

颜色特征提取项目报告

姓名 罗远浩

学号 2014E8013261184

单位 计算技术研究所

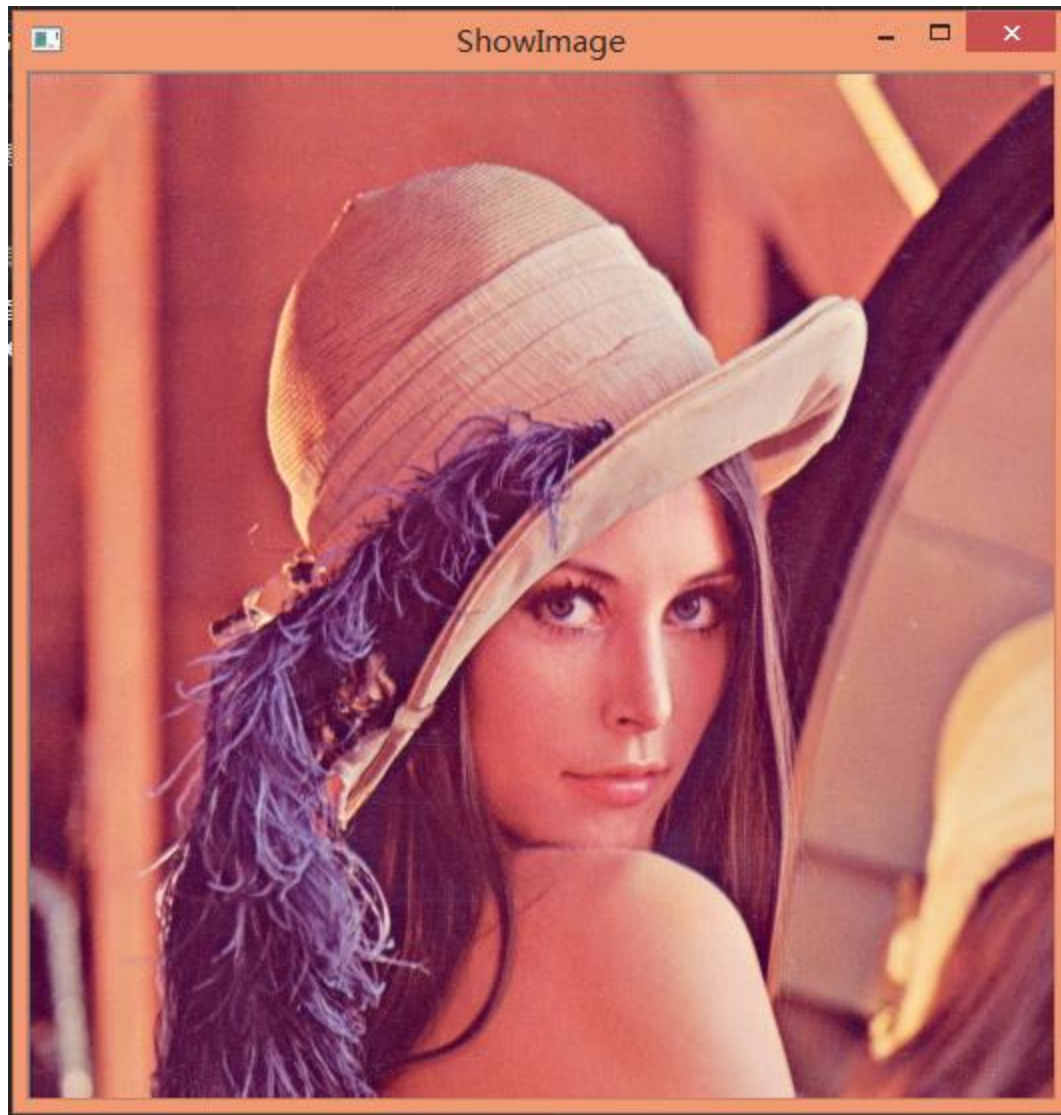
一 程序说明

该程序实现了图像的几种颜色特征的提取，分别是灰度直方图(Gray histogram)，颜色直方图 (Color Histogram)，HSV 直方图，颜色矩 (Color Moment)，颜色量化(Color Quantization)，颜色相关向量 (CCV: Color Coherence Vector)，以及颜色相关图 (Color Correlogram)。

