

基于区域分割技术的图像无损压缩算法研究

Lossless Image Compression Based on Image Segmentation Technique

元燕辉 杨长生

(浙江大学计算机系系统工程研究所 杭州310027)

Abstract First, some of main factors which play very important roles in the image compression are extracted. They include prediction coefficient, scan order and neighborhood function. Second, the effect of these factors on image compression is analyzed. Third, a model is given.

Keywords Spatial prediction, Image segmentation

1. 引言

在某些场合,需要压缩后重建图像的所有像素与原始图像的对应像素完全相等,这种压缩技术就是图像的无损压缩。无损压缩技术具有广泛的应用:医学射线图像往往要求进行无损压缩,以免造成诊断错误;卫星遥感图像往往也要求进行无损压缩,以减轻通信信道载荷,同时又保持遥测精度。进行无损压缩可以采取以下的两个步骤:首先进行解相关,尽量消除图像的冗余,得到误差图像;然后对误差图像进行可变长二进制编码。

图像的解相关常使用 DPCM 方法来实现。DPCM 解相关方法包括两个部分:一个是邻域函数,它用来确定在预测过程中使用哪几个已编码的邻接像素点;另一个组成部分是指由邻域函数确定的相邻像素点如何在预测中使用,一般使用线性模型,从而转化为求线性模型的系数即预测系数。

由于邻域函数在表示图像的空间结构方面限制很大,往往无法很好地表达图像的空间特性,这一缺陷很大程度上影响着图像的压缩效率。如果能够将图像的空间结构预先表示出来,那么邻域函数将会更有效。要将图像的空间结构表达出来比较困难,一个折衷的方案是对图像采取合适的扫描顺序。在构造扫描顺序的同时,对图像进行区域分割,从而能够在一定程度上表现图像的空间结构。下面将讨论预测系数、扫描顺序和邻域函数对图像压缩的作用;然后给出压缩模型,分析模型的误差;最后给出试验结果和结论。

2. 压缩因素

以往的预测编码模型对图像是采用 Z 字扫描方式,这种扫描方式的局限性很大,特别是它无法充分利用相邻的像素点的信息,像图1所显示的那样,像素点7只能利用1—6这六个点来做预测,但实际上很可能像素点8、11、12的作用更为明显。像 Z 扫描那样得到的邻域函数最多只能利用一半的空间有效信息,我们称这样的邻域函数为半度空间型;如果邻域函数能够自由选择邻域空间,我们就称之为全度空间型^[1]。文[1]的试验结果表明,使用全度空间型的邻域函数能够提高压缩效率,但是应当看到不管是用何种邻域函数都是为了充分使用图像本身的空间特性。图像的区域化能够对图像的空间特性给出一定描述,在这里我们采用适当的图像扫描策略来完成这一步。

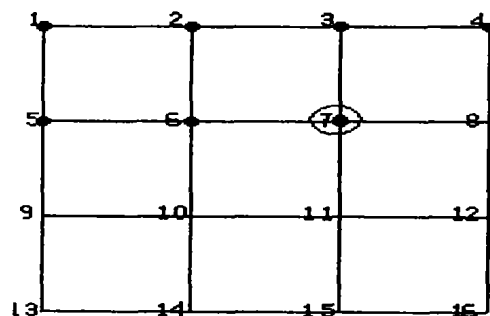


图1

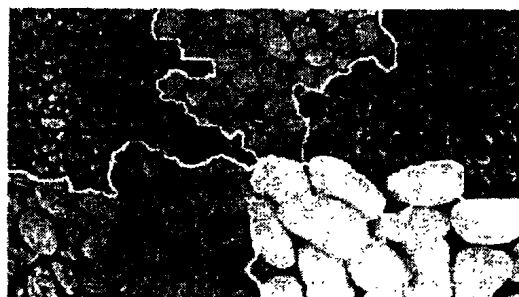


图2

在图像的扫描顺序确定下来以后,需要选择适当的邻域函数。在划分好的区域内图像可能存在着各种复杂的细节,使用单一的邻域函数无法表达这些细节,需要采用一些自适应的方式,但是这些方式不能太多,否则反而影响压缩效率。邻域函数在模型中的作用是表达大区域里头的图像细节,这一点表明图像可以以区域为单位的形式来进行压缩。传输时,可以一个区域一个区域进行传输,如上面图2所示。

