

基于旋转 Barnes-Wall 格的增益-波形 矢量量化器及其在序列图象 编码中的应用*

薛向阳 陈学青 樊昌信

(西安电子科技大学 信息科学研究所 西安 710071)

摘要 本文首先提出了一种基于旋转 Barnes-Wall 格的格型矢量量化器(LVQ)的构造方法及快速量化算法,然后研究了以此 LVQ 为核心的增益-波形矢量量化器(GSLVQ)的实现方法,最后探讨了 GSLVQ 在序列图象编码中的应用方案,并给出了较好的实验结果。

关键词 图象编码,矢量量化,格

1 引 言

统计型矢量量化器^[1](例如用 LBG 算法设计的矢量量化器 LBGVQ)具有很强的压缩能力,因而在数据压缩领域得到了广泛的应用。然而这一类矢量量化器编码复杂度非常高(与编码速率成指数关系),用聚类算法对训练序列进行训练而得到的码书通常不是全局最佳的,对不平稳信源(比如千变万化的图象信源)很难找到一个普遍适用的码书等,这些缺陷在很大程度上限制了统计型矢量量化在图象编码中的应用。

不少学者^[2~7]致力于另一种类型的矢量量化研究,文中称之为格型矢量量化(LVQ),它建立在格理论基础之上。由于格的对称几何特性和优良的代数特性,这使得 LVQ 具有许多特别的优点:构造码书非常简单,允许使用巨型码书且不必存储码书、具有快速量化算法等。但是, LVQ 主要适用于均匀分布的信源,对其它如高斯分布或拉普拉斯分布信源其量化性能往往不是最佳的,与统计型矢量量化相比其缺陷是压缩能力相对较低。尽管如此, LVQ 仍在图象编码中得到成功的应用^[3,4]。

LVQ 的研究主要在于寻求高性能的 LVQ 和快速量化算法。格型矢量量化器的性能一般由其维数、格本身的特性和由格点组成的码书的空间形状决定。16 维 Barnes-Wall 格 A_{16} 是 16 维空间的致密格,其旋转格具有与 A_{16} 相同的编码增益^[2,5~7]和特别好

1993-12-15 收到,1994-05-20 定稿

* 国家 863 计划 863-317 课题资助,校基金资助,综合业务网理论及关键技术国家重点实验室部分资助

薛向阳 男,1968 年生,博士生,现在复旦大学计算机科学博士后流动站从事图象编码研究。

陈学青 女,1967 年生,硕士,现从事数字通信和图象编码等研究。

樊昌信 男,1931 年生,教授,博士生导师,现从事语音和图象编码等研究。

的代数特性,因此本文将基于它的旋转格来实现格型矢量量化器,同时提出了一个简单有效的码书形成方案(即如何从无限多的格点中选取有限的格点来构成码书),接着重点研究了它的快速量化算法,然后以此 LVQ 为核心设计增益-波形矢量量化器(GSLVQ),最后将 GSLVQ 用于序列图象编码中。

2 矢量量化和格的基本概念

这里先给出一些基本概念,主要包括矢量量化(VQ)、格及基于格的 VQ 等,更详细的内容请参阅文献[1,2,4,6—8]。

2.1 矢量量化

令 $C = \{y_1, y_2, \dots, y_m\}$ 由 M 个码矢 y_i , $1 \leq i \leq M$ 组成的码书, C 的大小 $|C| = M$ 。令 Q 是对 N -维空间的全部划分,共划分为 M 个量化胞腔,使得每个胞腔对应一个唯一的码矢量 y_i ,最后令 Φ 是从码矢 y_i 到序号 i 的映射。那么码书 C 、划分 Q 和映射 Φ 一起构成了 VQ。

2.2 格 (Lattice)

