

颜色不变量的自适应聚类网络量化方法

李向阳 杨树元

(中国科学院声学所数字系统集成部, 北京 100080)

摘 要 给出了颜色不变量的自适应聚类网络量化算法。这种方法采用一组图象训练量化矢量, 根据训练图象能自适应地影响量化矢量, 把这种算法和均匀量化算法应用于CBIR系统中, 并对它们的检索结果和时间复杂度进行比较, 结果表明, 该算法在检索的正确率和时间的复杂度上均优于均匀量化方法, 因而颜色不变量的自适应聚类网络量化方法是一种很好的矢量量化算法。

关键词 基于内容的图象检索 颜色不变量 自适应聚类网络量化 聚类 均匀量化

中图法分类号: TN919.8 TP391.4 **文献标识码:** A **文章编号:** 1006-8961(2002)02-0120-05

Adaptive Cluster Network Quantification for Color Invariants

LI Xiang-yang, YANG Shu-yuan

(Institute of Acoustics, Chinese Academy of Sciences, Beijing 100080)

Abstract We aim at finding a suitable quantification algorithm to encode color invariant for indexing and retrieving images. To this end, an adaptive cluster network quantification algorithm for color invariants is proposed. By using a set of training images to train the color invariant vector, this algorithm can acquire a suitable color invariants vector having an adaptive number of members. In this paper, we discuss in detail how the threshold and the step in the algorithm influence the number of the vector members. After having done many experiments, we get a vector of 29 members for our training images when the threshold is 1.0 and the step is 0.3. In this setting, the vector is rather robust. Then we also apply the algorithm and another bench algorithm, named averaging quantification algorithm, to a content-based image retrieval system. Experiments have been conducted on a database consisting of 1 126 images taken from different image databases. In order to evaluate and compare the querying results, an application-specific software is developed. In the view of the correctness of the querying results, comparing the adaptive quantification algorithm with the averaging quantification algorithm, we find the former is superior to the latter by 4%. In the view of the time complexity, although the former takes a long time to train quantification vector and to acquire a lookup table for the image database, it is much superior to the latter when retrieval proceeds. Finally a conclusion is obviously obtained that adaptive quantification algorithm is an excellent quantification algorithm for color invariants.

Keywords CBIR, Color invariants, Adaptive cluster network quantification, Cluster, Averaging quantification

0 引 言

在基于内容的图象检索(Content-based image retrieval, 简称CBIR)系统中, 采用QBE(Query by example)模式检索时, 其目的是在图象库中找到包含有与检索图象相似的目标图象^[1]。目标图象通过颜色、纹理、边缘、形状等特征与检索图象相似^[1,2], 或者

通过区域分割匹配^[3], 达到在内容上与检索图象相似的目的。自从Swain和Ballard发表文献[4]后, 出现了许多以颜色为特征量的CBIR系统, 如QBIC、Photobook等。在这些系统中, 进行图象检索分为两个步骤。第1步是生成索引矢量库, 它们大多直接以颜色量为检索特征量, 对颜色量进行矢量量化, 生成量化表, 然后根据量化表为图象库中的每一个图象生成直方图, 并称这些直方图为索引矢量, 把索引矢量

基金项目: 海外青年基金课题(A09ZR01)

收稿日期: 2000-10-23; 成图日期: 2001-04-17

组织在一起形成索引矢量库;第 2 步是在图象库中检索输入图象,根据量化表,为输入图象生成一个输入矢量(也叫检索矢量),按照某一匹配准则在索引矢量库中找出与检索矢量最相似的若干矢量。

在理想的图象检索过程中,不论进行图象处理的环境如何发生变化,对于从同一物体中获取来的图象,其直方图应该保持一致。为了尽可能达到这一目的,可以将颜色量 RGB 进行预处理。例如文献 [1]采用式(1)对颜色量进行处理,将颜色量从 RGB 空间转换到 $C_4C_5C_6$ 空间。

$$\begin{cases} c_4(R, G, B) = \frac{R - G}{R + G} \\ c_5(R, G, B) = \frac{R - B}{R + B} \\ c_6(R, G, B) = \frac{B - G}{B + G} \end{cases} \quad (1)$$

