

一种改进的 HSV 颜色空间量化方法及其应用

曹建芳^{1,2}, 陈俊杰², 赵青杉¹

(1. 忻州师范学院计算机科学与技术系, 山西 忻州 034000)
(2. 太原理工大学计算机科学与技术学院, 山西 太原 030024)

[摘要] 随着图像数据的海量增长, 图像检索效率逐渐成为研究热点. 为了提高图像检索的准确率, 提出了一种改进的 HSV 颜色空间量化方法, 细化色调 H 的分类, 使量化结果更接近人类感知, 并在此基础上采用分块策略进行仿真实验. 实验结果表明, 提出的方法能更好地描述图像的颜色特征, 效果令人满意, 具有一定的实用性.

[关键词] 图像检索, 颜色特征, HSV 颜色空间, 量化处理

[中图分类号] TP391 [文献标志码] A [文章编号] 1672-1292(2014)02-0068-06

An Improved Method on Color Space Quantization and Application

Cao Jianfang^{1,2}, Chen Junjie², Zhao Qingshan¹

(1. Department of Computer Science and Technology, Xinzhou Teachers University, Xinzhou 034000, China)
(2. College of Computer Science and Technology, Taiyuan University of Technology, Taiyuan 030024, China)

Abstract: With the massive growth of images, the efficiency of image retrieval gradually becomes a research hotspot. In order to improve accuracy of the image retrieval, Hue-Saturation-Value (HSV) color space is improved, which details the classification of H hue and makes the quantitative results more close to human perception. On this basis, block color histogram is extracted as a retrieval feature. Experiments show that the proposed method can better describe color feature of images, the results are satisfactory and have some practical value.

Key words: image retrieval, color feature, Hue-Saturation-Value color space, quantification

随着数码设备、网络及多媒体技术的发展, 人们在工作、学习和日常生活中不断地产生大量的图像数据. 因为数字图像呈现了爆炸式的增长方式, 如何合理而又高效地组织海量的图像数据、结合图像低层特征, 将数字图像进行分类和检索是目前的一个研究热点. 在图像的形状、颜色、纹理等特征中, 颜色特征是最显著、最可靠、最稳定的视觉特征, 相对于其他视觉特征而言, 颜色特征对图像本身的尺寸、方向、视角、平移、旋转等的依赖性较小, 具有较强的鲁棒性, 同时, 在许多情况下, 颜色又是描述一幅图像最简便而有效的特征^[1-4]. 人们对于一幅图像的印象, 往往从图像中颜色的空间分布开始, 并非所有的颜色空间都与人的感觉相一致, 因此基于颜色特征的图像检索方法, 关键是选择一个合适的颜色空间和准确的量化方法.

本文对 HSV 颜色空间的量化方法进行了改进, 在颜色量化时细化了色调 H 的分类、增加了量化维数, 使量化结果更接近人类感知, 然后在此基础上采用分块直方图方法提取图像的颜色特征进行了实验. 实验结果表明, 基于改进的 HSV 颜色空间量化的分块颜色直方图法要好于一般的颜色直方图法, 检索精度大大提高.

1 传统的 HSV 颜色空间量化方法

1.1 HSV 颜色空间

HSV 颜色空间^[5]是一种由人类感知来描述颜色的空间. 它直接对应于人眼颜色视觉特征的三要素:

收稿日期: 2014-02-10.

基金项目: 国家自然科学基金(61202163)、山西省自然科学基金(2013011017-2)、山西省高校科技创新项目(2013150)、忻州师范学院重点学科专项课题(XK201308).

通讯联系人: 曹建芳, 博士研究生, 副教授, 研究方向: 情感计算、数字图像理解等. E-mail: kexdj122@126.com

色调(H)、饱和度(S)和亮度(V). 色调(Hue)是指红、绿、蓝、紫等基本色;饱和度(Saturation)是指颜色的纯度,鲜红色的饱和度高,而粉红色的饱和度低;而亮度(Value)就是光的强度. HSV 颜色空间可以描述成一个倒置的圆锥体,绕圆锥中心轴的所夹角度被定义为色调,取值范围为 $0^\circ \sim 360^\circ$. 从圆锥的横截面的圆心到横截面圆周点的距离被定义为饱和度,取值范围从 0 到 1. 从圆锥的横截面的圆心到圆锥顶点的距离被定义为亮度. 如图 1 所示.