从代数上讲,一个 N -维格 Λ 是矢量集合,其在 N -维空间中普通加法下构成群。从几何上讲,格是一个均匀覆盖 N -维空间的无限规则的矢量阵列,最简单的 N -维格是整数格 Z^N ,它由所有具有整数坐标的 N -维矢量构成。一般地,格 Λ 可表示成:

$$\Lambda = \{x | x = c_1 a_1 + c_2 a_2 + \dots + c_n a_n\},$$

其中 a_i 是格的基矢量, c_i 是整数。格是基矢量的整数线性组合,格的大小是无限的。一些著名的格有 E_8 格、Barnes-Wall 格和 Leech 格等。

2.3 基于格的矢量量化

VQ 的码书 C 有多种生成或构造方法。如果用聚类算法训练码书,则本文称之为统计型 VQ,例如基于 LBG 算法的 VQ^[1]。如果从某个格中选取有限个数的格点作为码书,则称之为 LVQ。因为格点的分布是均匀的,所以 LVQ 通常适合于无记忆均匀分布的信源。但是通过合理选取格点,可使 LVQ 适合于其它类型的源,比如对高斯分布的信源来说,应选取超球面之内的所有格点。一些研究者常将 LVQ 用作为波形量化器^[4]以推广 LVQ 的应用范围。

3 基于旋转 Barnes-Wall 格的格型矢量量化器的构造

有许多格可用来构造性能优良的 LVQ,比如 E_8 、 A_{16} 、24 维 Leech 格(A_{24})等^[2-4]。本文选用 A_{16} 的旋转格 RA_{16} 。 A_{16} 最早由 Barnes 和 Wall 提出,它有很多构造方法,本文用二次结构(squaring construction)^[6,7],即

$$A_{16} = 4Z^{16} + 2RM(4,3) + RM(4,1) = 4Z^{16} + 2(16,15,2) + (16,5,8), \quad (1)$$

其中 $RM(r, n+1)$ 为 Reed-Muller 码, Z^{16} 是 16 维整数格。

由(1)式可见, A_{16} 是模 4 二进制格,且为 $4Z^{16}$ 格的 2^{20} 个陪集之并集,这意味着寻找最近格点(即量化过程)时需要巨大的存储量并搜索 2^{20} 个陪集,其存储和运算复杂度很

大。为此,我们将 A_{16} 进行旋转得到旋转格 RA_{16} 。

$$RA_{16} = 4Z^{16} + 2RM(4, 2) + RM(4, 0) = 4Z^{16} + 2(16, 11, 4) + (16, 1, 16). \quad (2)$$

由 (2) 式可见, RA_{16} 是 $4Z^{16}$ 格的 2^{12} 个陪集之并集, 陪集数目减少为 $1/2^8$ 。显然基于 RA_{16} 格的量化过程要比 A_{16} 简单得多。

图 1 给出 RA_{16} 格的格点形成过程(实际上对应 VQ 的译码过程, 即已知序号(index) (用 $N + 12$ bit 表示的一个整数) 而得到重建码矢量。在 LBGVQ 中译码仅仅是一个查表过程, 但在 LVQ 中译码过程则是从 index 到相应格点的映射过程)。

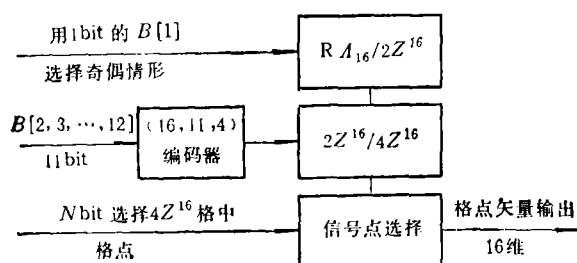


图 1 RA_{16} 格的格点形成过程

图 1 中, 分组码 (16, 11, 4) 的生成矩阵为 G_1 , 对 G_1 进行线性变换后为 G_2 。显然 G_1 和 G_2 将张成同样的码空间, 所以可用 G_2 来形成 RA_{16} 格的格点。

