

# 改进的 NAMK 彩色图像表示算法

郑运平

(华南理工大学计算机科学与工程学院, 广州 510006)

**摘 要:** 利用格雷码可以扩展或保持二值图像块状性的特点, 提出一种改进的 NAMK 彩色图像表示算法 NAMKG, 给出算法的形式化描述及其存储结构, 对其总数据量进行分析。理论分析和实验结果表明, 相比 NAMK 算法和线性四元树算法, NAMKG 可以有效减少子模式数, 减小存储空间。

**关键词:** 图像表示; 格雷码; K-line 子模式; 非对称逆布局模型; 线性四元树

## Improved NAMK Color Images Representation Algorithm

ZHENG Yun-ping

(School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China)

**【Abstract】** By taking advantages of the characteristic of Gray code that can extend or remain the block character of binary images, this paper proposes an improved Non-symmetry and Anti-packing Model with K-lines(NAMK) representation algorithm for color images, which is called NAMKG algorithm. The description of NAMKG algorithm is presented and the storage structure and the total data amount of the algorithm are analyzed. Theoretical and experimental results show that, compared with NAMK and Linear Quadtree(LQT), NAMKG can effectively reduce the number of subpatterns and reduce the storage room.

**【Key words】** image representation; Gray code; K-line subpattern; non-symmetry and anti-packing model; Linear Quadtree(LQT)

DOI: 10.3969/j.issn.1000-3428.2011.02.070

### 1 概述

图像表示在图像压缩、特征提取、图像检索、图像去噪和图像复原等应用中起着非常关键的作用<sup>[1]</sup>。图像表示和图像运算一起组成图像模型, 是模式分析中的重要组成部分。图像的分层表示是一种基于树结构的表示方法<sup>[2]</sup>, 这种表示形式不仅在空间上更具紧凑性, 而且许多图像处理运算都可在分层表示的基础上快速实现。四元树表示是研究得最早且最多的一种分层表示形式。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, 文献[3]舍弃了指针方案, 提出了线性四元树(Linear Quadtree, LQT)表示方法。借助于 Packing 问题的思想, 文献[4]提出了一种基于矩形子模式的非对称逆布局模型(Non-symmetry and Anti-packing Model, NAM)的彩色图像表示方法, 它是彩色图像的一种无损表示方法。而文献[5]提出了一种基于 K-line 子模式的彩色图像表示算法 NAMK (Non-symmetry and Anti-packing Model with K-lines)。以上 2 种方法均采用了位平面分解(BPD)方法来降低图像复杂度, 取得了较高的表示效率, 但直接采用 BPD 方法分解位平面有一个缺点: 像素点灰度值的微小变化会对位平面的复杂度产生较明显的影响。格雷码则可以避免这一缺点。因为格雷码在相邻位间转换时, 只有一位产生变化, 所以如果用格雷码来表示图像中所有像素的灰度值, 则有利于扩展或保持二值图像的块状性。借助于格雷码的这一特点, 本文提出了一种改进的 NAMK 彩色图像表示算法 NAMKG。理论分析和实验结果表明, 相比 NAMK 算法和线性四元树算法, NAMKG 可以显著减少子模式数和存储空间。

### 2 NAMKG 算法描述

#### 2.1 K-line 的定义

有关 K 码的详细描述可参考文献[4]。图 1 是 K 码走向示意图, 其形象地表示了 K 码的变化过程。K 码走向的折线段

称为 K-line<sup>[5]</sup>。

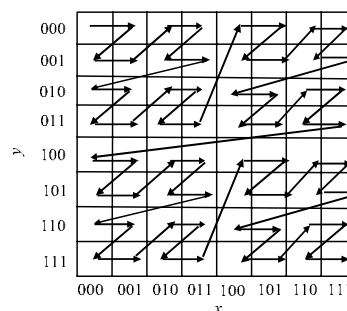


图 1 K 码示意图

#### 2.2 NAMK 算法示例

为了更好地理解 NAMKG 算法, 下面以一幅  $2^3 \times 2^3$  大小的二值图像为例(如图 2(a)所示)简单介绍 NAMK 算法<sup>[5]</sup>。不失一般性, 假定图 2 中的白点和黑点分别表示白色像素和黑色像素, NAMK 算法的逆布局结果用  $Q=\{(s, l)\}$  表示, 其中,  $s$  和  $l$  分别表示 K-line 子模式的起始点和游程长度。

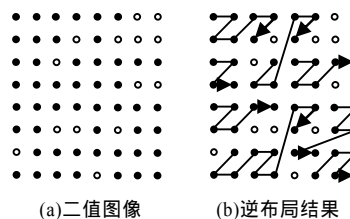


图 2 NAMK 算法示例

**基金项目:** 国家“863”计划基金资助项目(2006AA04Z211)

**作者简介:** 郑运平(1979 - ), 男, 讲师、博士, 主研方向: 图形图像处理

**收稿日期:** 2010-06-14

**E-mail:** zypdragon@163.com

通过对图 2(a)用 NAMK 算法进行逆布局, 不难获得如图 2(b)所示的逆布局结果:  $Q=\{(0, 7), (8, 4), (13, 6), (24, 6), (32, 6), (41, 10), (52, 6), (59, 5)\}$ 。从集合  $Q$  易知, 只需 8 个子模式即可表示图 2(a)。