HSV 颜色空间直接对应于人眼色彩视觉特征的三要素,在图像检索中,应用这种颜色空间模型会更适合用户的视觉判断. 因此,HSV 颜色空间成为了最常用的颜色空间.

1.2 传统的量化方法

有很多研究采用以下的量化方法^[3,4]:先将 H 、 S 、 V 3 个分量按照人的颜色感知进行非等间隔的量化,然后再计算图像的全局颜色直方图,并据此特征进行图像检索实验.

(1) 按照人的视觉分辨能力,把色调 H 空间分成 8 份,饱和度 S 和亮度 V 空间分别分成 3 份.

$$H = \begin{cases} 0, & \text{if } h \in [316, 20]; \\ 1, & \text{if } h \in [21, 40]; \\ 2, & \text{if } h \in [41, 75]; \\ 3, & \text{if } h \in [76, 155]; \\ 4, & \text{if } h \in [156, 190]; \\ 5, & \text{if } h \in [191, 270]; \\ 6, & \text{if } h \in [271, 295]; \\ 7, & \text{if } h \in [296, 315]. \end{cases}$$

$$S = \begin{cases} 0, & \text{if } s \in [0, 0.2); \\ 1, & \text{if } s \in [0.2, 0.7); \\ 2, & \text{if } s \in [0.7, 1]. \end{cases}$$

$$V = \begin{cases} 0, & \text{if } v \in [0, 0.2); \\ 1, & \text{if } v \in [0.2, 0.7); \\ 2, & \text{if } v \in [0.7, 1]. \end{cases}$$

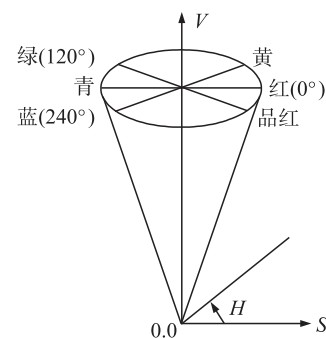


图 1 HSV 颜色空间模型

Fig.1 HSV color space model

(2) 根据色彩的不同范围和主观颜色感知进行量化.

(3) 根据光学理论,物体的颜色与光的波长和频率有关.不同的色光在真空中的波长和频率的范围不一样,因此可以将色调进行不等间隔量化.

(4) 构造一维特征矢量,按照以上的量化级,把 3 个颜色分量合成一维特征矢量: $L = 9H + 3S + V$.

这样, H 、 S 、 V 3 个分量在一维矢量上分布开来. 根据上述公式, L 的取值范围为 $[0, 1, \dots, 71]$, 计算 L 获得 72 柄(bin)的一维直方图.

2 改进的 HSV 颜色空间量化方法

2.1 处理方法

在 HSV 的颜色模型中,2 种颜色在颜色模型空间中的距离越短,那么这 2 种颜色就越相似,如果在 HSV 颜色模型的三维空间中,均匀地选取若干个量化点作为轴心点,然后将其他量化点按照颜色相似距离最短的原则聚类到这些轴心点上,只要轴心点尽可能均匀,就能达到在最低平均方差下聚类出最少数目的颜色的目的.

在实验中提取图像中每一个像素点的 RGB 值并转化为 HSV 颜色空间后,进行简单聚类如下:

if $0 \leq H < 200$, H :20 类,步距 10(0,10,……,200), H 取值 0 ~ 19;

if $200 \leq H < 300$, H :20 类,步距 5(200,205,……,300), H 取值 20 ~ 39;

if $300 \leq H < 360$, H :6 类,步距 10(300,310,……,360), H 取值 40 ~ 45;

S :4 类,步距 0.25(0.173,0.423,0.673,0.923), S 取值 0 ~ 3;

V :4 类,步距 0.25(0,0.25,0.5,0.75), V 取值 0 ~ 3.

从以上分类方法可以看出,将 H 分成了不等间隔的 46 类, S 分成了不等间隔的 4 类, V 分成了不等间

隔的4类.

