

# 一种新的基于 TNAM 的二值图像表示方法<sup>\*</sup>)

郑运平 陈传波 黄 巍

(华中科技大学计算机科学与技术学院 武汉 430074)

**摘 要** 分层数据结构在计算机视觉、机器人、计算机图形学、图像处理、模式识别等领域里是非常重要的区域表示方法,但是这种结构过于强调分割的对称性,因此不是最优的表示方法。借助于三角形 Packing 问题的思想,提出了一种新的基于 TNAM(三角形非对称逆布局的模式表示模型)的二值图像表示方法。通过描述 TNAM 模型,给出了一个新的基于 TNAM 的二值图像表示的编解码算法并分析了该算法的总的的数据量。理论分析和实验结果均表明:与流行的线性四元树表示方法相比,基于 TNAM 的二值图像表示方法能够更有效地减少数据存储空间,是二值图像的一种较好的表示方法。这种方法可以应用于二值图像模式表示的各个方面,在降低存储空间、提高传输速度、加快处理过程等方面具有良好的理论参考意义和实际应用价值。

**关键词** 图像表示, TNAM, 线性四元树, 图像复杂度, 二值图像

## Novel TNAM-based Representation Method for Binary Images

ZHENG Yun-ping CHEN Chuan-bo HUANG Wei

(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

**Abstract** The hierarchical data structures are very important region representation methods in computer visualization, robotics, computer graphics, image processing, and pattern recognition, but these methods put too much emphasis upon the symmetry of segmentation. Therefore, they are not the optimal representation methods. Inspired by the concept of the triangle packing problem, presented a novel triangle non-symmetry and anti-packing pattern representation model-based (TNAM-based) representation method for binary images. By describing the TNAM, a novel encoding and decoding algorithm of TNAM-based representation method for binary images was proposed. Also, the total data amount of the proposed algorithm was analyzed. The theoretical and experimental results presented both showed that the TNAM-based representation method for binary images can reduce the data storage much more effectively than the popular linear quadtree method can do, and therefore it is a better method to represent binary images. The TNAM-based representation method for binary images presented is valuable for the theoretical research and potential practical values such as decreasing the storage room, increasing the transmission speed, quickening the process procedure, and so on.

**Keywords** Image representation, Triangle NAM, Linear quadtree, Image complexity, Binary image

### 1 引言

分层数据结构在计算机视觉、机器人、计算机图形学、图像处理、模式识别等领域里是非常重要的区域表示方法<sup>[1-4]</sup>。传统的直观的区域表示方法是二维数组表示,但由于图像处理在各方面的广泛应用和其它学科的飞速发展,对图像处理的研究提出了处理速度和存储空间的更高要求。这样,二维数组的空间效率及其“样点-样点”的运算方式已不能适应发展的需要,取而代之的应该是既紧凑又便于做各种图像处理运算的表示方法<sup>[5-7]</sup>。四元树是图像的分层表示的一种形式,它是基于图像数组的四元分割。四元树数据结构之所以有效,可以归结于它的层次结构和“块-块”运算方式。四元树表示是研究得最早的,也是研究得最多的一种分层表示形式。20 世纪 70 年代初, Klinger<sup>[8]</sup> 首先提出了用四元树表示二值图像的方案。在 Samet<sup>[9]</sup> 给出了四元树的机器内部表示以后,四元树表示形式才得到了大量的研究。早期的四元树表示都是基于指针的四元树结构,为了进一步减少存储空间,

Gargantini<sup>[10]</sup> 消除了指针方案,提出了称之为线性四元树的表示方法。一般情况下,线性四元树可节省 66% 的存储空间;特殊情况下,可节省高于 90% 的存储空间。基于四元树表示的理论上的成果很多,运用于实际的也不少,并越来越多<sup>[11-14]</sup>。然而,尽管线性四元树表示有许多优点,但是它们过于强调分割的对称性,因此不是最优的表示方法。以寻找分割最大化的非对称分割方法为目标,文献[15]提出了一种基于非对称逆布局的模式表示模型(NAM)的彩色图像表示方法,该方法与流行的基于线性四元树的彩色图像表示方法相比,不仅前者的子模式数远小于后者的节点数,而且前者的总数据量也远小于后者的总数据量,因此前者能更有效地减少数据存储空间,是彩色图像模式表示的一种良好方法。不同于文献[15]中的矩形子模式,文献[16]提出了一种基于 k-line 子模式的彩色图像表示方法。矩形子模式和 k-line 子模式比较适合图像具有一定的块状性,对于图像的块状性不是很强或非块状类图像,可以考虑用非块状类子模式,比如:三角形子模式。有鉴于此,文献[17]提出了一种新的三角形非

