国防科技大学学报
JOURNAL OF NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY
第12卷第4期　1990年12月　Vol.12 No.4

点阵图形的二次压缩法

万 良 君

（计算机系）

摘　要　基于YH-1巨型机，本文详细介绍了一种计算机图形的二次压缩法。该方法已经在YHHT图形软件上成功地实现，应用结果证明这种方法是高效的。

关键词　计算机图形学，编码，图形/点阵图形，图形压缩，Huffman编码

分类号　TP335.3

随着计算机软件和硬件的高速发展以及绘图设备成本的降低，图形软件越来越丰富。图形软件已成为先进的计算机系统所必不可少的软件。计算机处理图形特别需要大的存储量，尤其是图形库的形成和丰富使得图形所占存储量更大。例如YH绘图软件的一幅点阵图约占768＊16个YH字（每个YH字64比特）。因此，寻找一种既能保证图形不变形和不丢失信息，又能满足可逆压缩的方案，才能解决计算机处理图形所占存储空间过大的问题。不仅如此，随着汉字处理的普及和完善以及汉字库的增加与发展，寻找一种可行的可逆压缩法也是必要的。随着目前分布式系统与分布式应用的迅速发展，对数据压缩技术的需要也越来越迫切，因为压缩是分布式计算的一个基本部分。如果没有压缩，那么实现分布式的图形应用几乎是不可能的。笔者以国产YH-1巨型机为背景，在YH绘图软件的基础上提出并实现了一种二次压缩法。二次压缩法是指：首先对整幅点阵图形进行分块特征压缩；然后再对每个分块利用Huffman编码技术进行第二次信息压缩。实践证明，这种压缩方法是一种高效的、可逆的压缩方法。

1　分块特征压缩

分块特征压缩的思想是，按某种规则把一幅整图划分成若干块，而以1个比特（0或1）来标识具有某种特征的块，这样，一个比特可以区分出两种不同的块。例如：YH-1巨型机上的一幅整图占768＊16个字，它的每个字是64比特的二进制数字，因此这一幅图所需要的二进制位的总数是768＊16＊64＝768＊1024比特。我们按64＊64比特把它分成192个方块（见图1）。这样，用一个12＊16比特的阵列表TAG就可标识一幅768＊1024比特的点阵图形的特征（见图2）。

TAG阵列表中的每个比特可以标识一幅整图中的一块（即64＊64比特）的特征。如果用t_i记TAG的一个比特，那么

$$t_i=\begin{cases}0\text{，标识第 } i \text{ 块全为 } 0\text{；}\\ 1\text{，标识第 } i \text{ 块不全为 } 0\text{；} \quad i=1,\cdots,192\end{cases}$$

1989年12月10日收稿

显然，第 i 块压缩掉当且仅当 $t_i=0$.

TAG共占用了12 * 16＝192比特，也就是3个YH字。所以，最差的情形（即一块也没有压缩掉）不过增加3个YH字的存储量，与原来一幅图768 * 16＝12288个YH字相比很微小，然而这种最差的情形一般是不会出现的。按统计规律来看，一幅图占用存储空间的有效比特（“1”位）与空比特（“0”位）之比约为1:1，而且往往是有效比特、空比特相对集中在同一块中，所以压缩量可望达到一半。换言之，通常会有12288/2＝6144个YH字可以压缩掉，从而实占存储量约6144＋3＝6147个YH字。尽管这只是比较乐观的估算和期望，但YHHT的多年运行证明，“分块特征压缩”的效果是很可观的。图3是第一次压缩的示意图。