一维特征矢量: $L=4H+3S+V$, L 的取值范围为 $[0,1,\dots,192]$, 计算 L 获得 193 柄(bin)的一维直方图.

由于在 HSV 颜色空间中 H 分量是表现颜色的最重要的分量, 因此, 采用本实验的量化降维方法不仅细化了 H 的分类, 而且增加了量化维数. 对于只依靠颜色特征的检索系统来说, 大大改善了检索效果.

2.2 量化降维实验依据

2.2.1 166 维量化

实验最初的量化处理采用以下方法:

H : 18 类, 步距 20 (0, 20, ..., 360), H 取值 0 ~ 17;

S : 4 类, 步距 0.25 (0.173, 0.423, 0.673, 0.923), S 取值 0 ~ 3;

V : 4 类, 步距 0.25 (0, 0.25, 0.5, 0.75), V 取值 0 ~ 3.

一维特征矢量: $L=9H+3S+V$, L 的取值范围为 $[0,1,\dots,165]$, 计算 L 获得 166 柄(bin)的一维直方图.

用二维数组 $L1$ 存储每个分类的像素个数, 采用平均分块策略, 将一幅图像平均分为 16 块, 编号为 $H1 \sim H16$, 如图 2 所示. 而且由于图像中的物体一般分布在图片的中央, 因此将中间的 $H6$ 、 $H7$ 、 $H10$ 、 $H11$ 作为重叠分块. 在提取颜色特征时, 从 $H1$ 到 $H16$ 依次提取. 分别存储这 16 个子块的颜色特征, 在经过处理之后用于相似度计算. 图像各个子块权重的选取应根据图像的特点决定, 通常图像中间子块的权重较大, 或者用户指定区域的权重较大, 以反映图像的位置信息.

因此, 本文中对图像中间部分加大权重. 分配方式如下: 可以看出图中中心区域 ($H6$ 、 $H7$ 、 $H10$ 、 $H11$) 占全图总面积的 $1/4$,

因此为中间区域加大权重到 $1/3$, 每个子块的权重为 $1/12$, 而周围 12 个子块的权重为 $1/18$.

以 Corel 图像库中的 690.jpg 为例, 每个子块 H 分类的结果如下: 数组中的每个数字代表该类中的像素点的个数.

$L1[0]$: 2 1 2 7 48 26 33 31 58 264 2760 1618 442 751 99 1 0 1

$L1[1]$: 4 0 1 3 29 10 24 16 31 138 1445 550 1000 2246 644 2 1 0

$L1[2]$: 2 1 0 3 41 68 85 53 127 812 3144 320 703 679 103 1 2 0

$L1[3]$: 0 1 0 4 2 3 2 8 14 102 842 785 3429 940 11 0 0 1

$L1[4]$: 0 0 0 0 28 12 11 10 23 111 1510 1266 2407 736 30 0 0 0

$L1[5]$: 3 2 2 6 23 17 7 7 23 86 764 404 1411 3112 271 3 1 2

$L1[6]$: 0 1 1 7 24 4 15 14 28 107 1137 1275 1499 1874 156 1 0 1

$L1[7]$: 2 2 0 8 91 24 32 36 72 228 2480 1471 758 772 167 1 0 0

$L1[8]$: 2 0 0 4 68 113 176 57 97 327 2829 1355 373 460 279 2 1 1

$L1[9]$: 7 0 3 9 52 49 62 35 61 198 1175 1178 831 2054 420 8 1 1

$L1[10]$: 3 2 1 5 56 34 60 59 112 548 3414 1095 353 339 62 0 0 1

$L1[11]$: 2 2 2 4 42 12 28 21 81 404 2825 1983 361 363 12 1 1 0

$L1[12]$: 2 1 4 18 99 32 37 41 59 179 1955 1208 408 1630 467 2 2 0

$L1[13]$: 1 1 2 8 21 27 23 33 69 322 2295 699 600 1921 115 3 2 2

$L1[14]$: 0 0 0 4 85 38 30 57 143 829 4878 80 0 0 0 0 0 0

$L1[15]$: 1 0 1 1 8 5 5 6 15 75 1354 399 1058 3015 196 4 0 1

由于标准 Corel 图像库中的图像大小都是 384×256 个像素点, 平均分成 16 块后, 每一块的像素点个数均为 6144. 从以上结果可以看出, H 的值比较集中, 甚至对于有些图像, 每一块只有一个分类中为 6144 个像素点, 其他全部为 0. 因此, 若都采用 20 为步距, 则整张图片的相似度可能只由像素点集中的少数 2 ~ 3 个分类决定, 结果非常不准确.