<sup>\*</sup>) 国家高技术研究发展计划(863)(2006AA04Z211)。郑运平 博士研究生,主要研究方向为图像处理与模式识别;陈传波 博士,教授,博士生导师,主要研究方向为计算机网络与信息工程、图像处理与模式识别;黄 巍 博士研究生,主要方向为计算机图形图像处理。

对称逆布局的模式表示模型(TNAM)的灰度图像表示算法,该算法对非块状类图像具有较强的适应性,是非块状类图像的一种高效算法。

事实上,文献[15]和[16]都是基于位平面分解思想,将一幅彩色图像转为对若干二值图像的处理,只是处理时所选取的子模式分别是矩形和 k-line,因此,从本质上来说,这两种表示算法均可归于对二值图像的处理。文献[17]并没有采用位平面分解的思想,而是直接对灰度图像进行处理,用三角形子模式对灰度图像进行逆布局,因此,该文的算法只适合对灰度图像表示,并不适合对二值图像表示。有鉴于此,本文提出了一种新的基于 TNAM 的二值图像表示方法。理论分析和实验结果均表明:与流行的线性四元树表示方法相比,基于 TNAM 的二值图像表示方法能够更有效地减少数据存储空间,是二值图像的一种较好的表示方法。

## 2 TNAM 的描述

本节先给出 TNAM 的抽象描述,然后以二值图像模式为例来说明 TNAM 的思想。

### 2.1 TNAM 的抽象描述

设原图像模式为  $\Gamma$ , 恢复后的非失真图像模式为  $\Gamma'$ , 失真图像模式为  $\Gamma''$ , 则: TNAM 模型是由  $\Gamma$  到  $\Gamma'$  的一个非失真变换, 或者是由  $\Gamma$  到  $\Gamma''$  的一个失真变换:

$$\Gamma' = T(\Gamma), \Gamma'' \approx T(\Gamma)$$

其中  $T(\cdot)$  是正向变换函数, 或称为编码函数。

正向编码过程为:

$$\Gamma' = \bigcup_{j=1}^n p_j(v, A | A = \{a_1, a_2, \dots, a_{m_j}\}) + \varepsilon(d)$$

其中  $\Gamma'$  是编码后合成的模式;  $P = \{p_1, p_2, \dots, p_n\}$  是预先定义的三角形子模式集合,  $n$  是三角形子模式类型的总数,  $p_j \in P$  是  $P$  中第  $j$  类三角形子模式 ( $1 \leq j \leq n$ );  $v$  是  $p_j$  的值,  $A$  是  $p_j$  的参数集合;  $a_i (1 \leq i \leq m_j)$  是  $p_j$  的形状参数集合,  $m$  是  $p_j$  的编号;  $\varepsilon(d)$  是残渣模式(容器中的垃圾),  $d$  为残渣空间尺度的阈值。

### 2.2 TNAM 模型的思想

三角形 Packing 问题是一类特殊的 Packing 问题。三角形 packing 问题的提法是: 给定一个矩形容器  $R$  和  $n$  个任意形状和大小的三角形, 问: 这  $n$  个三角形能否互不重叠地放入矩形容器  $R$  里? 若能, 则给出每个三角形在矩形容器里的位置和方向; 若不能, 则作出该算法放不下的回答。

TNAM 是三角形 Packing 问题的一个反问题, 其基本思想是: 给定一个模式和  $n$  个预先定义的三角形子模式, 现在要从这个给定的模式中抽出这些三角形子模式, 用这些子模式的组合来表示给定的模式。

下面以一个简单实例(如图 1 所示)来说明 TNAM 的思想。

设给定的待逆布局的二值图像模式为图 1(a), 其大小为  $2^2 \times 2^2$ 。图 1 中白色点表示图像背景点的像素值为 1, 黑色点表示图像区域的像素值为 0。在实例中, 预先定义的模式集合为一个三角形子模式。这里的三角形子模式是带参数的, 三角形的参数为三角形的三个顶点, 即  $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$ 。图 1(b), (c), (d), (e), (f), (g), (h) 分别是用不同逆布局算法所得到的 TNAM 表示结果。事实上, 逆布局的结果远远不止这些, 我们只是从中选择了比较有代表性的 7 个结果。由于形成一个三角形至少需要三个点, 因此在