其中, $-1 \leq c_i \leq +1, i=4, 5, 6$ 。因为 c_i 不随入射光的角度和强度等环境因素的变化而变化,故称 c_i 为颜色不变量,称 $C_4C_5C_6$ 空间为颜色不变量空间。

颜色不变量是一种不同于颜色量的新特征量,其矢量量化方法也应有其特殊性。文献 [1] 简单地将 $C_4C_5C_6$ 空间均匀量化为 64 等份,并据此生成索引矢量库。这种方法的优点是算法简单,检索结果能基本反映图象前景和背景的颜色,对图象内容有一定的反映,但是该方法主观地将 $C_4C_5C_6$ 空间均匀量化为 64 等份,没有理论依据。

本文提出一种新的颜色不变量量化方法,该方法利用了自适应聚类网络,在量化过程中能根据训练图象中颜色不变量的分布情况,自动修改量化表,增加量化表表项,以适应颜色不变量的实际分布情况。本文给出了根据两种量化方法得到的检索实验结果,从检索结果的正确率和时间复杂度两方面作了比较。实验结果表明,采用自适应聚类网络量化方法的检索结果正确率比采用均匀量化的正确率高 4%,而实时检索时间几乎相同。

1 颜色不变量的自适应聚类网络及其矢量量化算法

1.1 自适应聚类网络

本算法采用的自适应聚类网络^[5]结构如图 1 所示。

在图 1 中,对训练图象中任一点 $x_i(R, G, B)$,其对应颜色不变量 $I_i(C_4, C_5, C_6)$, $I_i(C_4, C_5, C_6)$ 对应的归一化形式为 $J_i(C_4, C_5, C_6)$; 输出为 $O_i(C_4, C_5,$

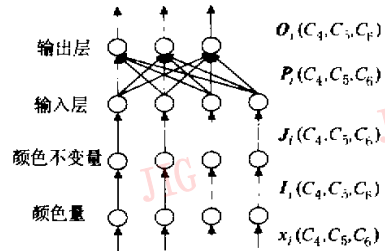


图 1 自适应聚类网络

C_6),它就是所需的颜色不变量量化表,量化表的每一表项都属于 $C_4C_5C_6$ 空间,其对应的归一化表示形式为 $P_i(C_4, C_5, C_6)$ 。整个自适应聚类网络由颜色量层、颜色不变量层、输入层、输出层组成。颜色量层将颜色量转换成颜色不变量;颜色不变量层则将颜色不变量标准化。输入层和输出层是本网络的核心。这两层是全连接的。输入层的输入为归一化的颜色不变量。到输出层的输入采用下式计算

$$I_k = P_k^T(C_4, C_5, C_6) J_i(C_4, C_5, C_6) \quad k=1, 2, \dots, c \quad (2)$$

式中, $P_k^T(C_4, C_5, C_6)$ 为第 k 个神经元的权值矢量,也是量化表的第 k 表项的归一化的转置; c 为当前量化表的表项数。应用“胜者取全部”的原则,获胜的输出层神经元将是具有最大的输入 I_k 的神经元。获胜神经元的输出根据情况分别处理。

1.2 对获胜神经元的处理

在以上的获胜规则中,由于只考虑了输入矢量与权值矢量之间的夹角,而忽视了它们的模,因而获胜神经元对应的量化表表项的模与输入的颜色不变量的模可能相差很大,未获胜神经元对应的量化表表项的模与输入的颜色不变量的模可能相差很小。从这种意义上来说,获胜神经元的权值矢量不一定与输入矢量最相似。