$$G_1 = \begin{bmatrix} 1111 & 0000 & 0000 & 0000 \\ 1100 & 1100 & 0000 & 0000 \\ 1010 & 1010 & 0000 & 0000 \\ 1111 & 1111 & 0000 & 0000 \\ 1100 & 0000 & 1100 & 0000 \\ 1010 & 0000 & 1010 & 0000 \\ 1111 & 0000 & 1111 & 0000 \\ 1000 & 1000 & 1000 & 1000 \\ 1100 & 1100 & 1100 & 1100 \\ 1010 & 1010 & 1010 & 1010 \\ 1111 & 1111 & 1111 & 1111 \end{bmatrix}, \quad G_2 = \begin{bmatrix} 1111 & 0000 & 0000 & 0000 \\ 0000 & 1111 & 0000 & 0000 \\ 0000 & 0000 & 1111 & 0000 \\ 0000 & 0000 & 0000 & 1111 \\ 1100 & 1100 & 0000 & 0000 \\ 1010 & 1010 & 0000 & 0000 \\ 1100 & 0000 & 1100 & 0000 \\ 1010 & 0000 & 1010 & 0000 \\ 1000 & 1000 & 1000 & 1000 \\ 1100 & 1100 & 1100 & 1100 \\ 1010 & 1010 & 1010 & 1010 \end{bmatrix}.$$

假设矢量 $B = (1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0)$, 这里用 $B[1]$ 表示 B 中第一个坐标 1, 它决定格点坐标的奇偶情形, $B[2 \cdots 12]$ 表示 B 中另 11 个坐标 (1 0 0 0 0 0 1 0 0 1 0), 用于选择陪集, 再假设 N bit 所选择的 $4Z^{16}$ 格点 Z 为 (0 0 0 0 0 0 0 0 0 0 0 0 -4 0 0 4), 则 $P[1 \cdots 16] = B[2 \cdots 12] \times G_1 = (1010 \ 1010 \ 0101 \ 1010)$, 于是由 (2) 式得格点 $L = Z + 2 \cdot ((-1)^{P[1]}, \dots, (-1)^{P[16]}) + (-1)^{B[1]} \cdot (1111111111111111) = (-3 \ 1 \ -3 \ 1 \ -3 \ 1 \ -3 \ 1 \ 1 \ -3 \ 1 \ -3 \ -7 \ 1 \ -3 \ 5)$ 。

从前面的构造可以看出, 用 $(12 + N)$ bit 就可以表示 RA_{16} 格中的任意格点。从 VQ 的角度来看, 每个格点矢量都可视为码书中的码矢量, 而相应的 $(12 + N)$ bit 就可作为相应格点的序号。

RA_{16} 格中的格点实际上是分布在以 $r(m) = 2\sqrt{2m}$ 为半径的一系列同心超球面上, 每个球面又称为壳(Shell), 则第 m 个壳的半径即为 $r(m)$ 。随着 m 的增大, 壳上的格点数目急剧增加, 例如 RA_{16} 格的壳 0, 1, 2, 3, 4, 5 上分别有 1, 0, 4320, 61440, 522720, 2211840 个格点。在实际使用时 LVQ 只能选取有限数目的格点构造码书。已有的码书形成方法是将某个壳以内的所有格点作为码书^[3,4]。如果选择壳 2 和 3 中所有格点作为码书, 则码书中共有 65760 个格点, 用等长码表示每个格点的序号时需要 17bit。如果再把壳 4 中的格点加入到码书中, 则共有 588480 个格点, 此时需要 20bit 才能表示码书中每个格点。显然如此之多的格点对提高 LVQ 的压缩比是非常不利的。

这里提出一种简单有效的码书形成方案, 即选 $4Z^{16}$ 格点为原点(即全零矢量)的那些 RA_{16} 格点。从 (2) 式可以看出这样的格点共有 2^{12} 个。显然这种选择方案不仅减小了码书的大小(即提高了压缩比), 而且又能够利用格的特性实现快速量化。不过因大大减少了码书中格点数目, 量化器的性能将有所下降, 表 1 给出两种格矢量量化器的性能比较, 从表 1 可看出, 当比特数目变为前者的 70.58% 时, 其平均信噪比为前者的 75.94%, 即信噪比的下降程度要小些, 从这个意义上讲我们提出的码书形成方法要优于通常的选取方法。