TNAM 逆布局结果中有可能出现线段和孤立点的情况, 对于线段需要存储线段的两个端点坐标, 而对于孤立点直接存储其坐标即可。

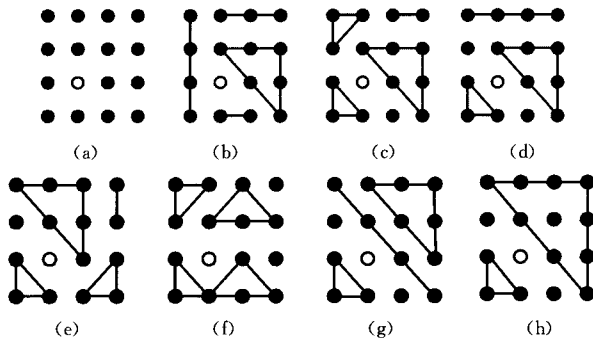


图 1 TNAM 思想的一个简单实例

由这个例子可以看出, 逆布局的方式不是唯一的, 不同的划分(逆布局)方法, 其表示效率是不一样的, 因此逆布局算法的优化是必要的。在 TNAM 表示中, 为了达到最佳的表示, 首先必须使三角形子模式尽可能最大, 然后在不能形成三角形的情况下, 必须使线段尽可能最长。

以寻求最大表示效率为目标, 本文提出了一种新的基于 TNAM 的二值图像表示算法, 在该算法中, 为了降低 TNAM 图像模式表示的总数据量, 将三角形子模式分成了两类, 即上三角形和下三角形。这样, 对于任何一个三角形, 不需要存储三个顶点的坐标, 而只要存储斜边的两个端点和一个用于标识三角形子模式类型的标识符(比如上三角用 0, 下三角用 1 表示)即可。相反, 以斜边的两个端点和三角形类型的标识符, 也可以解码出三角形子模式来。即本文是以  $p = \{triangle | triangle = (flag, point1\_hyp, point2\_hyp)\}$  表示一个三角形子模式的。事实上, 图 1(h) 就是用本文算法所获得的逆布局结果(包含一个大的上三角形和一个小的下三角形以及两个孤立点), 它明显优于其它的逆布局结果。下一节将给出该算法的具体描述。

## 3 基于 TNAM 的二值图像表示算法

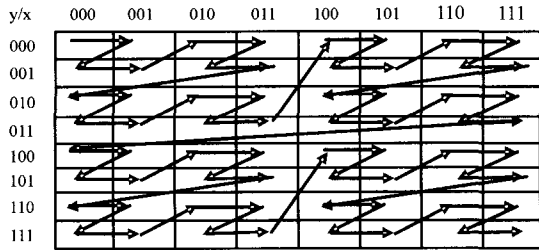
基于 TNAM 的二值图像表示的编解码算法中均用到了 K 码变换规则<sup>[15]</sup>, 因此在描述算法之前, 有必要对 K 码变换规则进行介绍。

### 3.1 K 码变换规则

设已经布局好了的图像模式用  $F = \{f(x, y)\}$  来表示, 并且  $F = \{f(x, y)\}$  的大小为  $2^n \times 2^n$ 。令  $x = (x_{n-1} x_{n-2} \dots x_1 x_0)_2$  和  $y = (y_{n-1} y_{n-2} \dots y_1 y_0)_2$ , 构造一个一维的坐标变量  $k$ , 有:  $k = (y_{n-1} x_{n-1} y_{n-2} x_{n-2} \dots y_1 x_1 y_0 x_0)_2$ , 这样就将一个二维图像模式转换为一个一维序列, 即  $F = \{f(x, y)\} = \{g(k)\}$ , 简称这种一维的表示为 K 码。由二维到一维的降维变换称为 K 码正变换, 记为:  $k = K(x, y)$ 。由一维到二维的升维变换称为 K 码的反变换, 记为  $(x, y) = k - 1(k)$ 。

y/x	000	001	010	011	100	101	110	111
000	0	1	4	5	16	17	20	21
001	2	3	6	7	18	19	22	23
010	8	9	12	13	24	25	28	29
011	10	11	14	15	26	27	30	31
100	32	33	36	37	48	49	52	53
101	34	35	38	39	50	51	54	55
110	40	41	44	45	56	57	60	61
111	42	43	46	47	58	59	62	63

