散傅立叶变换来分析频域的特征,找到共性。基于此,国内外专家学者提出了各种改进算法通过提高码书的质量来提高 VQ 的性能,如结构树矢量量化<sup>[3]</sup>和格型矢量量化<sup>[4]</sup>等。

目前,为提高图像传输的速度,各种图像压缩算法纷纷用硬件实现<sup>[5-7]</sup>。但是,在用硬件实现矢量量化算法时,码书的存储会耗用大量的硬件面积。例如,在一次芯片设计中,采用 256 阶 16 维的灰度码书所用的存储器面积甚至占到芯片面积的 1/3。因此如何对码书进行压缩也是迫切需要解决的问题。

本文提出将现有优秀码书中的码字进行旋转压缩的方法,希望能充分挖掘码字间方向上的相关性,在原始码书基础上生成旋转压缩码书,这样新码书可以大幅缩小码字数量。同时使得旋转压缩码书基本保持原来码书的编码效果。本文还给出了一种将现有码书旋转压缩的方法,并对该方法得到的码书进行了验证。

## 2 矢量量化原理和码书分析

矢量量化可被看成是从  $k$  维 Euclidean 空间  $R^k$  到  $R^k$  的有

**基金项目:**陕西省自然科学基金研究计划项目(No.2006F 29);应用材料创新基金资助项目(No.XA-AM-200714);西安理工大学优秀博士学位论文研究基金。

**作者简介:**王冬芳(1976-),女,博士研究生,讲师,研究方向为图像处理芯片设计和超大规模集成电路设计;廖裕民(1982-),男,硕士研究生,主要研究方向为视频压缩专用集成电路设计;余宁梅(1963-),女,工学博士,教授,主要研究方向为专用集成电路设计及工艺。

**收稿日期:** 2007-10-22 **修回日期:** 2008-01-16

限于集  $C$  的一个映射,即  $Q: \rightarrow C$ , 其中,  $C = \{y_0, y_1, \dots, y_{N-1} | y_i \in R^k\}$  称为码书,  $N$  为码书大小,  $y_i$  称为码字, 下标  $i$  是该码字在码书中对应位置的地址索引。该映射满足:  $Q(x | x \in R^k) = y_p$ , 其中  $x = (x_0, x_1, \dots, x_{k-1})$ ,  $y_p = (y_{p_0}, y_{p_1}, \dots, y_{p_{k-1}})$ , 并满足

$$d(x, y_p) = \min_{a \leq j \leq N-1} (d(x, y_j)) \quad (1)$$

其中,  $y_p$  的下标  $p$  代表最佳匹配码字在码书中的地址索引, 而  $d(x, y_j)$  为矢量  $x$  与码字  $y_j$  之间的失真测度, 常用的失真测度为平方误差测度, 其表达式为:

$$d(x, y_j) = \sum_{l=0}^{k-1} (x_l - y_{j,l})^2 \quad (2)$$

VQ 包括编码器和解码器两个主要部分。用矢量量化技术对一幅图像编码时, 首先要将原始图像按  $n \times n$  个像素大小为单位进行小块划分, 每个小块组成一个  $k$  维 ( $k = n \times n$ ) 的矢量, 含  $k$  个像素; 编码时, 首先通过在码书  $C$  中为每个输入图像矢量  $x$  搜索最匹配的码字  $y_p$ , 使其失真测度值最小; 然后将最匹配码字的索引  $p$  作为编码值代替矢量  $x$  进行存储或传输。解码时, 则先根据接收到的索引  $p$  从相同码书中找出对应的码字, 然后用这些码字重构图像。