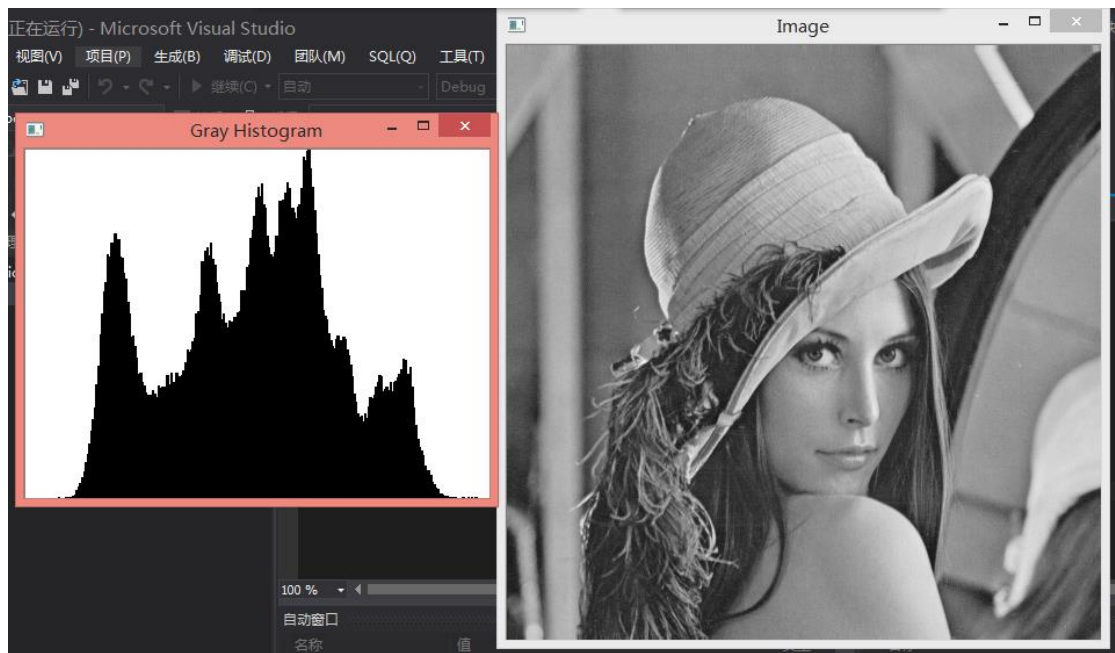
程序实现平台为 Windows8.1 + VC 2012 + OpenCV2.4.4。