采用这种学习方法,常常出现这样一种情况,如果在相隔较远的两个时间点上两次输入同样的样本,而在此期间又输入其他样本,那么将出现第 2 次与第 1 次分类结果不一致的现象,这说明第 1 次分类后,新学习得到的记忆内容有可能冲掉原有的学习记忆内容,从而导致第 2 次分类错误。这也说明,简单的竞争学习规则不能保证记忆具有足够的牢固性。为了解决这一矛盾,增加以下测试方法。假设输入 $I_i(C_4, C_5, C_6)$ 后,获胜神经元为 m 。那么该不该把 $I_i(C_4, C_5, C_6)$ 归入到第 m 类,则需要进行测试。测试的方法是计算样本 $I_i(C_4, C_5, C_6)$ 与第 m 类中心 $O_m(C_4, C_5, C_6)$ 之间的距离,如果距离小于阈值,则将

$I_i(C_4, C_5, C_6)$ 归于第 m 类, 并修改聚类中心 $O_m(C_4, C_5, C_6)$; 否则再看看输入 $I_i(C_4, C_5, C_6)$ 是否可归入其他类. 如果 $I_i(C_4, C_5, C_6)$ 不适合归入现有的所有类, 则增加新类, 并增加新的输出节点. 这样网络的输入结构依样本的分布不同而不同.

1.3 算法描述

(1) 设初始类别数 $c=2$, 设定阈值 L 和步长 S . 初始化输出层神经元的输出矢量 $O_k(1)$ 和权值矢量 $P_k(1)$, 并令

$$P_k(1) = O_k(1) / \|O_k(1)\| \quad k = 1, 2 \quad (3)$$

(2) 输入颜色不变量 $I_i(C_4, C_5, C_6)$ 及其归一化形式 $J_i(C_4, C_5, C_6)$.

(3) 按照式(2)计算每个神经元的输入 I_k .

(4) 按照下式找出获胜神经元节点 m .

$$m = \max_{\substack{k=1, 2, \dots, c \\ k \text{ 不是已属节点}}} \{I_k\} \quad (4)$$

(5) 测试. 计算输入颜色不变量 $I_i(C_4, C_5, C_6)$ 与神经元输出矢量 $O_m(C_4, C_5, C_6)$ 的欧氏距离 s . 如果 $s < L$, 表示通过测试; 否则, 表示未通过测试. 并分别进行如下处理:

- ① 如果获胜节点通过测试, 则转第 7 步;
- ② 如果获胜节点没有通过测试, 则屏蔽该获胜节点, 返回第 4 步, 选择新的获胜节点;
- ③ 如果现有的 c 个节点都没有通过测试, 则转第 6 步;

(6) 令 $c=c+1$, 增加一个输出节点, 并初始化该节点的输出矢量和权值矢量, 转至第 8 步;

(7) 按照下式修改输出矢量和权值矢量

$$O_m(l+1) = O_m(l) + S \times (I_i - O_m(l)) \quad (5)$$

$$P_m(l+1) = O_m(l+1) / \|O_m(l+1)\| \quad (6)$$

(8) 如果还要学习, 则转第 2 步, 否则算法结束.

算法中, 阈值 L 对聚类结果影响很大. 通常, L 越小, 聚类类别越多; L 越大, 类别越少.

2 算法实现和分析

2.1 实验环境

实验在 P III 50e CPU, 128M 内存的 PC 兼容机上进行. 软件环境为 Visual Studio 6.0; 颜色不变量矢量量化算法、索引库形成算法和检索算法均用 C 语言编写; 检索结果评估软件采用 C++ 编写.

实验图象库选自武汉大学出版社发行的“时尚素材精美图库”和精选的图片, 包括人物、花、鸟、蝴

蝶、庙宇、自然景观、鱼等共 12 组 1 126 个图象. 训练图象随机取自实验图象库.

由于颜色不变量量化算法应用于 CBIR 系统, 因而图象检索结果的优劣是衡量量化算法的重要标准. 实验通过对使用颜色不变量的均匀量化和自适应聚类网络量化这两种不同量化算法的比较, 从检索结果的正确率和算法复杂度对算法进行评价.

2.2 矢量量化

根据本文算法, 实现了颜色不变量的矢量量化. 在这个算法中, 阈值和步长参数对量化结果影响很大.