根据香农的率失真理论<sup>[8]</sup>, 码书尺寸的下限与图像的失真度成反比。当码书尺寸足够大时, 矢量量化性能有可能任意接近率失真限; 反之, 码书尺寸太小时, 图像的失真度必然超出可接受的范围。因此码书的尺寸也要控制在一定范围内, 常用的码书尺寸有 256、512、1 024 和 2 048 等。理论上, 使用的码书尺寸越大所获得的压缩图像质量越高, 读取码书的输入带宽越大编码的速度也就越快, 但在实际应用特别是硬件实现中, 码书尺寸越大所使用的存储空间也越大, 输入带宽越大所占用的 I/O 口就越多, 而用于存放码书的存储器和输入输出端口的面积又在整个系统电路的使用面积中占有很大比例, 因而研究如何减少码书存储数量和输入带宽并且同时保持图像压缩质量的方法是十分有意义的。而本文提出的旋转压缩码书充分利用了原始码书中码字方向上的相关性, 将原始码书压缩存储, 可以大幅降低硬件实现过程中码书的存储面积和输入带宽, 并设计了相应的实现方法和基于得到的旋转码书的快速码字搜索算法。

### 3 码书旋转压缩的原理与生成方法

在图 1 中, 可以看到 4 个不同的宏块, 但是这 4 个宏块在方向上的相关性是很大的, 如果将第一个宏块按顺时针方向旋转 3 次, 就可以发现每次旋转的结果和另外的 3 个宏块是几乎一样的。所以, 在实际的图像块中, 像素的排列往往具有一定的方向性, 而对于通用码书中的所有码字来说, 同样存在与图 1 相似的方向性。

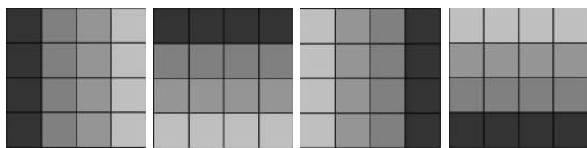


图 1 宏块方向相关性

所以, 为了利用这种码书内的方向上的相关性, 首先利用小波变换, 生成有规律的符合人眼视觉特征的 1 024 尺寸、矢量维数为 16 的原始码书<sup>[9]</sup>。

得到原始码书后, 如果将码书中的每一个码字  $X_i$  旋转, 如图 2。利用码字之间方向的相关性, 将  $X_i$  码字本身和旋转后的

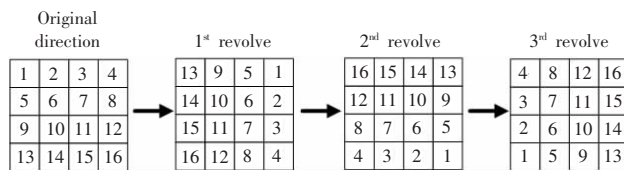


图 2 码书旋转原理

3 个矢量  $\hat{X}_i$  分别与码书中其他码字  $X_j$  进行比较。当

$$T < \sum_{m=1}^{16} |\hat{X}_i(m) - X_j(m)|, \text{ 其中 } T \text{ 为阈值}$$

算法就认为  $X_j$  码字是由  $X_i$  码字旋转而成, 并且在存储时只存放  $X_i$  的值。因为阈值  $T$  决定了可以压缩存放的码字的个数, 所以  $T$  的大小直接影响最后旋转压缩码书的尺寸。

为了最终获得尺寸合适的压缩码书, 在 Matlab 平台下, 分别设置阈值  $T=20, 21, \dots, 50$ , 得到原始 1 024 尺寸码书的压缩后尺寸如图 3 所示。图中横轴为阈值  $T$ , 纵轴为压缩后码书尺寸。由图可见随着阈值  $T$  的增大压缩后码书尺寸不断减小。