表 1

码书形成方法	码书大小	信源类型	平均信噪比	比特数
以 2 和 3 壳内的所有格点作为码书	65760	独立高斯分布信源 (均值为零, 方差为 1)	5.03 dB (10000 个测试矢量)	17 bit/点
取 $4Z^{16}$ 格点为全零矢量的 RA_{16} 格点作为码书	4096	作为测试信源	3.82 dB (同样的测试矢量源)	12 bit/点
比例关系	6.22%	/	75.94%	70.58%

4 快速格型矢量量化算法

假定 16 维欧氏空间 R^{16} 中一个实矢量 $X = (x_1, x_2, \dots, x_{16})$, 那么对矢量 X 的量化问题就是决定码书中(注意并不是指 RA_{16} 格中所有格点)离 Y 最近的格点。为减少运算量, 我们将 $RM(4, 2)$ 码的生成矩阵 G_1 进行线性变换得到 G_2 , 具体的量化过程如下:

(1) 首先根据输入矢量 X 计算一个欧氏距离测度表 $M_{n,k}$, 其中 $n = 1, 2, \dots, 16$, $k = -3, -1, 1, 3$, 见表 2。

(2) 确定最佳陪集模式和奇偶格点情形, 从而得到 $B[2 \dots 12]$ 和 $B[1]$ 。由生成矩阵 G_2 可见, 若将 G_2 的每 4 列划分为一组, 则在任一组的四坐标中只有 16 个不同的子模式, 并且构成 8 个互补对。

首先对每组的每个子模式计算距离度量 $GM_{g,i}, GM_{g,i}$ 为 4 个 M 值之和, 其中 g 指明组号, i 指明子模式。 M 值的选取方法: 当子模式中相应位为 0 时选用 $k = 3$ (对偶坐标

表 2 距离测度表 $M_{n,k}$

k	$M_{n,k}$
-3	$(X_n + 3) * (X_n + 3)$
-1	$(X_n + 1) * (X_n + 1)$
+1	$(X_n - 1) * (X_n - 1)$
+3	$(X_n - 3) * (X_n - 3)$

注: 表中 X_n 为矢量 X 的第 n 个坐标

情况)时的 M 值(对奇坐标情况, 取 $k = 1$); 当子模式中相应位为 1 时选用 $k = -1$ (对偶情况)时的 M 值(对奇情况 $k = -3$).

再由生成矩阵 G_2 的下 7 行形成 128 个模式, 对其中每个模式计算距离度量 SM , SM 为 4 个 GM 之和, 从而得到 256 个 SM 值(奇偶各 128 个), 选择最小的 SM , 与其对应的即是最佳的陪集模式(用来决定 $B[2 \cdots 12]$ 的取值)和奇偶坐标情形 $B[1]$.

(3)由(2)式的格构造公式就可以得到最近的格点, 从而完成了 VQ 的量化过程, 同时也得到了格点的标号 B .

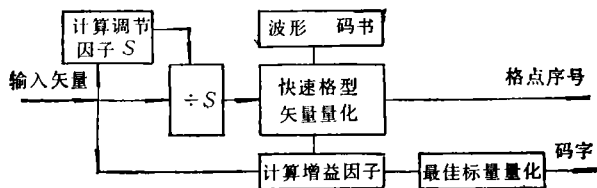
我们可以看出, 快速量化算法充分利用了格的优良代数特性, 避免了对整个码书空间进行全搜索(在 $LBGVQ$ 中常常使用这种耗时的全搜索量化方法), 从而大大地减少了运算复杂度, 整个量化算法仅需要 1599 次二元运算, 具体比较见表 3.

表 3

算法类型	加	乘	比较	二元运算总次数
全搜索算法	126976	65536	4095	196607
本文算法	1280	64	255	1599
复杂度之比	99.2:1	1024:1	16.05:1	122.95:1

5 基于 LVQ 的增益-波形矢量量化器

由于我们仅选择 RA_{16} 格中极少数目的格点构造码书, 所以此矢量量化器不能代表 R^{16} 中任意实矢量, 解决的办法是将它用作波形量化器, 见图 2.