当步长 $S=0.2$, 阈值 L 从 1.0 增长到 1.6 时, 对应的聚类数如图 2 所示.

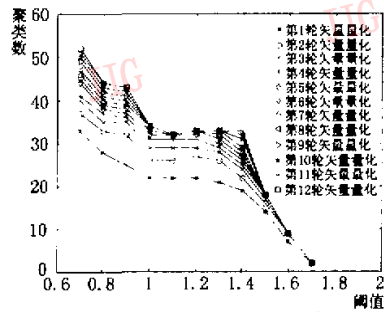


图 2 聚类数与阈值 L 间的关系

从图中可知, 阈值 L 对量化结果影响很大. 阈值越小, 聚类数越多; 阈值越大, 聚类数越少. 另外, 由于量化过程中聚类中心的不断改变, 用同样的训练图象进行多次训练时, 聚类数有所增加. 图 2 中画出了以同样训练图象进行 12 次训练得到的聚类数. 第 2 次比第 1 次平均多 18.9%. 继续进行训练, 增长的聚类数大幅度减少. 表 1 为步长 $S=0.2$, 阈值 $L=1.2$ 时, 多次训练得到的聚类数及增长率. 随着训练轮次的增加, 聚类数增大, 但增长率减小; 一般说来, 当训练轮次大于 8 后, 聚类数基本保持不变.

表 1 训练轮次与聚类数的关系

	1	2	3	4	5	6	7	8	9
聚类数	22	27	29	31	32	33	33	33	33
增长率(%)		22.7	6.8	6.9	3.2	3.1	0	0	0

从图 2 中可以看出, 当 $1.0 < L < 1.4$ 时, 各条曲线的斜率接近为 1, 也就是说, 当使用不同的分辨率时, 聚类数基本保持不变. L 越小, 系统具有越高的分辨率. 因此, 在后面的实验中, 选取 $L=1.0$. 当 $L > 1.4$ 时, 聚类数迅速减少; 当 $L=1.7$ 时, 聚类数为 2, 这时已不能正确量化了.

当阈值 L 分别为 1.0、1.2、1.4 时, 步长 S 从

0.1 增长到 1.0 所对应的第 3 轮聚类数如图 3 所示。从图中可以看出,聚类数随步长的变化而变化,随着步长的增大,聚类数也增大。当步长 $S < 0.3$ 时,聚类数变化幅度大;当步长 $0.3 < S < 0.5$ 时,聚类数保持相对稳定。为了使矢量量化算法具有良好的鲁棒性,选择 $L=1.0, S=0.3$ 进行后面的实验。

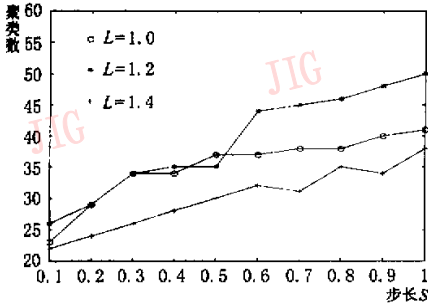


图 3 聚类数与步长 S 间的关系

2.3 索引及检索算法

选择 $L=1.0, S=0.3$, 并使用相同的图象库重复训练多次, 得到聚类数为 29 的颜色不变量量化矢量 I

$$I = I_0 I_1 \dots I_k \dots I_{28} \quad 0 \leq k \leq 28 \quad (7)$$

$$I_k = C_{4k} C_{5k} C_{6k}, \quad -1 \leq C_{jk} \leq 1, j = 4, 5, 6 \quad (8)$$

在式(7)、(8)中, I 表示颜色不变量量化矢量, 它由 29 个分量组成, 每个分量代表颜色不变量空间的一个点。

生成了颜色不变量量化矢量 I 后, 根据下式得到索引矢量库

$$H(k) \triangleq \eta (\min_{0 \leq i \leq 28} (d(x, I_i)) = k) / N \quad 0 \leq k \leq 28 \quad \forall x \in P \quad (9)$$