### 2.3 NAMKG 算法步骤

NAMKG 算法中预先定义的子模式为 K-line 子模式, 用符号  $p=\{k|k=(s, l)\}$  表示。设已经布局好的彩色图像  $F$  的大小为  $2^n \times 2^n \times 3$ , 灰度级参数为  $m$ , 其分解后的 3 幅灰度图像为  $G[1]$ 、 $G[2]$  和  $G[3]$ , 大小均为  $2^n \times 2^n$ 。在二值图像中, 假定“0”为“black”, 即黑色, “1”为“white”, 即白色。NAMKG 算法只需记录“black”像素点。编码算法的具体步骤如下:

**Input** 一幅  $2^n \times 2^n \times 3$  的彩色图像  $F$  及其灰度级参数  $m$

**Output** 编码结果  $E=\{E_0, E_1, \dots, E_{3m-1}\}$ , 其中,  $E_i (0 \leq i < 3m-1)$  表示第  $i$  个格雷码位面图的编码结果

**Step1** 分别取得由  $F$  的  $r$ 、 $g$ 、 $b$  颜色分量组成的大小为  $2^n \times 2^n$  的灰度图像  $G[1]$ 、 $G[2]$  和  $G[3]$ , 并把 K-line 子模式的计数变量  $h$  赋值为 0。

**Step2** 用灰度图像的 BPD 方法依次将 3 幅灰度图像  $G[1]$ 、 $G[2]$  和  $G[3]$  分别分解为  $m$  幅二值位面图  $BP_i (0 \leq i < m-1)$ 、 $BP_i (m \leq i < 2m-1)$  和  $BP_i (2m \leq i < 3m-1)$ 。

**Step3** 当  $k=1, 2, 3$  时, 根据如下公式依次计算出每  $m$  幅二值位面图  $BP_i (0 \leq i < m-1)$ 、 $BP_i (m \leq i < 2m-1)$  和  $BP_i (2m \leq i < 3m-1)$  所分别对应的  $m$  幅格雷码位面图  $AP_i (0 \leq i < m-1)$ 、 $AP_i (m \leq i < 2m-1)$  和  $AP_i (2m \leq i < 3m-1)$ :

$$AP_i = BP_i, i = k \times m - 1$$

$$AP_i = BP_i \oplus BP_{i+1}, (k-1) \times m \leq i < k \times m - 2$$

并令  $j=0$ 。

**Step4** 按 K 码扫描的顺序, 从  $AP_j$  的第 1 个入口开始, 先确定一个未被标识的 K-line 子模式的起始点  $(x, y)$ , 再根据 K-line 子模式的匹配算法来追迹相应的 K-line 子模式。

**Step5** 根据 K-line 子模式的效率尺度(即 K-line 子模式的游程长度)确定一个游程最长的 K-line 子模式, 并将这个子模式在  $AP_j$  中进行标识, 以便寻找下一个起始点。

**Step6** 令  $h=h+1$ , 并记录此最大 K-line 子模式的 2 个参数, 即起始点坐标  $(x, y)$  和游程长度  $l$ , 然后对起始点坐标  $(x, y)$  做 K 码降维变换, 即  $s \leftarrow K(x, y)$ 。最后将  $s$  和  $l$  存储到队列  $E_j$  中, 即  $E_j\{h\} \leftarrow \{s, l\}$ 。

**Step7** 循环执行 Step4~Step6, 直到不能形成新的 K-line 子模式为止。

**Step8**  $j=j+1$ 。如果  $j \geq 3m-1$ , 则执行 Step4。

**Step9** 输出编码结果  $E=\{E_0, E_1, \dots, E_{3m-1}\}$ 。

### 2.4 NAMKG 的复杂度分析

假定彩色图像  $F$  中元素的总数为  $N$ , 灰度级参数为  $m$ 。由于一幅彩色图像被分解为  $3m$  幅二值图像来处理, 因此对 NAMKG 算法来说, 编码所需的时间正比于  $mN\mu$ , 其中,  $\mu$  表示二值图像中每个像素平均分割的次数, 且  $\mu$  的上限为  $O(\log N)$ 。因此, 在最坏情况下, 编码算法时间复杂度为  $O(mN \log N)$ 。在空间开销方面, 编码算法除  $3m$  幅二值图像矩阵外只增加了很少的中间变量, 因此, 空间复杂度与  $3m$  幅二值图像的大小呈正比, 即  $O(mN)$ 。

### 3 NAMKG 的数据量分析

在分析算法的数据量前, 先给出如下定理:

**定理** 对于给定的灰度级参数为  $m$  的灰度图像  $G$ , 经 BPD 方法处理后, 设格雷码位面图  $AP_i$  和二值位面图  $BP_i$  逆布局

后的 K-line 子模式数为  $M(i)$  和  $N(i)$ , 则  $\sum_{i=0}^{m-1} M(i) < \sum_{i=0}^{m-1} N(i)$ , 其中,  $0 \leq i < m-1$ 。