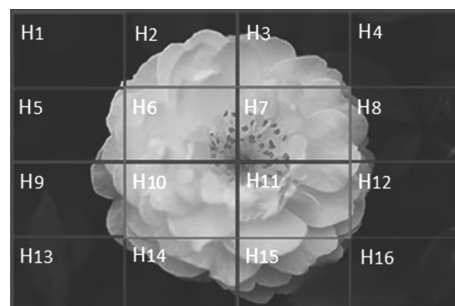


图2 平均分块策略

Fig.2 The average sub-block strategy

2.2.2 652 维量化

由于 166 维量化的效果很不好,将 H 的分类进行细化,设置步距为 5,这样可以将 H 分为 72 类. 一维特征矢量: $L=9H+3S+V$,这样将颜色特征量化到了 652 维. 图像 690. jpg 的实验结果如下:

```
L1[0]:2 0 0 0 0 0 1 0 1 0 1 0 1 2 4 0 8 4 24 12 12 5 5 4 18 3 7 5 4 2 7 18 18 9 7 24 17 55 61 131 276
503 802 1179 962 313 187 156 132 97 77 136 162 159 189 241 83 11 5 0 0 1 0 0 0 0 0 0 0 1 0 0
L1[1]:4 0 0 0 0 0 0 0 0 0 1 0 0 0 2 1 5 3 9 12 0 4 4 2 15 0 5 4 4 4 2 6 7 8 4 12 12 22 29 75 117 296 521
511 165 94 127 164 168 195 281 356 433 440 527 846 478 155 11 0 1 1 0 0 0 1 0 0 0 0 0 0
L1[2]:2 0 0 0 0 0 0 1 0 0 0 0 2 1 0 0 2 0 8 31 21 21 23 3 62 2 15 6 3 18 18 14 38 16 16 57 49 130 193
440 676 1058 899 511 108 53 89 70 92 102 199 310 267 161 127 124 87 15 1 0 0 0 1 0 0 1 1 0 0 0 0 0
L1[3]:0 0 0 0 0 0 1 0 0 0 0 0 0 0 2 2 1 0 0 1 0 1 1 1 1 0 1 0 2 3 3 0 5 0 3 6 11 11 21 59 130 260 250 202
187 140 169 289 479 786 1254 910 488 202 179 71 11 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
L1[4]:0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 13 8 4 4 4 0 8 1 0 2 0 0 4 6 6 5 5 7 9 29 26 47 87 176 372 875
434 184 228 420 594 728 655 430 212 174 239 111 25 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
L1[5]:3 0 0 0 0 1 0 1 1 0 0 1 0 1 3 2 6 4 8 5 9 4 1 3 4 2 0 1 0 2 1 4 3 6 8 6 11 20 19 36 84 214 275 191
101 72 81 150 236 287 362 526 553 632 929 998 217 44 8 2 0 1 2 0 0 1 0 0 0 1 0 1
L1[6]:0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 6 3 6 9 6 1 1 2 0 1 4 8 2 0 7 0 7 2 6 2 18 9 18 23 57 80 109 269 679
510 247 252 266 253 349 481 416 425 441 530 478 125 27 3 1 0 0 1 0 0 0 0 0 0 0 0 0 1
L1[7]:2 0 0 0 0 0 0 2 0 0 0 0 0 0 4 4 19 24 31 17 9 5 7 3 18 3 6 5 4 13 8 11 23 3 18 28 23 56 46 103 186