式中, \triangleq 表示定义为, d 表示欧氏距离, x 表示图象中的任意一点在颜色不变量空间的对应表示, \min 表示取 I_i 各分量中与点 x 距离最近的分量, η 表示 \min 等于 k 的总数, N 表示需要产生索引矢量的图象 P 的总象素点数。

从检索图象 Q 可以计算得到检索直方图 H_Q , 即检索矢量, 为了便于比较, 匹配准则按照归一化直方图相交来进行, 其原理可用下式描述

$$D(H_Q, H_{B_i}) = \sum_{k=0}^{28} \min\{H_Q(k), H_{B_i}(k)\} \quad (10)$$

式中, $H_Q(k)$ 表示检索矢量 H_Q 的第 k 分量, H_{B_i} 表示与图象库中图象 B_i 对应的索引矢量的第 k 分量。这样, 当 i 变化时, 可以计算出对应的 $D(H_Q, H_{B_i})$, 这个值代表了检索图象 Q 与图象库中图象 B_i 的匹配程度, 且 $D(H_Q, H_{B_i})$ 越大, 匹配越好。

3 算法的评价

3.1 从检索结果的准确度评价算法

为了评估检索质量, 设计了检索结果评估软件, 其界面如图 4 所示。

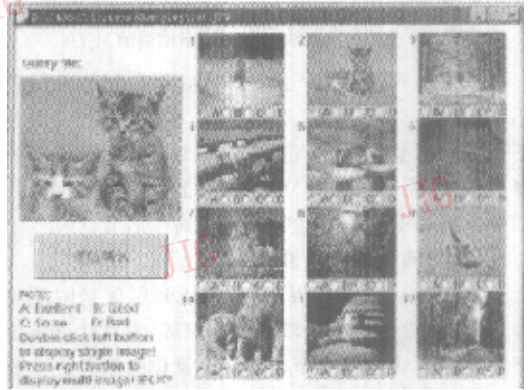


图 4 检索结果评估软件界面

图 4 中, 左图为检索图象, 右边为检索结果, 检索结果按匹配程度的好坏从 1 到 12 排列。

由多名评判员对检索结果打分, A 表示优秀, 为 100 分, B 表示一般, 为 70 分, C 表示较差, 为 40 分; D 表示很差, 为 0 分。将多名评判员的评判结果按照下式进行平均

$$M_k(r) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} S_{ij} / (N_1 \times N_2) \quad k = 1, 2 \quad (11)$$

$$S_k(r) = \sum_{j=r}^r M_i(j) / r \quad k = 1, 2 \quad (12)$$

式中, N_1 为评判员人数, N_2 为参加评分的检索图象的个数, S_{ij} 为第 i 名评判员给第 j 个图象的评分, r 表示检索结果图象的匹配等级, $M_k(r)$ 表示在第 k 种方法中匹配等级为 r 的检索结果图象的平均得分, $S_k(r)$ 表示在第 k 种方法中匹配等级小于等于 r 的所有检索结果图象的平均得分。 k 为 1, 对应于利用自适应聚类网络量化方法将颜色不变量空间量化成 29 份的检索结果, k 为 2, 对应于将颜色不变量空间均匀量化成 64 份的检索结果。 $M_k(r)$ 的统计结果如表 2 所示。 $S_k(r)$ 的统计结果如图 5 所示。

表 2 检索结果 $M_k(r)$ 评估对比

	r											
	1	2	3	4	5	6	7	8	9	10	11	12
$M_1(r)$	77	58	47	42	40	39	39	37	37	30	37	38
$M_2(r)$	73	49	48	32	33	32	34	30	32	37	35	38

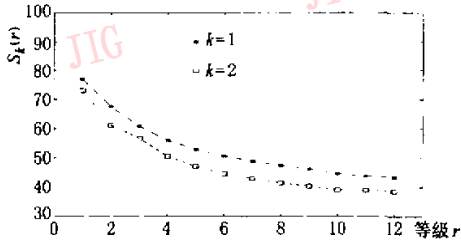


图5 两种量化方法对应的检索结果的对比

从图5可以看出,虽然自适应聚类网络将颜色不变量空间只量化成29份,而均匀量化方法将颜色不变量空间量化成64份,但前者的量化数远小于后者,而从检索结果的评分看,前者的前 r 个匹配等级的检索结果图象的平均得分明显高于后者。