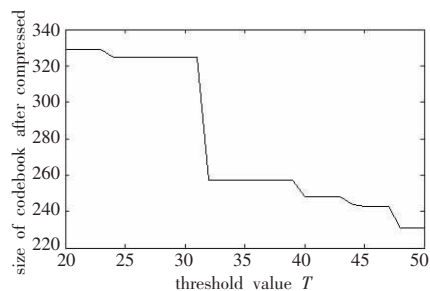


图 3 不同阈值  $T$  对应的码书压缩后尺寸

为了进一步确定压缩后的码书尺寸对图像压缩效果的影响, 使用阈值  $T=20, 21, \dots, 50$  时对应的范围由 329 到 230 尺寸的压缩后码书对 512x512 尺寸、256 灰度级的标准静止灰白图像 lena 进行全搜索匹配, 得到的匹配结果如图 4。图 4 中横轴为阈值  $T$ , 纵轴为峰值信噪比 (PSNR)。从图 4 中可以看到当  $T$  从 20 到 50 时, 使用对应尺寸旋转压缩码书对 lena 进行编码, 重构图像峰值信噪比仅从 28.35 dB 变化到 28.30 dB, 由此可见在图示阈值范围内旋转压缩码书尺寸对编码效果影响不大。而当原始码书被压缩为 256 尺寸时, 在编码时就可以很容易地由压缩后的 256 尺寸码书旋转 4 个方向恢复成 1 024 尺寸码书。所以当阈值  $T$  为 40 时, 1 024 尺寸的原始码书可以被压缩为 256 尺寸。

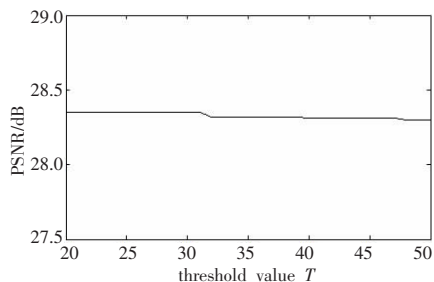


图 4 不同阈值  $T$  对应的编码重构效果

### 4 使用旋转压缩码书的快速搜索算法

为了配合旋转压缩码书的使用和等和值块扩展矢量量化码字搜索算法<sup>[10]</sup>, 本文设计了基于旋转压缩码书的搜索算法,

具体流程如下:

(1) 开始搜索前, 计算各旋转压缩码字的和值  $S_i, i=1, 2, \dots, 256$  然后按码字和值的大小对旋转压缩码书进行升序排列。

(2) 将码书分成  $L$  块, 并把每一块中间或中间附近的码字的和值作为本块码书的特征和值。

(3) 宏块矢量开始输入, 搜索开始, 搜索流程如图 5。首先, 计算输入矢量  $X$  的和值  $S_x$ 。因为特征和值码书是升序排列的, 所以可以通过二分比较查找法来找到与和值  $S_x$  最相近的码书块的特征和值  $S_p$  和其所对应的码书块  $p$ 。

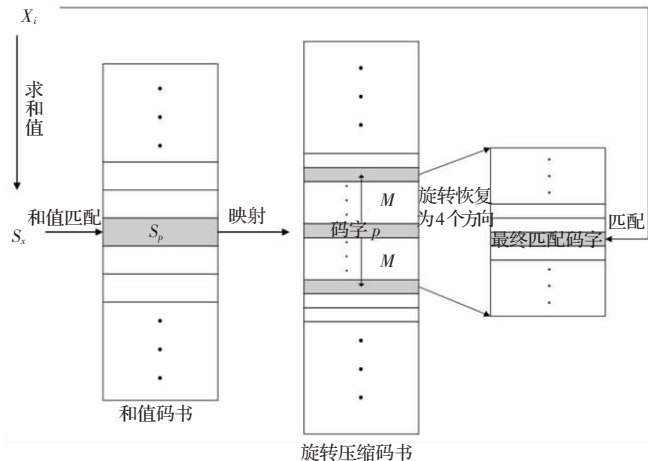


图 5 算法匹配搜索流程

(4) 将码书块  $p$  的中间码字作为初始匹配码字, 该码字附近同时上下扩展搜索  $M$  个相邻码字, 搜索范围内的码字旋转恢复出 4 个方向的码字, 在恢复码字中找到与输入矢量  $X$  最优匹配码字  $j$  和旋转次数  $i$ 。