二 程序演示

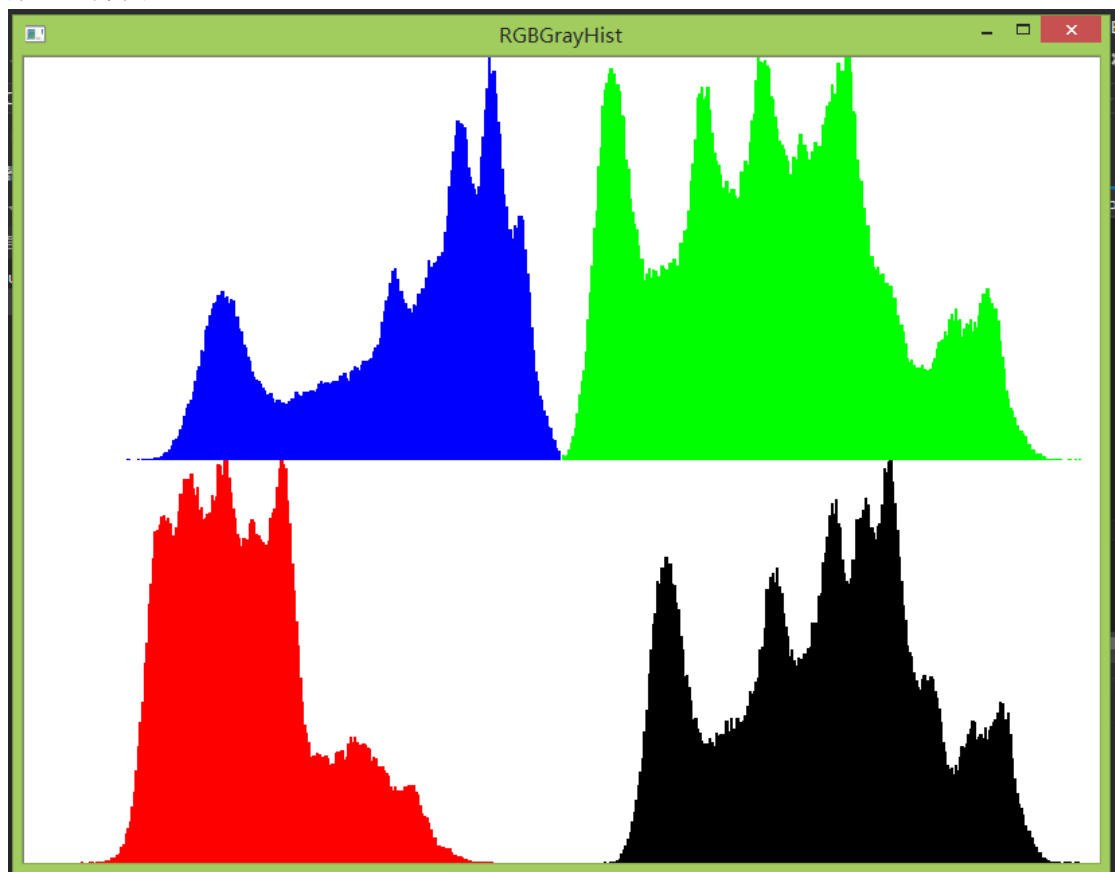
1. 显示图像



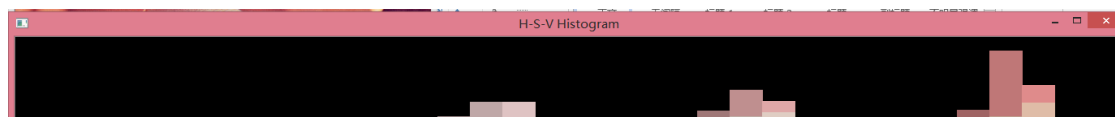
2. 灰度直方图



3. 颜色直方图



4. HSV 直方图



5. 颜色矩

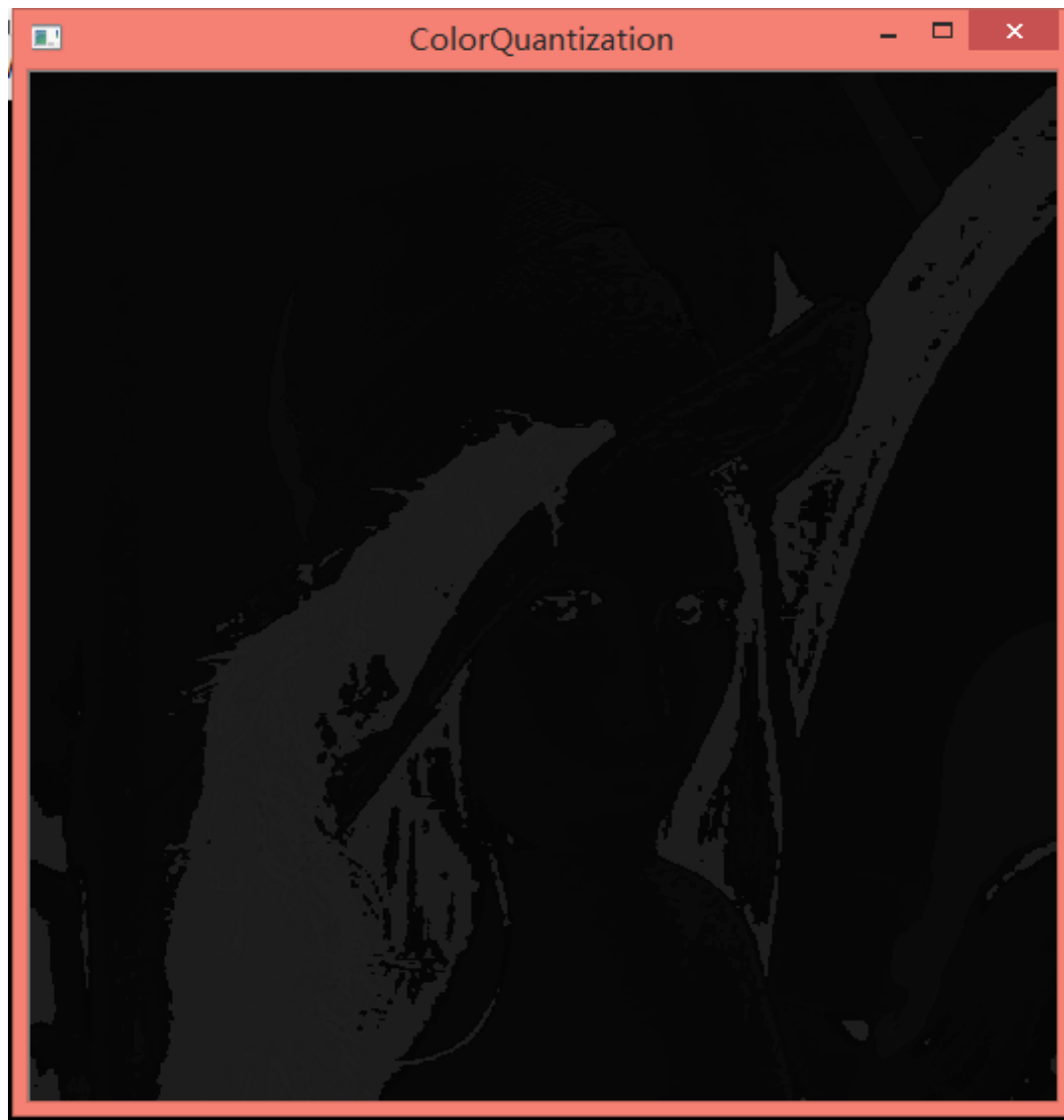
```
E:\StudyF
Mean of H:89
Mean of S:132
Mean of U:179

Variance of H:82
Variance of S:92
Variance of U:121

Skewness of H:-30
Skewness of S:-98
Skewness of U:-137
请按任意键继续. . .
```

6. 颜色量化

参考汪华章，何小海，宰文姣，王伟的论文《基于色彩量化及索引的图像检索》，将图像首先量化到32个颜色区间，具体实现将在下文中详述。计算结果如下图所示：



7. 颜色聚合向量 CCV

根据文献所述阈值通常置为：图像宽度*图像长度/100。

```
请按任意键继续. . .
CCV
      coh      n_coh
bin 0:  0        0
bin 1:  0        0
bin 2:  0         1
bin 3:  0       95
bin 4:  0       65
bin 5:  0     10872
bin 6: 74245    8054
bin 7: 94041    8677
bin 8:  0         0
bin 9:  0         0
bin 10: 7539    7946
bin 11:  0      477
bin 12:  0         0
bin 13:  0         0
bin 14:  0       24
bin 15:  0         4
bin 16:  0         0
bin 17:  0         0
bin 18:  0         0
bin 19:  0         0
bin 20:  0         0
bin 21:  0         0
bin 22:  0         0
bin 23:  0         0
bin 24:  0         0
bin 25:  0         0
bin 26:  0       93
bin 27:  0         0
bin 28:  0     1833
bin 29: 36291   3844
bin 30: 3483   3734
bin 31:  0      826
```

