

阶段报告——阶段 2

实验主题：线积分卷积与双色调映射相结合的彩色素描模拟方法实现

17 小组成员：张家侨，张涵玮，张彬熠，张漫榕，张骏，袁均良

一. 理论学习

1.K-means 算法学习报告

基本概念

图像分割：图像分割是指将一幅图像分解成若干互不相交区域的集合，可以看作是像素的聚类。

k-means 算法：基于距离相似性的聚类算法，通过比较样本之间的相似性，按照距离将样本聚成不同的簇。两个点的距离越近，其相似度就越大。彩色图像中的每一个像素是三维空间中的一个点，三维对应红、绿、蓝三原色的强度，基于 K-means 聚类算法的图像分割，以图像的像素为数据点进行聚类，然后将每个像素点以其对应的聚类中心替代，重构该图像。

1、 随机选取k个聚类质心点 (cluster centroids) 为 $\mu_1, \mu_2, \dots, \mu_k \in \mathbb{R}^n$ 。

2、 重复下面过程直到收敛 {

对于每一个样例 i ，计算其应该属于的类

$$c^{(i)} := \arg \min_j \|x^{(i)} - \mu_j\|^2.$$

对于每一个类 j ，重新计算该类的质心

$$\mu_j := \frac{\sum_{i=1}^m 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^m 1\{c^{(i)} = j\}}.$$

}

k-means 算法的具体过程为：

- 确定聚类中心的个数 k ，即该图像当中主要色调的个数为 K 。论文中给出了 k 值的选取准则——将图像转化到 HSV 色彩空间，通过计算色调直方图来判定彩色图像中主色调

的个数，进而自动地确定 K 值的大小。（存疑）

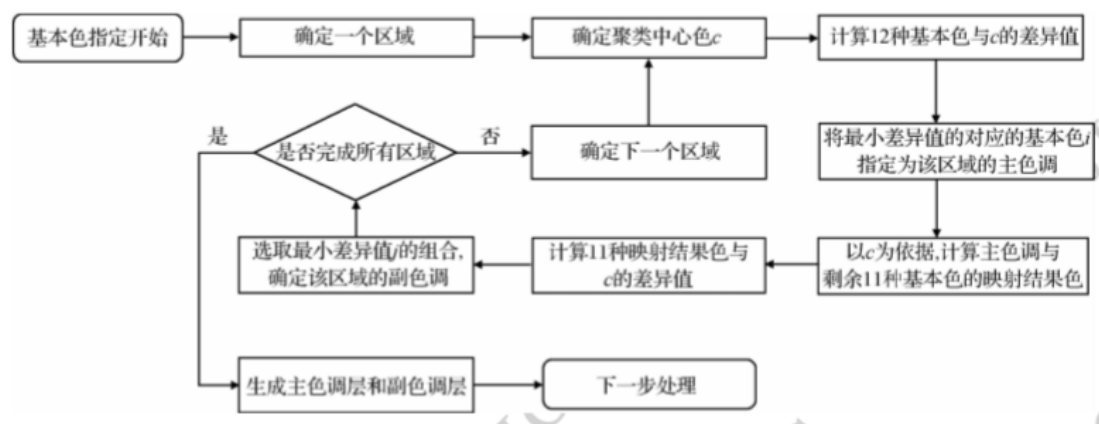
- **随机初始化 K 个聚类中心。**由于 K-means 算法对初始聚类中心的选取较为敏感，若选择的聚类中心不好，得到的聚类效果会非常差，因此 K-means 算法有许多改进方法，其中一种算法如下：
 - a) 在数据集中随机选择一个样本点作为第一个初始化的聚类中心
 - b) 选择其余的聚类中心
 - 1) 计算样本中每一个样本点与初始化的聚类中心之间的距离，并选择其中最短的距离
 - 2) 以概率选择距离最大的样本作为新的聚类中心
 - 3) 重复第（1）（2）步，直到 k 个聚类中心被确定
- **确定 k 个聚类中心**
 - a) 计算每个样本与每个聚类中心之间的相似度，将样本划分到最相似的类别中
 - b) 对每个簇，以所有样本的均值作为该簇新的聚类中心
 - c) 重复第（a）（b）步，直到聚类中心不再改变或达到设定的迭代次数
- **将每个像素点以其对应的聚类中心替代，重构该图像，即可得到图像分割的结果。**