邻域函数确定以后,要确定函数里头各个参量的系数,即求取预测系数,预测系数表明的是相邻像素点之间的作用关系。下一节,我们将依据这些压缩因素给出压缩模型。

3. 压缩模型

压缩模型包括两个部分:一是差值函数,用来表示预测值和真实值之间的差别;另一个是目标函数,用来计算模型的预测误差。

$$\text{差值函数 } \epsilon_i = V(S_{i+1}) - C_{T_i} T_i(U_i)$$

元燕辉 研究生,研究方向为多媒体数据压缩、移动数据库等。杨长生 教授,研究方向为多媒体数据压缩等。

1) S 是扫描函数, S_i 表示第 i 步扫描到哪个点 ($i=1, 2, \dots, M \times N$), S 能够满足以下几点:

• S 能够遍历整个图像

• S 能够保证对图像的每一个点只遍历一次

• 在 S_i 周围有尽量多的已知点, 并且这些点最好在某个度量空间具有一定的相似性, 这一点是为了将图像进行区域化而设置的

• S 的表达尽量简洁, 最好能够函数化

以上几点中第一点和第二点是必需的, 利用第三点我们可以将图像区域化, S_i 周围点在某个度量空间的相似性, 其实也是构造扫描函数的一个评判标准。而这个度量空间对于图像而言, 不仅包括像素点的灰度值, 还包括其它一些测度, 如像素点的走向 θ 等等。

2) V 是求当前点的灰度值函数

3) U 是一个集合函数, 它的定义: $U_1 = \{S_1\}$, $U_i = U_{i-1} \cup \{S_i\}$, 其中 $i=2, 3, \dots, M \times N$

4) T 是一个求给定集合某个子集并将该子集表示成一个向量的函数, T_i 表示求集合 U_i 的某个子集并将它表示成一个向量, 子集和向量的对应关系是: 子集的每一个元素对应向量的一个分量, $\{S_1, S_2, \dots, S_i\} \Rightarrow (S_1, S_2, \dots, S_i)$

5) C 是一个相关系数向量, C_{T_i} 表示是面向向量 T_i 的。

• 目标函数 $\sigma = \sum_{i=1}^{M \times N} \epsilon_i^2$, 目标是使 σ 最小

从差值函数的定义可以看出, 它利用了第二部分所分析的三个压缩因素。 S 对应压缩因素中的扫描顺序, T 和 U 对应着邻域函数, 而 C 则表示了预测系数。对于整个图像而言在确定了 S 以后 U 也就确定了, 从而整个图像的区域化也就完成了。但是对于图像区域中的每一个点, 为了求得最小的 ϵ , 使用的 T 可能不一样, 而 C 也会因此而变化。为了表示这些 T, C 的信息, 可能就要耗费大量的资源, 这是让人无法接受的。要减少 T, C 的个数, 可以将图像分块, 以块代替点; 另外由于用来预测的相邻像素不是越多越好, 因此可以将区域里头 T 的维数定下来, 这样减少计算的复杂性。