(a) K 码变换示意图



(b) K 码走向示意图

图 2 K 码示意图

图 2 (a)是一个二维到一维变换的形象表示,图中的数字顺序代表了 K 码的走向。图 2 (b)是 K 码走向示意图,形象地表示了 K 码的变化过程。

### 3.2 基于 TNAM 的二值图像表示的编解码算法

本算法中被逆布局的子模式对象是任意大小的上三角形和下三角形,即三角形子模式为  $p = \{triangle \mid triangle = (flag, point1\_hyp, point2\_hyp)\}$ ,当  $flag=0$  时为上三角形子模式,而当  $flag=1$  时为下三角形子模式。下面将分别给出编解码算法的表述及其具体步骤。

编码算法的表述是:对于给定的一个大小为  $2^n \times 2^n$  二值图像  $f$ ,将编码结果存入三个编码队列集合  $Q\_tri$ ,  $Q\_line$  和  $Q\_point$  中。

以下是编码算法的具体步骤:

Step 1 从二值图像  $f$  的第一个入口开始,首先确定一个未被标识的上(或下)三角形子模式,再根据三角形子模式的匹配(逆布局)算法来尽可能地形成最大的上(或下)三角形子模式(这里最大的三角形是指该三角形的面积最大,也即此三角形包含的像素总数最多)。

Step 2 如果找到的最大三角形子模式为上三角形,则将  $flag$  赋为 0,且将这个最大上三角形子模式作标识;否则,找到的最大三角形子模式为下三角形,则将  $flag$  赋为 1,且将这个最大下三角形子模式作标识。

Step 3 将三角形的计数变量  $tri\_num$  的值加 1,记录斜边端点的两个坐标  $(x_1, y_1)$  和  $(x_2, y_2)$ ,然后对斜边的两个端点作 K 码降维变换,即:  $point1\_hyp \leftarrow K(x_1, y_1)$ ,  $point2\_hyp \leftarrow K(x_2, y_2)$ ,最后将  $flag, point1\_hyp, point2\_hyp$  这三个变量存储到队列  $Q\_tri$  中,即有  $Q\_tri\{tri\_num\} \leftarrow \{flag, point1\_hyp, point2\_hyp\}$ 。

Step 4 循环执行 Step 1 到 Step 3,直到不能形成新的上(或下)三角形子模式为止,输出三角形的 NAM 队列  $Q\_tri$ 。

Step 5 从标记过的图像  $f$  的第一个入口开始,首先确定一个未被标记的点,再根据子模式的匹配(逆布局)算法来尽可能地形成最长的线段,如果能形成线段,则将线段的计数变量  $line\_num$  的值加 1,记录线段端点的两个坐标  $(x_1, y_1)$  和  $(x_2, y_2)$ ,然后对线段的两个端点作 K 码降维变换,即:  $point1\_line \leftarrow K(x_1, y_1)$ ;  $point2\_line \leftarrow K(x_2, y_2)$ ,最后将这两个变量存储到队列  $Q\_line$  中,即有  $Q\_line\{line\_num\} \leftarrow \{point1\_line, point2\_line\}$ ,且将存储过的此线段在模式中作标识,否则,说明只能形成孤立点,执行 Step 6。

Step 6 直接存储孤立点的坐标  $(x, y)$ ,将孤立点的计数变量  $point\_num$  的值加 1,然后将这个点作 K 码降维变换,即:  $single\_point \leftarrow K(x, y)$ ,最后将这个变量存储到队列  $Q\_point$

中,即有  $Q\_point\{point\_num\} \leftarrow \{single\_point\}$ ,并将此点在模式中作标识。

Step 7 循环执行 Step 5 到 Step 6,直到不能形成新的线段和孤立点为止,输出线段和孤立点的 NAM 队列  $Q\_line$  和  $Q\_point$ 。

解码算法的表述是:对于给定三个编码队列集合  $Q\_tri$ ,  $Q\_line$ ,  $Q\_point$ ,以及图像的分辨率  $n$ ,将解码结果存入二值图像  $f$  中。本算法中不妨以矩阵形式存放图像。