图 2 基于 LVQ 的增益-波形矢量量化

从图 2 可见, 输入矢量首先需要除一个标量调节因子 S , 从而将被量化矢量压缩到码书所规定的空间之内, 以保证能在码书所规定的格点中找到一个与之对应的最近格点; 然

后将得到的格点与输入矢量作点积相关运算,从而得到增益因子 G ; 最后再对 G 进行量化编码。标量因子 S 的选取直接影响量化器的性能, 这里给出一个确定最佳 S 和实现 GSLVQ 的算法:

- (1) 从被量化矢量 $X = (x_1, x_2, \dots, x_{16})$ 中寻找绝对值最大的坐标, 即

$$S^{\max} = \max\{\text{abs}(x_i)\}, i = 1, 2, \dots, 16, \text{abs}(\cdot)$$
 为取绝对值运算。
- (2) 令 $t = 0, 1, 2, \dots, 11$ 得到相应的 S 值为:

$$S' = 1 + 0.5t, S = S^{\max}/S'.$$
- (3) 用 S 去调节输入矢量 X , 得到 $Y = X/S$ 。
- (4) 用上述 LVQ 量化 Y , 得到相应格点 Y^Q 和标号 $B[1 \dots 12]$ 。
- (5) 计算最佳增益因子: $G = (X \cdot Y^Q)/(Y^Q \cdot Y^Q)$, “ $\cdot$ ”表示矢量的内积运算。
- (6) 对 G 进行最佳标量量化得 G^Q 。
- (7) 译码重建矢量为: $X^{\text{rec}} = Y^Q \cdot G^Q$ 。
- (8) 计算量化失真: $D' = (X - X^{\text{rec}}) \cdot (X - X^{\text{rec}})$ 。
- (9) 转(2)循环执行 12 次。
- (10) 在上述循环过程中记下最小的失真 D^* 以及相应的量化结果。

为了进一步提高 GSLVQ 的压缩比, 可用 Huffman 码表示码书中每个格点的标号, 但必须先统计每个格点被使用的概率, 然后构造一个具有 4096 个码字的 Huffman 码表; 此外还可以对最佳增益因子 G 进行 DPCM 编码, 以减少表示增益因子的比特数目。

6 增益-波形矢量量化器在序列图象编码中的应用

图 3 给出 GSLVQ 在序列图象编码中的应用方案。GSLVQ 用来对运动补偿后的差值图象进行压缩。

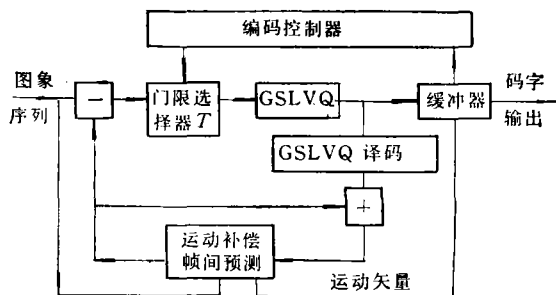


图 3 序列图象编码框图

由于差值图象中绝大部分信号的能量都很小, 因此在图 3 中用门限选择器来选取信号幅度大于给定门限 T 的 16 维矢量, GSLVQ 仅对符合条件的矢量进行压缩, 这样做大大提高了压缩比, 同时编码控制器能够依据码字缓冲器的状态来调节 T 的大小, 从而防止了缓冲区中的码字上溢或下溢。

在实验中我们采用 CIF 格式的 Miss America 序列图象来测试编码方案的性能, 取门限 $T = 10$ 。图 4 给出编码比特率和峰-峰信噪比曲线, 其中编码比特率除了 GSLVQ

码字外,还包括运动矢量的 Huffman 码和各种控制信息的编码比特(运动矢量和某些控制信息的编码参考了 H·261 建议^[9]).