8. 颜色相关图

颜色相关图中的距离设为 10，即针对 32 种颜色计算它们 1-10 距离内的颜色自动相关图。计算结果如下图所示：

E:\StudyFiles\Workspace\VSProject\Solutions\x64\Debug\OpenCV.exe										
bin 31: 0	826									
distance	1	2	3	4	5	6	7	8	9	10
color										
0:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
1:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3:	0.30263	0.11053	0.05965	0.04803	0.03632	0.02588	0.02632	0.02533	0.02076	0.01632
4:	0.03192	0.01346	0.01154	0.00769	0.00077	0.00256	0.00714	0.00192	0.00513	0.00462
5:	0.66523	0.52192	0.44650	0.39680	0.36149	0.33494	0.31154	0.28958	0.27115	0.25525
6:	0.90929	0.86531	0.83822	0.81764	0.80009	0.78362	0.76764	0.75293	0.73919	0.72632
7:	0.93591	0.90524	0.88378	0.86599	0.85076	0.83690	0.82411	0.81259	0.80222	0.79246
8:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
9:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
10:	0.84754	0.77159	0.72415	0.69036	0.66410	0.64032	0.61784	0.59748	0.57641	0.55625
11:	0.46855	0.29691	0.21471	0.15907	0.12862	0.10613	0.08925	0.07842	0.07262	0.06494
12:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
13:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
14:	0.16667	0.02604	0.00694	0.01302	0.00625	0.00347	0.00298	0.00521	0.00231	0.00312
15:	0.18750	0.06250	0.02083	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
16:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
16:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
17:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
18:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
19:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
20:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
21:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
22:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
23:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
24:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
25:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
26:	0.30108	0.07392	0.03405	0.03024	0.02258	0.01703	0.01575	0.01109	0.01135	0.00887
27:	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
28:	0.21496	0.13923	0.12036	0.10600	0.10317	0.09623	0.09339	0.09309	0.09042	0.08804
29:	0.89393	0.82453	0.77456	0.73801	0.70967	0.68649	0.66616	0.64827	0.63218	0.61745
30:	0.67328	0.51895	0.44649	0.41029	0.38654	0.36857	0.35341	0.33943	0.32911	0.32095
31:	0.22406	0.10836	0.07736	0.06049	0.05451	0.05065	0.04666	0.03938	0.03407	0.03161

三 特征提取方法与代码说明

1. 颜色矩 (Color Moment)

颜色矩即通过以下公式计算：

$$\mu_i = \frac{1}{N} \sum_{j=1}^N p_{ij} \quad \text{一次矩 (mean)}$$

$$\sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - \mu_i)^2 \right)^{\frac{1}{2}} \quad \text{二阶矩 (variance)}$$

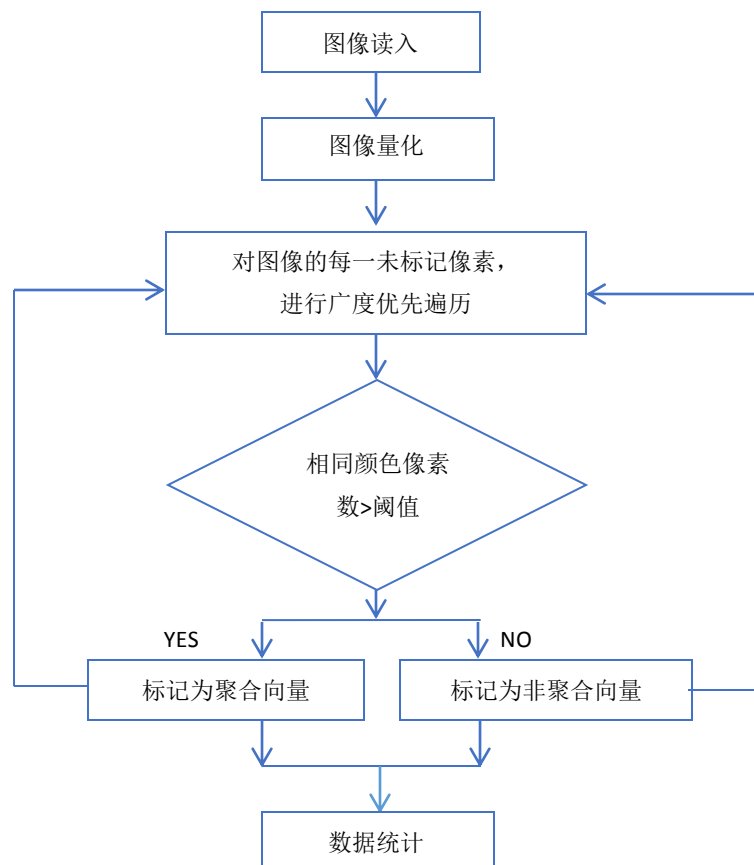
$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - \mu_i)^3 \right)^{\frac{1}{3}} \quad \text{三阶矩 (skewness)}$$

