

# 改进人工蜂群算法在二维 Otsu 图像分割中的应用<sup>①</sup>

孟宪臣, 郭立侠, 潘 丰

(江南大学 轻工过程先进控制教育部重点实验室, 无锡 214122)

**摘 要:** 为了迅速准确的分割图像, 通过对传统蜂群算法选择蜜源方式和缺陷蜜源的调整, 提出了一种基于改进的人工蜂群算法分割二维 Otsu 图像的新方法. 此方法把图像阈值由人工蜂群算法中的蜜蜂表示, 通过引领蜂、侦查蜂和跟随蜂之间的信息共享和分工协作来求出最佳阈值, 成功解决了传统二维 Otsu 图像分割计算量大、运行时间长的缺陷. 实验结果表明, 所提出的算法不仅能得到理想分割结果, 而且分割速率快.

**关键词:** 人工蜂群算法(ABC); 图像分割; 二维Otsu方法

## Improved Artificial Colony Algorithm in the Application of Two-Dimensional Otsu Image Segmentation

MENG Xian-Chen, Guo Li-Xia, PAN Feng

(Key Lab of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, China)

**Abstract:** In order to segment images exactly and quickly, based on the traditional adjustment of colony algorithm selection strategy and defect honey, a new method based on a improved Artificial Bee Colony algorithm segmenting two dimensional Otsu images is proposed. This method looked on the image threshold value as artificial colony algorithm of the bees. The best threshold is approached in parallel via the division of labor, cooperation and information sharing of employed bees, onlookers and scouts. Effectively solved the problem of the traditional two dimensional Otsu image segmentation calculation defects, long operation time. Experimental results show that the proposed algorithm not only can get the ideal segmentation results, but only improved the segmentation speed.

**Key words:** artificial bee colony(ABC) algorithm; image segmentation; two dimensional algorithm

## 0 引言

图像分割是将图像分成若干个具有独特性质的互不相干的区域并提出感兴趣特征的过程<sup>[1]</sup>. 这里特征指的是像素的灰度、纹理、颜色等, 单个区域和多个区域的目标都可以用预先定义的目标来表示. 它是由图像处理到图像分析的必不可少的阶段. 现有的图像分割方法主要分以下几类: 基于阈值的分割方法<sup>[2]</sup>、基于区域的分割方法<sup>[3]</sup>、基于边缘的分割方法<sup>[4]</sup>以及基于特定理论的分割方法等, 本文依据灰度特征分割图像. 图像分割是一个低层次的图像处理任务, 分割算法的结果通常作为图像识别和图像理解等高层次处理任务的输入. 阈值法是一种常用的分割方法, 它假定

图像的直方图呈双峰分布, 通过在两峰之间确定某一准则来选择一个恰当的阈值, 便可实现图像的目标和背景的分. 其中, Otsu法、最小误差阈值法和最佳直方图熵法等为常用的阈值分割法. Otsu阈值法通过选择一个阈值使得类间方差最大, 由于传统的Otsu方法是可分离性的分割方法, 其仅仅反映了图像的灰度分布而没有考虑像素的空间位置信息, 当图像直方图的双峰特性不明显或者图像信噪偏低时, 其分割的准确性不理想, 对此, 刘健庄<sup>[5]</sup>等人提出了二维Otsu自适应阈值分割算法. 该方法不仅考虑了空间邻域信息和灰度信息, 而且采用邻域平均灰度值的方法统计平滑图像和原始图像的二维直方图, 为了使二维类间方差获

①基金项目:国家自然科学基金(61273131);江苏高校优势学科建设工程

得最大的结果, 通常采取二维直方图选择最优二维阈值向量的方法来求得, 而此二维 *otsu* 方法更适合低信噪比图像的分割.

基于二维 *otsu* 的分割方法虽然有较好的分割性能, 但随着维数的增加, 最佳阈值的搜索空间和计算量都将变大. 另外, 一维直方图情形下的一些特征量将变成二维, 其计算的复杂性也增加了. 为了降低计算量, 文中采用了一种改进的智能搜索算法, 即人工蜂群算法(Artificial Bee Colony, ABC)<sup>[6]</sup>来进行图像分割.

### 1 二维 *otsu* 阈值分割方法

用  $f(x, y)$  表示一幅  $M \times N$  大小的图像上坐标为  $(x, y)$  的像素的灰度值, 计算每个像素点  $(x, y)$  的邻域平均灰度值来得到一个平滑的图像  $h(x, y)$ .

$$h(x, y) = \left[ \frac{1}{9} \sum_{m=-1}^1 \sum_{n=-1}^1 f(x+m, y+n) \right] \quad (1)$$

其中,  $0 < x+m < M, 0 < y+n < N$ . 二元组  $(i, j)$  由  $h(x, y) = j$  和  $f(x, y) = i$  组成. 此二元组出现的频率由  $(i, j)$  出现的频数  $c_{ij}$  表示:

$$P_{ij} = \frac{c_{ij}}{M \times N}, 0 \leq i, j \leq L-1 \quad (2)$$

可得  $0 \leq P_{ij} \leq 1$ , 这样便产生了二维灰度直方图. 如图1所示的由阈值点  $(s, t)$  分割的4个区域.