(5) 将码字索引  $j$  和旋转次数  $i$  输出。

## 5 参数选定

码书的分块和上下扩展搜索范围的选取直接影响算法的性能和实现复杂度。所以, 为了确定最终选用的码书的分块和上下扩展搜索范围, 做了以下的实验。在 Matlab 平台下, 取码书块数  $L=128, 64, 32$  和搜索范围  $M=2, 3, 4, \dots, 10$ , 对  $512 \times 512$  尺寸、256 灰度级标准静止灰白图像 lena、couple、man、splash、lax 进行矢量量化压缩, 然后对得到的每幅重建图像效果取平均值, 结果如图 6。图 6 中横轴为搜索范围  $M$ , 纵轴为峰值信噪比 PSNR。

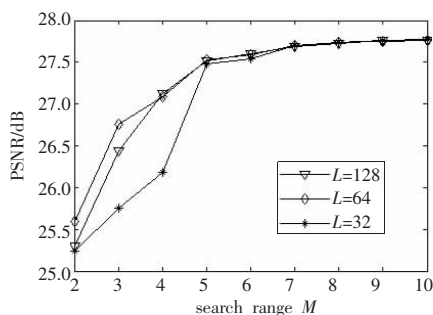


图 6 不同参数的压缩效果

从图 6 中可以看出, 当搜索范围  $M$  由 4 到 5 时 PSNR 跳变最大, 而且当搜索范围为 5 时, 码书块  $L=32$  与  $L=64, L=128$

的 PSNR 相差不大。经过权衡性能和硬件实现的复杂度, 本文决定采用 256 尺寸旋转压缩码书, 码书块  $L=32$ , 上下搜索范围  $M=5$  的旋转压缩码书算法。

## 6 实验结果与分析

在算法和算法中需要的所有参数值确定以后, 将使用本文算法对  $512 \times 512$  尺寸、256 灰度级标准静止灰白图像 lena、couple、splash、lax 编码后的峰值信噪比与使用 256、1 024 尺寸码书的全搜索算法的峰值信噪比进行了对比, 三种算法对比的结果见表 1。

表 1 算法性能对比

| Image name | Adopted_algorithm ( $L=32, M=5$ ) |       | 256cb_ full_search |
|------------|-----------------------------------|-------|--------------------|
|            | 1 024cb_ full_search              | dB    |                    |
| splash     | 29.97                             | 29.77 | 27.72              |
| couple     | 26.79                             | 26.49 | 25.32              |
| lax        | 23.78                             | 23.71 | 22.86              |
| lena       | 30.11                             | 29.62 | 28.31              |

根据实验结果可以看出, 本文算法比 256 尺寸码书全搜索算法性能平均提高 1.33 dB, 比 1 024 尺寸码书全搜索算法性能平均下降 0.28 dB。图 7 是原始图像与本文算法压缩重建后的图像对比。



(a) 原图

(b) 解码图

图 7 原始图像与压缩图像比较

下面从匹配计算量、码书存储量和输入码书带宽三个方面对实验结果进行分析:

(1) 对于典型的矢量量化全搜索算法, 当码书尺寸为 1 024 时, 需要对输入矢量和码书进行 1 024 次匹配计算。而本文算法只需要进行 6 次和值匹配运算和  $4 \times (2M+1)$  次码书匹配运算, 当  $M=5$  时, 匹配运算数为  $6+44=50$  次, 为典型算法的 4.88%。

(2) 典型的矢量量化算法需要存储全部的码书, 如果码书尺寸为 1 024, 则需要存储 1 024 个码字。本文提出的算法是各个码书旋转后进行压缩存储, 当阈值  $T$  为 40 时, 1 024 尺寸的原始码书被压缩为 256 尺寸, 只是原来典型矢量量化码书存储量的 25%。

(3) 本文算法在搜索匹配过程中是由搜索范围内的每一个码字都旋转恢复成四个码字, 所以每读入一个码字就相当于典型矢量量化中同时输入四个码字, 相对典型矢量量化输入码书带宽减少了 75%。

## 7 结论

为降低硬件设计时矢量量化码书的存储容量, 本文利用码书中码字之间方向上的相关性, 提出了将码书中的码字旋转压缩的理论。并根据这一理论提出了一种将现有码书旋转压缩的

(下转 58 页)