396 869 1029 756 340 220 155 179 155 188 236 209 176 209 178 83 45 35 4 0 0 1 0 0 0 0 0 0 0 0 0
L1[8]:2 0 0 0 0 0 0 0 0 0 0 0 0 1 1 2 0 2 2 23 41 40 35 26 12 156 1 12 7 6 15 11 25 17 25 21 34 39 62 82
144 206 371 600 1652 888 240 134 93 83 108 101 81 71 76 101 212 249 26 4 0 0 1 0 1 0 1 0 0 0 0 0 1
L1[9]:7 0 0 0 0 0 0 0 1 0 2 0 1 0 3 5 12 0 12 28 18 18 9 4 42 3 17 0 3 11 2 19 13 10 14 24 11 69 38 80
121 255 372 427 552 353 131 142 229 157 209 236 263 332 555 904 336 57 19 8 3 2 1 2 0 1 0 0 1 0 0 0
L1[10]:3 0 0 0 0 1 0 1 0 0 0 1 1 1 1 2 17 3 17 19 9 11 8 6 31 5 24 0 7 25 16 11 41 14 20 37 35 110 123
280 518 754 907 1235 586 211 211 87 78 88 97 90 66 52 75 146 56 5 1 0 0 0 0 0 0 0 0 0 0 0 0 1
L1[11]:2 0 0 0 0 1 0 1 1 0 1 0 0 1 3 0 4 0 23 15 4 4 3 1 15 0 13 0 3 6 4 8 43 4 9 25 26 142 60 176 314
603 779 1129 1120 437 213 213 141 70 85 65 77 123 117 46 5 4 1 2 0 1 0 0 0 1 0 0 0 0 0 0
L1[12]:2 0 0 0 0 0 1 0 1 1 0 2 1 0 3 14 34 5 39 21 9 17 4 2 21 6 7 3 3 12 13 13 21 13 9 16 9 42 39 89
131 248 503 1073 778 239 126 65 61 80 114 153 206 300 430 694 421 35 4 7 0 0 0 2 0 2 0 0 0 0 0 0
L1[13]:1 0 0 0 0 1 0 0 0 0 0 2 0 4 1 3 2 0 9 10 9 6 9 3 14 1 7 1 3 12 8 10 7 15 12 35 34 58 57 173 420
548 707 620 379 152 74 94 123 160 166 151 245 318 534 824 102 3 8 2 1 0 2 0 0 1 1 0 2 0 0 0
L1[14]:0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0 3 51 31 16 11 4 7 11 4 6 9 13 13 12 19 26 37 21 59 75 106 211
437 863 1488 1794 733 53 16 10 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
L1[15]:1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 3 4 0 1 4 0 0 4 0 1 0 2 0 2 2 1 2 6 6 4 9 13 49 148 454 472 280
111 89 81 118 199 230 309 320 433 525 909 1148 186 7 1 2 3 1 0 0 0 0 0 0 0 1 0 0
```