k-means 算法的缺点

从 K-means 算法框架可以看出，该算法的每一次迭代都要遍历所有样本，计算样本到所有聚类中心的距离。因而当样本规模非常大时，算法的时间开销是非常大的。

联系

通过 k-means 分割实现图像的预处理，对于图像中的每一区域，通过用户交互或者自动计算指定区域的基本色，即为该区域的主色调和副色调。具体流程如图所示：



在确定两种基本色之后，即可利用双色调映射的方法分别算出两个基本色层的颜色密度。

2.噪声图原理学习及初步实现

理解了论文中生成噪声图像算法，并尝试去使用了 Matlab 去实现。

结果如下：

原图：



生成的噪声图像：



色彩缩放后的噪声图像：



下一周的计划，将 Matlab 代码转换成 C++代码，完成工程语言统一。

思考：论文中的参数的给出并没有给出理由，单纯是说为经验值，可能以后的改进方向在于参数的改进，以及然参数更加符合现实的设定。

公式如下：

$$\text{noise} = \begin{cases} \text{noise}_1 = \begin{cases} \text{noise}_{\max} & r \geq t_1 \\ \text{noise}_{\min 1} & \text{其他} \end{cases} & p \leq l_1 \\ \text{noise}_2 = \begin{cases} \text{noise}_{\max} & r \geq t_2 \\ \text{noise}_{\min 2} & \text{其他} \end{cases} & l_1 < p \leq l_2 \\ \text{noise}_3 = \begin{cases} \text{noise}_{\max} & r \geq t_3 \\ \text{noise}_{\min 3} & \text{其他} \end{cases} & l_2 < p \end{cases}$$

Matlab 代码如下：

```
1. k=[0.9 0.7 0.7];
2. noise_min=[5 25 80];
3. noise_max=255;
4. p=imread('samplePicture.bmp');
5. p=rgb2gray(p);
6. l=[24 79];
7. [m,n]=size(p);
8. noise_photo=zeros(m,n);
9. for i=1:m
10.     for j=1:n
11.         temp=double(p(i,j));
12.         t=(1-temp/255)*k;
13.         r=rand(1,1);
14.         if (temp<=l(1))
15.             if (r>=t(1))
16.                 noise_photo(i,j)=noise_max;
```


```




17.         else
18.             noise_photo(i,j)=noise_min(1);
19.         end
20.     elseif l(1)<p<=l(2)
21.         if (r>=t(2))
22.             noise_photo(i,j)=noise_max;
23.         else
24.             noise_photo(i,j)=noise_min(2);
25.         end
26.     elseif l2<p
27.         if (r>=t(3))
28.             noise_photo(i,j)=noise_max;
29.         else
30.             noise_photo(i,j)=noise_min(3);
31.         end
32.     end
33. end
34. end
35. noise_photo=uint8(noise_photo);
36. imwrite(noise_photo,'samplePicture1.bmp');

```

二. 项目实施进度

在这一周中，我们在实现部分主要是进行了项目框架的搭建以及图片文件的像素提取，读写等基本功能的实现。整个项目将在 [GitHub 仓库](#) 上进行，免除繁琐的代码传递，增强代码编写的管理。


ZhangJiaQiao / DMT_Project

 Watch
1
 Star
0
 Fork
0







Code
Issues 0
Pull requests 0
Projects 0
Wiki
Settings
Insights


这是数字媒体技术基础的大作业
Edit

Add topics

25 commits
1 branch
0 releases
2 contributors

Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download

	HinaawiTenshi [Add]Set up Color_Sketch_Generator, ready to develop.	Latest commit 34792ac a day ago
	include [Add]Set up Color_Sketch_Generator, ready to develop.	a day ago
	references The file structure has been optimized.	2 days ago
	resources The file structure has been optimized.	2 days ago
	src [Add]Set up Color_Sketch_Generator, ready to develop.	a day ago
	README.md [Modified]1.1 is in progress.	2 days ago

 README.md

《线积分卷积与双色调映射相结合的彩色素描模拟方法》

主要类：

Class `bf`——图像类，用于存储 `bmp` 格式图片以及进行基本的读写操作；

Class `Color_Sketch_Generator`——产生彩图和轮廓图的类；

三. 下周任务

- 1.K-means 算法的代码实现
- 2.噪声图的代码实现
- 3.进行额外的工作（可选）