颜色矩实现过程中算法不复杂，主要精力花费在 C++ 的代码调试上，比如进行除法运算时 double 型与 int 的表示形式，以及对 IplImage 数据内容的操作等。比如下面代码：

该部分对算法要求不高，但通过颜色矩的编写，使编程能力有所提高，对 IplImage 的操作也更加熟悉。

2. 颜色聚合向量 (CCV)

颜色聚合向量属于程序中最复杂的一块。具体过程如下图所示：



图像量化参考了汪华章，何小海，宰文姣，王伟的论文《基于色彩量化及索引的图像检索》，将图像首先量化到32个颜色区间，具体划分区间如下：

a. $V < 0.1$

b. $S < 0.1$ 且 $0.1 < V \leq 0.4$

$S < 0.1$ 且 $0.4 < V \leq 0.7$

$S < 0.1$ 且 $0.7 < V \leq 1$

c.

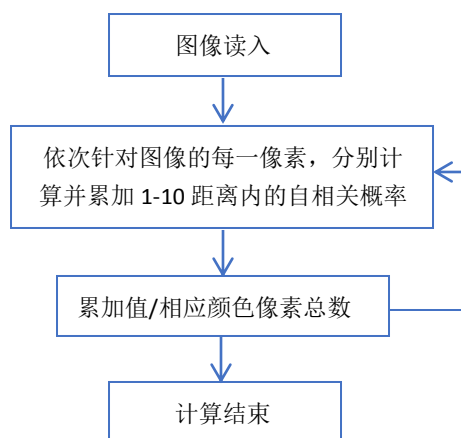
$$h' = \begin{cases} 0 & h \in (0, 20] \cup (330, 360] \\ 1 & h \in (20, 45] \\ 2 & h \in (45, 75] \\ 3 & h \in (75, 155] \\ 4 & h \in (155, 210] \\ 5 & h \in (210, 270] \\ 6 & h \in (270, 330] \end{cases}$$

$$s' = \begin{cases} 0 & s \in (0.1, 0.5] \\ 1 & s \in (0.5, 1.0] \end{cases} \quad v' = \begin{cases} 0 & v \in (0.1, 0.5] \\ 1 & v \in (0.5, 1.0] \end{cases}$$

对于图像像素的遍历，起初采用的是深度优先遍历，因为相对用到的数据结构较少，流程比较清晰，但是深度优先遍历在图像的聚合像素较多的情况下，会出现栈溢出(产生负值，无穷大值等)，原因是函数递归调用的层数太多，以至于函数调用的栈空间不够用。因此更改为广度优先遍历。具体代码比较复杂，可以参见 Image.cpp。

3. 颜色相关图 (Color Correlogram)

颜色相关图的具体过程如下图所示：



颜色相关图计算过程中遇到的以后可借鉴的主要问题有两个：

首先，在 double 中零并不是绝对的零值，而是以一个绝对值非常接近零的数值表示，所以如果一种颜色的每个像素位置的自相关概率都始终为零，则最后的输出会输出非常小的负值，而非零，为了避免这种干扰，在最后的输出时进行了过滤，当结果小于零时，直接输出零。虽然这样逻辑上并无差异，但为了更好的鲁棒性仍值得后续的研究。

其次，在进行除法的时候，要记得判断除数是否为零，若除数为零，会出现绝对值为无穷大值的结果。具体代码如下所示：

```

if(pixelNum[c] != 0)
    correlogram[c][d] = correlogram[c][d] / ((double)pixelNum[c]);
  
```


四 不足及进一步的工作

- 1 在计算 CCV 和颜色相关图时没有预先进行图像平滑。
- 2 在颜色聚合向量 (CCV) 的计算过程中, 部分聚合向量结果的值中会出现很小的数值 (小于阈值), 这在逻辑上与实际算法也不相符, 暂时没有找到原因, 因此当此种问题出现时, 将其看做零值。目前估计问题可能也是出现在 C++ 中数据类型存储形式的问题。
- 3 在颜色相关图计算过程中, double 类型数据零值表示的问题, 尝试找到鲁棒性更好的处理方法。
- 4 颜色聚合向量的计算时间因图像的不同而有较大差异, 对于大部分图像计算很快, 但对于部分聚合颜色块较大的图像计算时间较长。