从以上 652 维的分类结果可以看出,像素点个数分布有了非常大的改善. 但是总体来说,还是 H 的值为 200 ~ 300 区间的像素点比较多. 量化到 652 维计算量已经比较大了,而且,对于 H 的值在 100 ~ 200 和 300 ~ 360 之间的计算对于结果的影响不大. 因此,将 H 在 100 ~ 200 以及 300 ~ 360 之间的步距改为 10,得到了最终的量化方法,即 193 维的量化.

2.2.3 193 维量化

虽然 652 维的量化计算量比较大,但是检索效果还是比较好的. 由于在使用数据库时,数据库表中一行的容量限制为 8 060 字节,因此量化到 652 维之后就无法将每一个子块中的一维特征矢量存入数据库中,因此只能使用本实验中的量化降维方法. 实验结果如下:

```
L1[0]:2 0 0 1 1 1 3 4 12 36 17 9 21 12 6 25 27 31 72 192 276 503 802 1179 962 313 187 156 132 97 77
136 162 159 189 241 83 11 5 0 1 0 0 0 1 0
```

```

L1[1]:4 0 0 0 0 1 0 3 8 21 4 6 15 9 8 8 15 16 34 104 117 296 521 511 165 94 127 164 168 195 281 356
433 440 527 846 478 155 11 0 2 0 1 0 0 0
L1[2]:2 0 0 1 0 0 3 0 2 39 42 26 64 21 21 32 54 73 179 633 676 1058 899 511 108 53 89 70 92 102 199
310 267 161 127 124 87 15 1 0 0 1 1 1 0 0
L1[3]:0 0 0 1 0 0 0 4 1 1 1 2 1 1 5 3 5 9 22 80 130 260 250 202 187 140 169 289 479 786 1254 910 488
202 179 71 11 0 0 0 0 0 0 0 1 0
L1[4]:0 0 0 0 0 0 0 0 7 21 8 4 9 2 0 10 11 12 38 73 87 176 372 875 434 184 228 420 594 728 655 430
212 174 239 111 25 2 3 0 0 0 0 0 0 0
L1[5]:3 0 1 1 1 1 1 5 10 13 13 4 6 1 2 5 9 14 31 55 84 214 275 191 101 72 81 150 236 287 362 526 553
632 929 998 217 44 8 2 1 2 1 0 1 1
L1[6]:0 0 1 0 0 1 0 7 9 15 2 2 5 10 7 7 8 20 27 80 80 109 269 679 510 247 252 266 253 349 481 416 425
441 530 478 125 27 3 1 0 1 0 0 0 1
L1[7]:2 0 0 2 0 0 0 8 43 48 14 10 21 11 17 19 26 46 79 149 186 396 869 1029 756 340 220 155 179 155
188 236 209 176 209 178 83 45 35 4 0 1 0 0 0 0
L1[8]:2 0 0 0 0 0 2 2 4 64 75 38 157 19 21 36 42 55 101 226 206 371 600 1652 888 240 134 93 83 108
101 81 71 76 101 212 249 26 4 0 1 1 1 0 0 1
L1[9]:7 0 0 0 1 2 1 8 12 40 36 13 45 17 14 21 23 38 80 118 121 255 372 427 552 353 131 142 229 157
209 236 263 332 555 904 336 57 19 8 5 3 1 0 1 0
L1[10]:3 0 1 1 0 1 2 3 20 36 20 14 36 24 32 27 55 57 145 403 518 754 907 1235 586 211 211 87 78 88
97 90 66 52 75 146 56 5 1 0 0 0 0 0 0 1
L1[11]:2 0 1 1 1 1 1 3 4 38 8 4 15 13 9 12 47 34 168 236 314 603 779 1129 1120 437 213 213 141 70 85
65 77 123 117 46 5 4 1 2 1 0 1 0 0 0
L1[12]:2 0 0 1 2 2 1 17 39 60 26 6 27 10 15 26 34 25 51 128 131 248 503 1073 778 239 126 65 61 80
114 153 206 300 430 694 421 35 4 7 0 2 2 0 0 0
L1[13]:1 0 1 0 0 2 4 4 2 19 15 12 15 8 15 18 22 47 92 230 420 548 707 620 379 152 74 94 123 160 166
151 245 318 534 824 102 3 8 2 1 2 1 1 2 0
L1[14]:0 0 0 0 0 0 0 4 3 82 27 11 15 15 26 31 63 80 181 648 863 1488 1794 733 53 16 10 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
L1[15]:1 0 0 0 0 1 0 1 4 4 5 0 4 1 2 4 3 12 13 62 148 454 472 280 111 89 81 118 199 230 309 320 433
525 909 1148 186 7 1 2 4 0 0 0 1 0

```

从以上结果可以看出,采用本实验的量化降维方法后,像素点的分布在 652 维的基础上又有了一定改善. 经过反复实验,得到像素点的分布在 193 维时的量化降维处理结果最佳.

3 实验结果及分析

以 Matlab 为平台,开发了一个图像检索系统,采用平均分块策略,提取图像的颜色直方图,以 Corel 图像库中的每类图片的第 11 张图像的检索结果为例说明.

查准率(Precision Rate)^[6]是评测检索结果最常用的指标,指的是检索结果输出的相关图像的数目占结果输出图像总数目的比值,其中相关图像指的是用户满意的图像. 在本实验中,由于使用的是标准图像库 Corel,里面已经将图像分好了类别,因此,本实验中的相关图像指的是用户要检索的图像属于同种类别的图像. 公式表示如下:

$$Pp = n/N,$$

其中 n 代表检索结果输出的相关图像的数目, N 代表检索结果输出的图像的总数目.