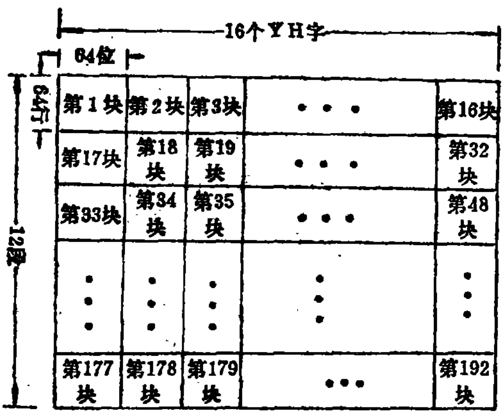


图 1　一幅图划分为192块

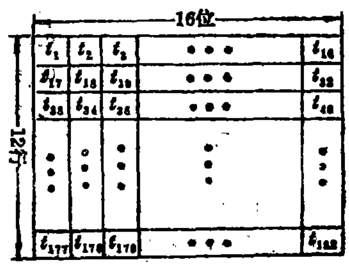
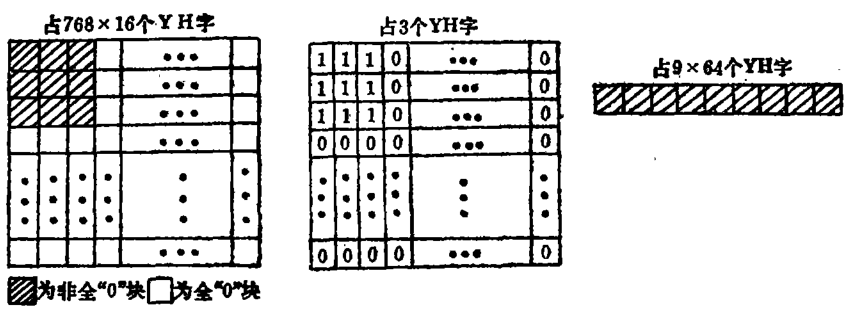


图 2　阵列表TAG



（a）压缩前的一幅图　　（b）对应的TAG点阵　　（c）压缩后的图形存储

图 3　分块特征压缩

2　点阵块的图形压缩

第二次压缩是在分块特征压缩的基础上，再对每个点阵块进行图形压缩。一般而言，对于64 * 64比特的点阵块，若以 N 个点（位）为单位再把它划分若干子块，则每个子块 B_i 的形式为 $b_1b_2\cdots b_N$，其中 $b_k\in\{0,1\}$；$k=1,2,\cdots,N$. 根据信息论的基本概念，子块 B_i 的二进制码的信息源熵（信息源所包含的平均信息量）H 为

$$H=-\Sigma P_i\log_2 P_i \tag{1}$$

式中P_i为B_i出现的概率。而二进制码的信息冗余量为

$$T=1-H/N \tag{2}$$

对于表1所示的例子，$H=2.52$，$T=16\%$，即每个64*64比特点阵块的冗余量为655.36比特。也就是说，N取3时64个YH字至少有10个冗余字（约六分之一），这个数值还是不小的。那么如何改进子块B_i的信息格式以减少这个冗余量呢？本文采用的是Huffman变长即时编码法。

表1　二进制码长$N=3$的一种概率统计

B_i	出现概率（P_i）
B_1	0.4
B_2	0.2
B_3	0.1
B_4	0.1
B_5	0.05
B_6	0.05
B_7	0.05
B_8	0.05

首先，变长码的优点就在于，它在一定情况下可以用更少的位数来存储相同的图形信息。当然，采用Huffman编码必须对被标识的信息格式的统计特性有所了解。只有在某些信息格式比另外一些信息格式出现机会更多时，采用这种方法才合适。其次，根据Hamming的论证，即时码能很好地满足可译性。所谓即时码，在这里是指一旦有一个二进制位串，可以唯一地识别出这个编码串，进而识别出该编码串所标识的原始信息格式。即时码的特性是没有一个单一码是其它单一码的字头。

本文采用Huffman编码就是因为这种变长的即时码具有图形描述的有效性和可译性。如果用P_i表示分组码B_i的信息格式在点阵图块中的出现概率，l_i表示B_i的Huffman编码长度，则所有Huffman编码的平均二进制位长是

$$L=\sum_{i=1}^{K}P_il_i，(K=2^N) \tag{3}$$

按下降次序取P_i，不影响推论的一般性。要想得到有效的编码，必须满足

$$\begin{cases}P_1\geqslant P_2\geqslant\cdots\geqslant P_K\\ l_1\leqslant l_2\leqslant\cdots\leqslant l_K\end{cases} \tag{4}$$

否则，另指定编码可以得到更短的平均二进制位长。按（4）式的思想，对原始分块信息

表2　缩减过程

B_i	P_i						
B_1	0.4	0.4	0.4	0.4	0.4	0.4	0.6
B_2	0.2	0.2	0.2	0.2	0.2	0.4	0.4
B_3	0.1	0.1	0.1	0.2	0.2	0.2	
B_4	0.1	0.1	0.1	0.1	0.2		
B_5	0.05	0.1	0.1	0.1			
B_6	0.05	0.05	0.1				
B_7	0.05	0.05					
B_8	0.05						