以下是解码算法的具体步骤:

Step 1 将二值图像  $f$  赋值为  $2^n \times 2^n$  大小的全 1 矩阵。

Step 2 根据队列  $Q\_tri$ ,计算出三角形子模式的总数  $tri\_num$ ,对于其中的每个三角形,依次获取三角形的类型的标识符  $flag$  及斜边的两个端点  $point1\_hyp, point2\_hyp$ ,然后将这两个点作 K 码升维变换(即  $(x_1, y_1) \leftarrow k^{-1}(point1\_hyp)$ ;  $(x_2, y_2) \leftarrow k^{-1}(point2\_hyp)$ ),得到斜边的这两个端点坐标后,当  $flag=0$  时即可解码出上三角形,否则即可解码出下三角形,并将矩阵  $f$  对应元素赋值为 0。

Step 3 根据队列  $Q\_line$ ,计算出线段的总数  $line\_num$ ,对于其中的每条线段,依次获取每条线段的两个端点  $point1\_line$  和  $point2\_line$ ,然后将这两个点作 K 码升维变换(即  $(x_1, y_1) \leftarrow k^{-1}(point1\_line)$ ;  $(x_2, y_2) \leftarrow k^{-1}(point2\_line)$ ),得到线段的这两个端点坐标后即可解码出线段,然后将矩阵  $f$  对应元素赋值为 0。

Step 4 根据队列  $Q\_point$ ,计算出孤立点的总数  $point\_num$ ,对于其中的每个孤立点,依次获取每个孤立点  $single\_point$ ,然后将这个点作 K 码升维变换(即  $(x, y) \leftarrow k^{-1}(single\_point)$ ),得到孤立点的坐标,然后将矩阵  $f$  对应元素赋值为 0。

Step 5 输出矩阵  $f$ 。

## 4 基于 TNAM 的二值图像表示的存储结构

本节描述  $p = \{triangle \mid triangle = (flag, point1\_hyp, point2\_hyp)\}$  情况下的基于 TNAM 的二值图像表示的存储结构,其中  $flag$  只占 1 个 bit,  $flag=0$  时表示上三角形;  $flag=1$  时表示下三角形,  $point1\_hyp$  和  $point2\_hyp$  表示斜边的两个端点。

由于形成一个三角形至少需要三个点,因此在 TNAM 逆布局结果中除了三角形外,还有可能会出现线段和孤立点的情况。本节分别讨论这三种图形的存储结构。

### 4.1 三角形的存储结构

在三角形子模式部分,算法的输出是一个顺序存储的队列  $Q\_tri$ ,它的构成主要由 3 个元素组成,即三角形子模式的类型标识符以及斜边的两个端点坐标。存储结构如图 3 所示。

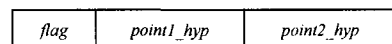


图 3 三角形的存储结构

对于一个  $2^n \times 2^n$  图像模式来说,  $point1\_hyp$  和  $point2\_hyp$  均为坐标对  $(x_i, y_i)$ ,  $x_i$  和  $y_i$  的二进制码长度都为  $n$ 。两个点的具体存储记录用 K 码表示。K 可以用相对值来记录,本次记录的 K 域用本次 K 码减去上一个 K 码的差值来记录,即  $\Delta K = K_i - K_{i-1}$ 。在统计意义,  $\Delta k$  长度为  $n$ ,在实际情况下,如果  $\Delta K$  的长度超过了  $n$ ,则可以将该块拆分为两个

块,用两个记录来表示。因此,按照 K 的定义,存储一个三角形子模式占  $(2n+1)$  位。

#### 4.2 线段和孤立点的存储结构

算法中线段和点的存储结构相对较为简单,对于线段来说,只需存储线段的两个端点  $point1\_line$  和  $point2\_line$ ,而对于孤立点直接存储  $single\_point$  即可,并且线段的两个端点和孤立点也用 K 码来表示,按照 K 的定义,存储一条线段占  $2n$  位,存储一个孤立点占  $n$  位。

### 5 算法的数据量分析

设二值图像模式的大小为  $2^n \times 2^n$ ,即像素总数为  $N_f = 4^n$ ,逆布局后的三角形、线段、孤立点的子模式数分别为  $N_t$ ,  $N_l$  和  $N_p$ ,其总数据量记为  $H_t$ 。定义线性四元树的黑色节点数为  $N_{LQT}$ ,总数据量为  $H_{LQT}$ 。