模型确定后有两个问题需要解决: 一是如何定义度量空间, 另一个是如果得到的区域个数比较多, 应该怎么做。第一个问题是一个比较复杂的问题, 因为它的选取关系到区域分割的好坏, 而区域的分割是个比较复杂的问题^[3]。但是在本文中由于区域的分割只是手段不是目的, 因此我们在算法的精确度和效率上寻求一种平衡。为此定义度量空间上点与集合的距离: $d = \sum w_i |d_{i, index}|$, $0 \leq i \leq index$, 其中 w_i 是点 I_i 与点 I_{index} 的距离 $d_{i, index}$ 在点 I_{index} 与点集距离中所具有的权值。其中 $d_{i, index}$ 指的是两个点在空间上的能量差距, 而 w_i 则表现为区域空间上的频率统计特性。 w_i 决定了区域的分布特征, 而 $d_{i, index}$ 则决定了区域细节。为了简化计算, 我们对 w_i 作出限制, 设定一个阈值 D_{ipare} , 当两个点的空间距离大于等于 D_{ipare} 时, w_i 取零值。如果得到的区域比较多, 则需要将一部分区域进行合并。一般选择两个边界相邻尽量多和区域的平均能量尽量接近的区域来进行合并。

在得到区域的划分后, 由于每个区域内部的像素分布不

一样, 需要分别为它们选择 T 和计算预测系数向量 C 。 T 的选择标准主要是参照点在空间位置上的相邻关系和在扫描序列中的相邻关系。在 T 的结构确定以后通过 $\sigma = \sum_{i=1}^{|W_{region}|} \epsilon_i^2$ (其中 $|W_{region}|$ 表示区域中点的个数) 来求 C 。令 $\frac{d\sigma}{dC} = 0$, 即 $AC = 0$ (其中 A 是一个 $n \times n$ 的相关矩阵, 完全取决于区域的信息)。

4. 试验结果和结论

这里使用 JPEG 标准委员会提供的数字化为 8bpp 的几幅灰度图像来测试所提出的无失真编码方法, 图像分辨率为 720×576 。下表给出了各种压缩算法对测试的图像进行压缩后的 bpp (bits per pixel) 对比。表第一列为 JPEG 无失真模式, 使用 JPEG 建议的 8 个预测器中最好的一个预测器和静态 Huffman 表; FELICS, SUNSET, CQM 在文[1]中有所论述, LOCO-I 是 JPEG-LS 的核心算法。

实验数据表格 (B/Pixel)

算法 图像	JPEG	FEL ICS	SUN SET	CQM	UCM	LOCO -I	本文 算法
Balloon	3.17	3.21	2.89	2.82	2.81	2.90	2.77
Board	4.18	4.03	3.73	3.62	3.57	3.64	3.53
Girl	4.26	4.35	3.91	3.82	3.81	3.90	3.79
Gold	4.85	4.75	4.56	4.44	4.45	4.47	4.44
Hotel	4.93	4.71	4.46	4.35	4.28	4.35	4.31

从上面的表格中可以看出本文算法的压缩效果要优于 JPEG、FELICS、SUNSET 和 LOCO-I^[1] 等通行的压缩算法; 而与被认为是图像无失真编码极限的 UCM 算法和 CQM^[1] 算法效果相当。另外, 由于本文算法很好地利用了图像在空间的分布特性, 对图像进行区域化, 从而使得算法对区域特征明显的图像压缩时效果更好。从第三节的算法实现来看, 本文算法的时间复杂度也比较低。对于 $N \times N$ 的图像, 本文算法核心部分的时间复杂度 $T_p = O(4D_{ipare}^2 N^2)$, 其中 D_{ipare} 是一个阈值常数 (定义见第 3 节)。

参 考 文 献

- 1 Wu Xiao. Lossless Compression of Continuous-Tone Images via Context Selection, Quantization, and Modeling. IEEE TRANSACTIONS ON IMAGE PROCESSING VOL. 6, NO. 5, MAY 1997. 656~664
- 2 Won C S. A Block-Based MAP Segmentation for Image Compressions. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, 1998, 8(5): 592~601
- 3 Ma Wei-Ying, Manjunath B S. EdgeFlow: A Technique for Boundary Detection and Image Segmentation. IEEE TRANSACTIONS ON IMAGE PROCESSING, 2000, 9(8): 1375~1388
- 4 Weinberger M J, Seroussi G, Sapiro G. The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS. IEEE TRANSACTIONS ON IMAGE PROCESSING, 2000, 9(8): 1309~1324