3.2 从时间复杂度评价算法

均匀量化不需要训练,且生成索引矢量时间也很短;自适应量化不仅需要多次训练,而且计算直方图时还需要计算每一个象素点与量化矢量每一个分量之间的距离,这就需要较多的时间。表3列出了两种方法生成1126幅图象索引矢量库所用的总时间及每幅图象的平均时间。

表3 两种方法生成索引矢量库所用总时间及平均时间的比较

量化方法	总时间(s)	平均时间(s)
自适应量化	2 550	2.3
均匀量化	464	0.4

从表3可知,用自适应量化方法生成图象索引矢量的平均时间为2.3s,其虽然比均匀量化所用的时间长,但是,它产生图象索引矢量库的工作是一次性的,在进行图象检索时,不需要计算图象库图象的索引矢量。

在检索图象时,直接使用检索图象的检索矢量作为输入矢量。由于自适应聚类网络量化方法的量化数为29,比均匀量化的量化数(为64)少,因而在检索图象时,检索矢量与索引矢量库比较时间就短,检索时间也短。当图象库很大时,这一优点将表现得更加明显。表4为在有1126个图象的图象库中用两种算法连续检索4幅大小为 600×399 的图象所用的时间及平均时间。

表4 两种算法的时间复杂度比较

	图象				平均
	1	2	3	4	
自适应量化	4.7	3.3	3.3	3.3	3.7
均匀量化	5.0	4.6	4.6	4.6	4.7

4 结论

从以上的讨论可知,在基于内容的图象检索中,将图象从RGB转换到颜色不变量空间是一种很好的特征抽取方法;颜色不变量的均匀量化方法是一种简单、有效的方法;而颜色不变量的自适应聚类网络量化方法虽然在产生索引矢量库时增加了时间复杂度,但在图象检索时,它在检索结果的正确率和时间复杂度方面却得到了明显改善,因此这种方法是一种很有研究价值和发展前景的图象检索方法。

本文实验中,只使用了颜色不变量这一特征量,如果再加上纹理、梯度等特征,检索的正确率将会更高。而颜色不变量的纹理和梯度的提取,正是今后将要研究的内容。

本研究的后续工作还有:选取不同的阈值和步长来求得具有最优检索质量和时间复杂度的量化矢量;选择不同神经元获胜准则;寻找索引矢量生成的快速算法等。如果这些工作能完成,那么自适应聚类网络量化方法将更加成熟和完整。

参考文献

- 1 Gevers T, Smeulders A W M. PicToSeek: Combining color and shape invariant feature for image retrieval[J]. IEEE Trans. Image Processing, 2000,9(1):102~117.
- 2 Cox J J, Miller M L *et al.* The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments[J]. IEEE Trans. Image Processing, 2000,9(1):21~37.
- 3 F Chiou-Shann, C Shun-Wen, Kai E. Hierarchical color image region segmentation for Content-Based Image Retrieval system [J]. IEEE Trans. Image Processing, 2000,9(1):102~117.
- 4 Swain M J, Ballard D H. Color indexing[J]. Int. J. Comput. Vis., 1991,7(1):11~32.
- 5 Pandya A S, Macy R B. Pattern recognition with neural networks in C++ [M]. CRC Press, Inc., 1996,165~223.



李向阳 1966年生,现为中国科学院声学研究所信号与信息处理专业博士研究生。研究方向为数字图象处理、基于内容的图象检索、大规模并行处理及VLSI信号处理。



杨树元 1945年生,1968年毕业于哈尔滨军事工程学院,现为中国科学院声学研究所研究员,博士生导师,数字信号处理研究室主任。研究方向为模式识别、数字图象处理、大规模并行处理及VLSI信号处理。主要研究领域为模式识别。