

# 遗传算法实现图像分割

Andy

Email: guguant[at]yahoo.com

## 编译环境

Opencv3.2.0 + VS2015

## 设计思路

### 编码方式

采用 8 位二进制编码。

### 染色体结构

```
struct ga {  
    int threshold; // 阈值  
    double fitness; // 适应度  
};
```

### 适应度

设计遗传算法的关键是，寻找适应度函数。采用遗传算法进行图像分割，适应度函数为 OTSU 算法，类间方差作为适应度指标。

### 选择-复制

使用赌轮算法，选择适应度强的子代作为下一代。规定子代规模为 64。

### 交叉

遍历 64 个个体。随机生成概率  $p$ ，如果概率  $p$  大于交叉概率，选择该个体为待交叉个体。随机生成两个  $\text{rand}() \% 8$  [0, 7] 之间的随机数， $p1$ ,  $p2$ ，交叉  $p1 \sim p2$  之间的位数。

### 变异

遍历 64 个个体，随机生成概率  $P = \text{rand}() \% 64$ ，比较  $P$  与变异率的大小，如果  $P$  大于变异率，则该个体进行变异。

将变异后的个体作为初始种群，继续以上过程，知道迭代次数达到规定的遗传次数  $T = 1000$ 。

实验结果

1、读取灰度图片，单通道分割

将彩色图片以灰度的形式读取，使用遗传算法求灰度图的 OTSU 阈值。最后，使用 OTSU 对灰度图像进行分割，得到最终的结果。

(otsu 阈值：求 10 次阈值，求平均值)


素材 1: Lena.tiff

来源: <http://sipi.usc.edu/database/download.php?vol=misc&img=4.2.04>



512x512 pixels, 768kb

Color (24 bits/pixel)

方法	阈值				图像分割结果
Otsu	118				
	种群代数	种群规模	交叉率	变异率	阈值 10 次
遗传 算法	10	64	0.95	0.05	122.1
	50				119.9
	250				117.2
	500				128.9
	1000				125.6
	100	10	0.95	0.05	112.4
		50			123.4
		250			124.6
		500			118.4
		1000			120.4
	100	64	0.50	0.05	128.9

			0.80		122.7
			0.85		118.7
			0.95		122.1
			0.98		115.7
	100	64	0.95	0.01	120.8
				0.03	123.1
				0.05	117.5
				0.10	116.4
				0.50	117.5

分析：

采用不同的种群规模，种群代数，交叉率，变异率，输出的阈值不相同，但总体而言接近 OTSU 算法的阈值。采用遗传算法进行图像分割，分割的准确度具有很大的随机性。

结论：

使用遗传算法时，交叉率选择 0.900~0.970，变异率选择 0.001~0.100。

## 二、三通道分割

对彩色图片的 R、G、B 通道分别运用遗传算法求阈值分量，然后根据阈值对每一个通道进行图像分割，最后将 R、G、B 通道合在一起，输出最终的结果。

素材 2: House.tiff

来源: <http://sipi.usc.edu/database/download.php?vol=misc&img=4.1.05>



House

256x256 pixels, 192kb

Color (24 bits/pixel)

## Otsu 算法

阈值:

R: 155 G: 144 B: 134



原图 R0



G0



B0



二值化 R1



G1



B1

图像分割结果



## 遗传算法

种群代数: 10000      种群规模: 64

交叉率: 0.95      变异率: 0.05

RGB 三通道阈值

R: 164    G: 144    B: 137



原图      R0



G0



B0



二值化      R1



G2



B1

图像分割结果