对于基于 TNAM 的二值图像表示方法来说,存贮一个三角形记录占  $2n+1$  位,一个线段记录占  $2n$  位,一个孤立点占  $n$  位,则其逆布局后的总数据量  $H_t$  为:

$$H_t = (2n+1)N_t + 2nN_l + nN_p \quad (1)$$

对于基于线性四元树的二值图像表示方法来说,存贮一个节点占  $(3(n-1)+2)$  位<sup>[10]</sup>,则线性四元树的总数据量  $H_{LQT}$  为:

$$H_{LQT} = (3n-1)N_{LQT} \quad (2)$$

设  $\varphi_{LQT}$  为线性四元树的总数据量与 TNAM 的总数据量的比值,则有:

$$\varphi_{LQT} = \frac{(3n-1)N_{LQT}}{(2n+1)N_t + 2nN_l + nN_p} > \frac{(3n-1)N_{LQT}}{(2n+1)(N_t + N_l + N_p)} \quad (3)$$

式(3)反映了线性四元树的总数据量与 TNAM 的总数据量的比值。通过这一比值,可以比较 TNAM 相对于线性四元树的优劣。由于线性四元树分割是对称形的分割,其分割方式受到很大的限制,而 TNAM 的分割是非对称的逆布局,分割是最大限度地形成子模式,且三角形分割的灵活性更大,一般情况下, TNAM 逆布局后的子模式总数比线性四元树的节点数少得多,即  $N_t + N_l + N_p < N_{LQT}$ ,因此  $\varphi_{LQT} > (3n-1)/(2n+1)$ 。比如:当  $n=8$  时,线性四元树的总数据量至少是 TNAM 的总数据量的 1.35 倍以上。因此,与流行的基于线性四元树的二值图像表示方法相比,从理论上来说,基于 TNAM 的二值图像表示方法能够更有效地减少数据存储空间。

### 6 实验结果及分析

为了更好地验证基于 TNAM 的二值图像表示方法的理论结果,本节从实验的角度来加以说明。在分析实验结果之前,先给出二值图像模式的复杂度的定义。

**定义 1** 二值图像模式的复杂度:设二值图像模式  $f$  的大小为  $2^n \times 2^n$ ,即像素总数为  $N_f = 4^n$ ,且该图像模式用线性四元树表示的节点数为  $N_{LQT}$ ,则称  $C_p(F) = N_{LQT} / N_f = N_{LQT} / 4^n$  为二值图像模式的复杂度,简记为  $C_p$ 。

二值图像模式的复杂度是参照线性四元树的块数来定义的,其目的主要是为了和线性四元树进行比较。

实验中用来测试的二值图像模式的分辨率参数  $n=8$ ,即  $2^8 \times 2^8$  的图像模式,如图 4 所示。这 6 幅图像的复杂度和纹理各不相同,具有较好的代表性,能够说明本方法的适应性。

我们比较了 TNAM 表示和线性四元树表示的实验结果。相应的实验数据如下表 1 和图 5 所示,其中, TNAM 是基于 TNAM 的二值图像表示方法;Quadtree 是基于线性四元树的二值图像表示方法; $\varphi_{LQT}$  是 Quadtree 的总数据量与 TNAM 的总数据量的比值。

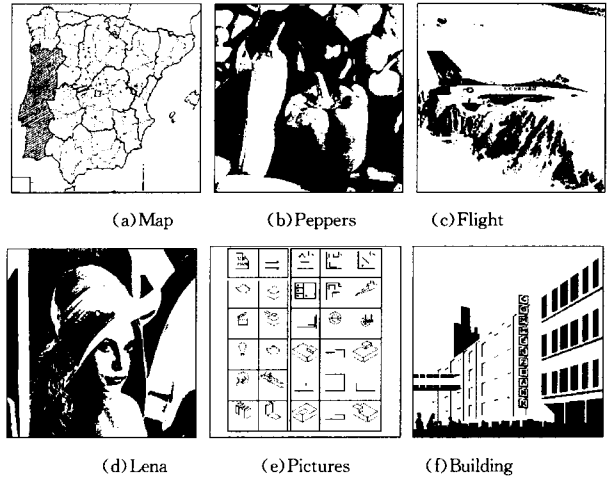


图 4 六幅测试图像

表 1 TNAM 和 Quadtree 编码的实验数据比较

图像	$C_p$	子模式或节点个数		$\varphi_{LQT}$
		Quadtree	TNAM	
Map	0.0882	5780	2804	3.96
Peppers	0.0856	5608	2994	2.99
Flight	0.0850	5568	2526	3.53
Lena	0.0981	6430	3414	2.99
Pictures	0.0835	5470	851	12.98
Building	0.0702	4602	1692	4.01