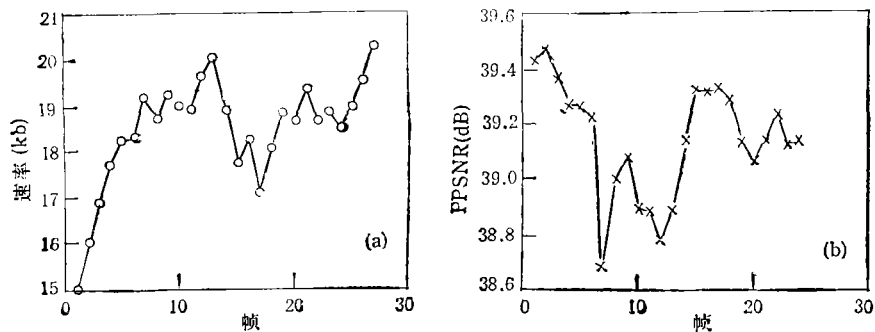


图 4 编码实验结果(编码比特率和峰-峰信噪比)



图 5 实验结果图象

图 4 给出连续二十多帧序列图象的编码结果,得到平均比特率为 18.47kbit/帧,平均峰-峰信噪比高达 39.14dB.

图 5 给出 5 幅图象(由于是序列图象,本文只取其中一帧的编码结果),包括相继两帧原始图象(图 5(a),5(b)),运动补偿后的帧间差值图象(图 5(c)),经 GSLVQ 编码后的重建差值图象(图 5(d))和译码图象(图 5(e)),其中图 5(c),5(d)的像素值被放大 3 倍.从图 5(c)来看,GSLVQ 有选择地对差值较大的区域进行量化编码,而这些区域能否进行有效编码对译码图象质量来说是非常关键的.图 5(e)表明译码图象质量非常高,几乎看不

出图象的失真。

7 结 论

本文给出了基于旋转 Barnes-Wall 格的矢量量化器设计方法和快速量化算法,在此基础上设计了增益-波形矢量量化器,并将它用于序列图象编码之中。从实验结果可看出, GSLVQ 非常适合于对差值图象进行压缩。

致谢 作者非常感谢王新梅教授给予的指导。

参 考 文 献

- [1] Linde Y, Buzo A, Gray R M. IEEE Trans. on COM, 1980, COM-28(1): 84—95.
- [2] Conway J H, Sloane N J A. Sphere Packings, Lattices and Groups. New York: Springer-Verlag, 1988, 31—131.
- [3] Antonini M, Barlaud M, Mathieu P. Image coding using lattice vector quantization of wavelet coefficients, ICASSP' 91, 2273—2276.
- [4] Sampson D, Ghanbari M. IEE Proc.-I, 1993, 140(1): 56—66.
- [5] 陈学育. 分组编码调制研究: [硕士论文]. 西安: 西安电子科技大学, 1992.
- [6] Forney, Jr G D. IEEE Trans. on IT, 1988, IT-34(9): 1123—1151.
- [7] Forney, Jr G D. IEEE Trans. on IT, 1988, IT-34 (9): 1152—1187.
- [8] Eyuboglu M V, Forney Jr G D. IEEE Trans. on IT, 1993, IT-39(1): 46—59.
- [9] Recommendation H. 261, Geneva: 1990.

ROTATED BARNES-WALL LATTICE BASED VECTOR QUANTIZER AND ITS APPLICATION IN IMAGE-SEQUENCE CODING

Xue Xiangyang Chen Xueqing Fan Changxin

(Information Science Institute, Xidian University, Xi'an 710071)

Abstract A new method is presented to design a rotated Barnes-Wall lattice based vector quantizer (LVQ). The construction method of the LVQ and its fast quantizing algorithm are described, then the gain-shape vector quantizer (GSLVQ) with LVQ as shape VQ is discussed. Finally the GSLVQ is used in image-sequence coding and good experimental results are obtained.

Key words Image coding, Vector quantization, Lattice