证明: 由格雷码编码公式可知: (1) 当  $0 \leq i < m-2$  时,  $AP_i = BP_i \oplus BP_{i+1}$ 。由格雷码的特点可知, 2 个相邻像素的格雷码只有 1 个比特位的区别。因此, 如果用格雷码表示图像中所有像素的灰度值, 则像素点灰度值的小幅变化不会影响所有的位平面, 从而有利于扩展  $AP_i$  中图像的块状性, 使得对  $AP_i$  进行逆布局得到的 K-line 子模式数少于对  $BP_i$  进行逆布局所得的 K-line 子模式总数, 即  $M(i) < N(i)$ 。(2) 当  $i=m-1$  时,  $AP_i = BP_i$ , 因此,  $AP_i$  中图像的块状性与  $BP_i$  中图像的块状性是一样的, 即  $AP_i$  能保持  $BP_i$  的块状结构, 因此,  $M(i)=N(i)$ 。综上所述, 当  $0 \leq i < m-1$  时,  $AP_i$  能保持或扩展  $BP_i$  的块状性, 即  $\sum_{i=0}^{m-1} M(i) < \sum_{i=0}^{m-1} N(i)$ 。定理得证。

设给定的彩色图像  $F$  (灰度级参数为  $m$ ) 的大小为  $2^n \times 2^n \times 3$ 。令第  $i$  个色彩分量的第  $j$  个格雷码位面图逆布局后的 K-line 子模式总数为  $M(i, j)$ , 其中,  $1 \leq i \leq 3$  且  $0 \leq j < m-1$ 。

对于 NAMKG 表示方法来说, 其子模式的存储结构与文献[5]的存储结构是一样的, 即对大小为  $2^n \times 2^n$  的二值图像来说, 存储一个 K-line 子模式占  $2n$  位。因此,  $F$  用 NAMKG 表示方法逆布局后的  $3m$  幅格雷码位面图的总数据量  $T$  为:

$$T = \sum_{i=1}^3 \sum_{j=0}^{m-1} 2nM(i, j) = 2n \sum_{i=1}^3 \sum_{j=0}^{m-1} M(i, j) \quad (1)$$

由文献[5]可知,  $F$  用 NAMK 表示方法逆布局后的  $3m$  幅二值位面图的总数据量  $S$  为:

$$S = \sum_{i=1}^3 \sum_{j=0}^{m-1} 2nN(i, j) = 2n \sum_{i=1}^3 \sum_{j=0}^{m-1} N(i, j) \quad (2)$$

其中,  $N(i, j)$  表示第  $i$  个色彩分量的第  $j$  个二值位面图逆布局后的 K-line 子模式总数 ( $1 \leq i \leq 3, 0 \leq j < m-1$ )。

设  $\psi$  为 NAMK 和 NAMKG 的总数据量之比, 则有:

$$\psi = \frac{2n \sum_{i=1}^3 \sum_{j=0}^{m-1} N(i, j)}{2n \sum_{i=1}^3 \sum_{j=0}^{m-1} M(i, j)} = \frac{\sum_{i=1}^3 \sum_{j=0}^{m-1} N(i, j)}{\sum_{i=1}^3 \sum_{j=0}^{m-1} M(i, j)} \quad (3)$$

根据定理 1, 由式(3)不难推出  $\psi > 1$ 。

### 4 实验结果及分析

图 3 是 4 幅大小为  $2^9 \times 2^9 \times 3$  的实验用彩色图像, 这些图像分辨率参数  $n=9$ , 灰度级参数  $m=8$ , 即  $2^8=256$  级。表 1 给出了 LQT 算法、NAMK 算法和 NAMKG 算法的性能测试结果, 其中,  $N$  为子模式或节点个数;  $\lambda$  为 LQT 与 NAMK 的总数据量之比;  $\omega$  为 LQT 与 NAMKG 的总数据量之比;  $\psi$  为 NAMK 与 NAMKG 的总数据量之比。

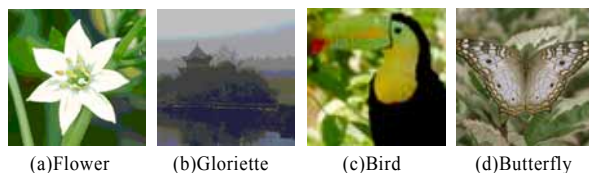


图 3 测试图像

表 1 NAMKG、NAMK 和 LQT 的编码性能比较

彩色图像	$N$			$\lambda$	$\omega$	$\psi$
	LQT	NAMK	NAMKG			
Flower	153 621	87 893	72 090	3.301 5	3.937 9	1.192 8
Gloriette	174 309	71 993	60 108	4.573 3	5.419 8	1.185 1
Bird	607 317	671 779	533 065	1.707 6	2.122 5	1.243 0
Butterfly	699 519	668 865	512 421	1.975 8	2.459 1	1.244 6

(下转第 209 页)