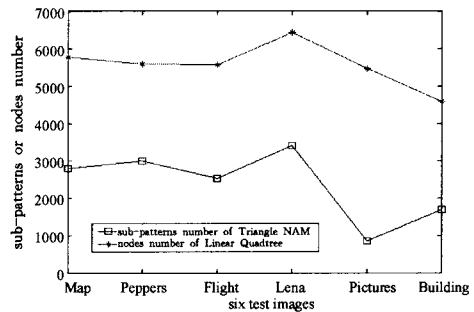


图 5 TNAM 子模式总数和 Quadtree 节点总数的对比图

表 1 中  $C_p$  是实验图像的复杂度,  $C_p$  是以四元树为基础计算的,从  $C_p$  的数字来看,图 4 (f) 的复杂度相对较低,图 4 (d) 的复杂度相对较高,其余图像的复杂度大致相同。各种方法的实验数据结构也基本证实的这一点。表 1 中给出了 TNAM 和 Quadtree 的子模式数(节点数)的数据,图 5 是 TNAM 子模式总数和四元树节点总数的对比图,从实验数据来看, TNAM 方法在数据量方面的效果都是非常明显的,其子模式数远小于 Quadtree 方法的节点数,对图像的适应性也非常好。

对于给定实验的 6 幅图像而言, Quadtree 的总数据量是 TNAM 的总数据量的 2.99 到 12.98 倍,尤其是对 Pictures 这幅图像,是 TNAM 的总数据量的 12.98 倍,这说明 TNAM 表示对线画类图像具有很强的表示能力,其余的 5 幅图像由于其纹理相对比较复杂一些,因此  $\varphi_{LQT}$  相对比较低,但

Quadtree的总数据量仍然是 TNAM 的 2 倍以上,效果还是十分明显的。显然,这些图像均证实了理论分析的结果,即当  $n=8$  时,Quadtree 的总数据量至少是 TNAM 的总数据量的 1.35 倍以上。

因此,与流行的基于线性四元树的二值图像表示方法相比,基于 TNAM 的二值图像表示方法能够更有效地减少数据存储空间,是二值图像的一种较好的表示方法。

**结束语** 以分层数据结构表示为背景,以模式表示方法为研究对象,借助于三角形 Packing 问题的思想,提出了一种新的基于 TNAM 的二值图像表示方法。理论分析和实验结果均表明:与流行的线性四元树表示方法相比,基于 TNAM 的二值图像表示方法能够更有效地减少数据存储空间,是二值图像的一种较好的表示方法。这种方法可以应用于二值图像模式表示的各个方面,在降低存储空间、提高传输速度、加快处理过程等方面具有良好的理论参考意义和实际应用价值。

### 参 考 文 献

[1] Cline D, Egbert K P. Terrain Decimation through Quadtree Morphing. *IEEE Trans. on Visualization and Computer Graphics*, 2001,7(1): 62-69

[2] Laguardia J J, Cueto E, Doblare M. A Natural Neighbour Galerkin Method with Quadtree Structure. *International Journal for Numerical Methods in Engineering*, 2005, 63 (6): 789-812

[3] Minglun G, Yee-Hong Y. Quadtree-based Genetic Algorithm and its Applications to Computer Vision. *Pattern Recognition*, 2004, 37 (8): 1723-1733

[4] 甘祥超,武栓虎,谈正.基于小波四叉树的图像去噪方法. *计算机研究与发展*,2001,38(3):341-343

[5] Voronin V. Holographic Representation in Image Processing Tasks. *Pattern Recognition and Image Analysis*, 2001,11(1): 265-267

[6] Liu Y, Ranganath S, Zhou X. Wavelet-based Image Segment Representation. *Electronics Letters*, 2002,38(19):1091-1092

[7] Monasse P, Guichard F. Fast Computation of a Contrast-invariant Image Representation. *IEEE Transactions on Image Processing*, 2000,9(5): 860-872