表3　分裂过程

B_i	编码							
B_1	1 — 0.4	1　0.4	1　0.4	1　0.4	1　0.4	1　0.4	1　0.6	0
B_2	01 — 0.2	01　0.2	01　0.2	01　0.2	01　0.2	01　0.4	00　0.4	1
B_3	0010 — 0.1	0010　0.1	0010　0.1	0010　0.2	000　0.2	000　0.2	01	
B_4	0011 — 0.1	0011　0.1	0011　0.1	0011　0.1	0010　0.2	001		
B_5	00010 — 0.05	00010　0.1	0000　0.1	0000　0.1	0011			
B_6	00011 — 0.05	00011　0.05	00010　0.1	0001				
B_7	00000 — 0.05	00000　0.05	00011					
B_8	00001 — 0.05	00001						

步步缩减，尔后步步分裂，分裂到最后就实现了Huffman编码。

仍以表1为例，表2和表3分别是具体缩减过程和分裂过程。对于这个例子可以求出Huffman编码的平均二进制位长 $L=2.6$ 比特，显然很接近于理想最短平均二进制位长。

上面已经指出这种方法是完全可译的，所以对压缩后的图形进行复原时，不会产生任何偏差，即压缩是可逆的。第二次压缩对每个64 * 64比特的点阵来讲，压缩量接近六分之一（按 $N=3$ 划分子块），因此，在第一次压缩的基础上大约又有 6144/6＝1024 个 YH 字可望被压缩掉。故经二次压缩之后，通常可以压缩掉6144＋1024＝7168个YH字。这是十分令人满意的。

3　结束语

点阵图形的可逆压缩的可能性和实用性，是图形软件研制人员与使用人员所共同关心的问题。在超级计算机和分布式系统快速发展与普及应用的今天，尤其迫切需要研究点阵图形的压缩技术。本文介绍的二次压缩法是一种通用技术。如果在并行计算机上实现这种技术，可以利用“分块”的良好无关性，全并行地完成第一次压缩——各个分块的特征压缩。第二次压缩——Huffman编码法，也可独立使用。

参 考 文 献

[1] Hamming R W. Coding and Information Theory. Prentice-Hall, 1980
[2] Campbell G, Defanti T, Frederiksen J, Joyce S, Leske L, Lindberg J, Sandin D. Two Bit/Pixel Full Color Encoding. Computer graphics, 1986, 20(4)

国防科技大学学报
JOURNAL OF NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY
第12卷期4第 1990年12月 Vol.12 No.4

最佳(n, 2, w)二进制等重检错码的研究

肖 戎

(电子技术系)

摘 要 本文研究了最佳(n,2,w)二进制等重检错码的存在性问题。对于文[5]中n为偶数时所得的结论，本文给了一个简练的证明。更为重要的是，利用以上方法，作者证明了文[5]中关于n为奇数时的一个猜想。

关键词 编码，检错码，等重码

分类号 O157.4

在数字通信中，有时使用"n中取w码"作为差错控制系统中的检错码。"n中取w码"即为码长为n，每个码字的重量都为w，码字之间的最小距离为2的(n,2, w)码，也即二进制等重码。例如，在国内，常采用(5,2,2)码或其对偶码——(5,2,3)码。而在国际上，也常使用(7,2,3)码，(8,2,4)码或它们的对偶码。显然，这些检错码的检错性能与效率依赖于这些码的不可检错误概率，因而，如何分析这些码的不可检错误概率就是一个无论从编码学理论或从实际角度来看，都是具有重要意义的问题。

最佳检错码的存在性即是与码的不可检错误概率密切联系的一个极为重要的问题。

在二进制对称信道中，误码率$p_e \leqslant 0.5$.当$p_e = 0.5$时，则信道不能传送任何信息。此时，无论发端发送任何码字，收端都以等概的形式接收所有可能错误图样中的任何一个。例如，若发端是发送(n, 2, w)码中的码字，则有2^n种可能的错误图样，而

1989年7月26日收稿

A Double Compression Method for Dot Matrix Graphics

Wan Liangjun

(Department of Computer Science)

Abstract

This paper describes a double compression method with characteristic block compression, and then Huffman block encoding for computer graphics in detail. The method has been implemented in the YHHT graphic software. It has been shown by applications that the method is highly efficient.

Key words computer graphics, coding, graph/dot matrix graphics, graphic compression, Huffman encoding