表 1 列出了不同维数量化结果的查准率.

表 2 是对 Corel 图像库中的 10 类图像在不同维数量化方法下的检索查准率的比较.

表 1 不同量化方法的查准率

Table 1 The precision rate of different quantization method

图像	查准率		
	166 维量化	652 维量化	193 维量化
10. jpg	0.10	0.75	0.80
110. jpg	0.25	0.45	0.65
210. jpg	0.45	0.20	0.55
310. jpg	0.40	0.75	0.80
410. jpg	0.95	0.95	1.00
510. jpg	0.25	0.30	0.60
610. jpg	0.65	0.85	0.95
710. jpg	0.75	1.00	0.95
810. jpg	0.05	0.30	0.55
910. jpg	0.35	0.70	0.85

表 2 不同量化方法的平均查准率比较

Table 2 Comparison of average precision rate of different quantization method

图像	166 维	652 维	193 维
土著居民	0.80	0.85	0.88
沙滩	0.40	0.50	0.50
建筑	0.60	0.64	0.65
巴士	0.70	0.79	0.80
恐龙	1.00	1.00	1.00
大象	0.55	0.55	0.63
月季	0.70	0.60	0.70
马	0.95	0.90	1.00
雪山	0.50	0.60	0.75
美食	0.64	0.75	0.80

以上结果说明采用 193 维的量化方法的检索效果还是比较好的。

另外,时间消耗也是衡量检索效率的一个重要指标。经实验,166 维量化计算一幅图像从开始计算到输出相似图像为止,消耗的总时间为 27 s;652 维量化消耗的总时间为 38 s;193 维量化消耗的总时间为 31 s。

从整体上看,本文提出的改进的 HSV—193 维量化方法的检索效率是令人满意的。

4 结语

图像检索是一个有着广阔应用前景的研究领域,本文采用改进的 HSV 颜色空间方法提取颜色的分块直方图特征进行检索,更加客观地描述了图像的颜色特征,提高了图像的检索效率。但是该方法没有很好地解决分块直方图对图像旋转、平移敏感的问题,在今后的研究工作中,要进一步解决这个问题,并将图像的纹理和形状特征引入进来,而且考虑图像蕴含的情感语义,从而进一步提高图像的检索精度。

[参考文献](References)

- [1] Liu Ying, Chen Xin, Zhang Chengcui, et al. Semantic clustering for region-based image retrieval[J]. J Vis Commun Image R, 2008, 20: 157–166.
- [2] Cho S B, Lee J Y. A human-oriented image retrieval system using interactive genetic algorithm[J]. IEEE Trans on Systems, Man and Cybernetics, 2002, 32(3): 452–458.
- [3] 李巧玲. 基于内容的图像检索技术研究[D]. 西安: 西安科技大学计算机科学与技术学院, 2011.
Li Qiaoling. Research on content-based image retrieval technology[D]. Xi'an: College of Computer Science and Technology, Xi'an University of Science and Technology, 2011. (in Chinese)
- [4] 张磊. 基于机器学习的图像检索若干问题研究[D]. 青岛: 山东大学信息科学与工程学院, 2011.
Zhang Lei. Research on image retrieval with machine learning techniques[D]. Qingdao: College of Information Science and Engineering, Shandong University, 2011. (in Chinese)
- [5] 崔屹. 数字图像处理技术与应用[M]. 北京: 电子工业出版社, 1997.
Cui Yi. Digital Image Processing Technology and Application[M]. Beijing: Electronics Industry Publishing House, 1997. (in Chinese)
- [6] 郭士会, 杨明, 王晓芳, 等. 基于 FSRM 的相关反馈图像检索算法[J]. 计算机科学, 2012, 39(s1): 540–542.
Guo Shihui, Yang Ming, Wang Xiaofang, et al. Relevance feedback algorithm based on FSRM in image retrieval[J]. Computer Science, 2012, 39(s1): 540–542. (in Chinese)

[责任编辑: 丁 蓉]