[8] Klinger A. Data Structure and Pattern Recognition//*Proceeding of IJCPR*. Washington, DC, USA, 1973:497-498

[9] Samet H. Region Representation: Quadtrees from Binary Arrays. *Computer Graphics and Image Processing*, 1980,13 (1): 88-93

[10] Gargantini I. An Effective Way to Represent Quadtrees. *Comm. ACM*,1982,25(12):905-910

[11] Wang C L, Wu S C,Chang Y K. Quadtree and Statistical Model-based Lossless Binary Image Compression Method. *Imaging Science Journal*, 2005,53(2):95-103

[12] Laszlo M, Mukherjee S. A Genetic Algorithm Using Hyperquadrees for Low-dimensional k-means Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006, 28 (4):533-543

[13] 冀俊峰,李胜,刘学慧,等.细节高度复杂表面模型的视点相关渐进传输. *软件学报*,2006,17(10):2192-2198

[14] Chen S K. An Exact Closed - form Formula for d - dimensional Quadtree Decomposition of Arbitrary Hyperrectangles. *IEEE Trans. Knowledge and Data Eng.*, 2006, 18(6):784-798

[15] 郑运平,陈传波.一种基于非对称逆布局模型的彩色图像表示方法研究. *软件学报*,2007,18(11):2932-2941

[16] Zheng Y P, Chen C B, Huang W, et al. A new non-symmetry and anti-packing model with k-lines and its application to color image representation // *Proceedings of the 2007 International Conference on Image Processing, Computer Vision, and Pattern Recognition (ICCV'07)*. Monte Carlo Resort, Las Vegas, Nevada, USA,2007:526-532

[17] Zheng Y P, Chen C B, Sarem M. A novel algorithm for triangle non-symmetry and anti-packing pattern representation model of gray images//*Proceedings of the Third International Conference on Intelligent Computing ICIC'07*, LNCS 4681. Qingdao, China, August 2007:832-841

(上接第 99 页)

阵由  $z \times z (12 \times 12)$  个分块矩阵组成,  $H_p$  矩阵第一个列块度数为 3,系统比特节点的度数均为 4。码长为 384 时,基于二次扩展的类 RA 码的围线 (girth) 长为 6,性能比 Mackay 随机码略好。码长为 1536 时,基于二次扩展的类 RA 码的围线 (girth) 长为 8,错误平层 (error floor) 较深,性能优于 Mackay 随机码,在误码率为  $1 \times 10^{-5}$  时,基于二次扩展的类 RA 码比 Mackay 随机码性能提高约 0.3dB。

**结束语** 本文提出了一种基于二次扩展的类 RA 码的构造方法,该方法设计的类 RA 码的码长和码率参数灵活可变,在 AWGN 信道下,与相同参数下的 Mackay 随机码性能相当。同时基于该码结构的特殊性,提出了编码复杂度与码长成线性关系的编码算法,该码具有准循环结构,节省校验矩阵存储空间,易于硬件实现,有望能成为能够实用的好码之一。

### 参 考 文 献

[1] Gallager R G. Low-density parity-check codes. *IRE Trans Inform Theory*,1962,IT-8:21-28

[2] Andrews K, Dolinar S, Thorpe J. Encoders for block-circulant LDPC codes//*Proc. IEEE ISIT*. New York; IEEE Press, 2005:

2300-2304

[3] Li Z, Kumar B. A class of good quasi-cyclic low-density parity check codes based on progressive edge growth graph//*Proc. IEEE SSC*. New York; IEEE Press,2004:1990-1994

[4] IEEE-SA Standards Board. IEEE Std. 802. 16e-2005 and IEEE Std 802. 16-2004/Cor 1-2005 IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. New York; IEEE, 2005

[5] Jin H. Analysis and design of Turbo-like codes[D]. California: California Institute of Technology,2001

[6] Liva G, Song S, Lan L, et al. Design of LDPC Codes; A Survey and New Results. *Journal of Communication Software and Systems*, 2006,2(3)

[7] Campello J, Modha D S, Rajagopalan S. Designing LDPC codes using bit-filling//*Proc. ICC 2001*. New York; IEEE Press,2001: 55-59

[8] Campello J, Modha D S. Extended bit-filling and LDPC code design// *Proc. IEEE Globecom 2001*. New York; IEEE Press, 2001:985-989