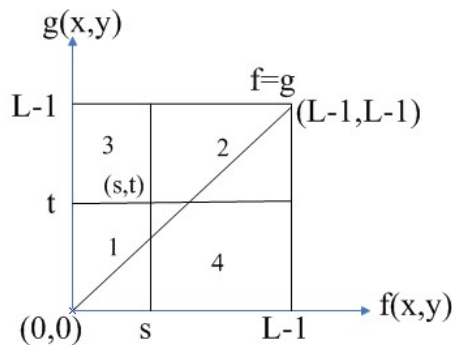


图 1 二维灰度直方图

区域 1 与 2 分别代表图像的目标和背景, 区域 3 和 4 则表示图像的边缘点和噪声. 由于噪声点和边缘点占很少的一部分. 所以, 在计算时候只考虑区域 1 和 2, 而认为远离对角线的分量的概率近似为 0. 定义目标和背景出现的概率  $\omega_0(s, t)$  和  $\omega_1(s, t)$  分别为:

$$\omega_0(s, t) = \sum_{i=0}^s \sum_{j=0}^t P_{ij}$$

$$\omega_1(s, t) = \sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} P_{ij} = 1 - \omega_0(s, t) \quad (3)$$

其中  $P_{ij}$  表示图像中(灰度级, 邻域平均灰度级)对出现的联合概率.

两类对应的均值矢量  $\mu_0$  和  $\mu_1$  为:

$$\mu_0 = (\mu_{0i}, \mu_{0j})^T = \left[ \frac{\sum_{i=1}^t \sum_{j=1}^s iP_{ij} / \omega_0(s, t), \sum_{i=1}^t \sum_{j=1}^s jP_{ij} / \omega_0(s, t)} \right] \quad (4)$$

$$\mu_1 = (\mu_{1i}, \mu_{1j})^T = \left[ \frac{\sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} iP_{ij} / \omega_1(s, t), \sum_{i=s+1}^{L-1} \sum_{j=t+1}^{L-1} jP_{ij} / \omega_1(s, t)} \right] \quad (5)$$

二维直方图总的均值矢量为

$$\mu_T = (\mu_{Ti}, \mu_{Tj})^T = \left( \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} iP_{ij}, \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} jP_{ij} \right)^T \quad (6)$$

假定远离直方图对角线的  $P_{ij}$  为零, 即二、四象限的  $P_{ij}$  可以忽略不计, 则可以得到一个类间离散测度:

$$\begin{aligned} S_B(s, t) &= \omega_0(s, t)(\mu_0 - \mu_T)^T(\mu_0 - \mu_T) + \\ &\omega_1(s, t)(\mu_1 - \mu_T)^T(\mu_1 - \mu_T) = \omega_0(\mu_0 - \mu_T)^T(\mu_0 - \mu_T) + \\ &\omega_1(\mu_1 - \mu_T)^T(\mu_1 - \mu_T) = \omega_0[(\mu_{0i} - \mu_{Ti})^2 + (\mu_{0j} - \mu_{Tj})^2] + \\ &\omega_1[(\mu_{1i} - \mu_{Ti})^2 + (\mu_{1j} - \mu_{Tj})^2] \end{aligned} \quad (7)$$

那么最佳的阈值  $(s, t)$  满足下式:

$$S_B(s, t) = \max \{S_B(s, t)\} \quad (8)$$

### 2 人工蜂群算法

通常将蜜蜂采蜜机制解决函数优化的问题由人工蜂群算法来模拟, 大体可将人工蜂群分为跟随蜂、引领蜂和侦查蜂三类, 引领蜂、跟随蜂用来搜索蜜源, 侦查蜂则通过随机产生蜜源来避免陷入局部最优<sup>[7]</sup>. 在人工蜂群算法(ABC)对图像进行分割时, 每个阈值用一个蜜源位置来代表, 适应度函数值用蜜源的花蜜量来对应. 起初, 引领蜂凭借记忆在其邻域搜索产生一个新的位置来进行比较, 如果记忆中的最优位置劣与新位置时, 用新位置替换它, 否则, 保持原有位置不变. 所有的引领蜂完成搜索后, 通过“摇摆舞”的形式与跟随蜂共享蜜源信息, 然后跟随蜂通过所获的信息按与花蜜量相关的概率选择一个蜜源位置. 如果在采蜜过程中, 蜜源经很多次搜索后不变, 对应的引领蜂改变成侦查蜂, 随机搜索新蜜源. 通过以上方法

不停地循环迭代,直到达到算法的终止条件为止<sup>[8]</sup>。

算法运行中引领蜂、跟随蜂、蜜源数量都为 N。

ABC 算法的具体步骤如下

步骤1: 按照式(9)随机产生2N个位置。

$$W_{ij} = x_{jL} + rand \times (x_{jH} - x_{jL}) \quad (9)$$

式中,  $W_{ij}$  为第 i 个蜜蜂第 j 维搜索后的位置;  $x_{jL}$ 、 $x_{jH}$  分别表示第 j 维变量的上下界。

步骤2 在蜜源附近引领蜂按式(10)搜索新蜜源。

$$V_{ij} = x_{ij} + R_{ij} \times (x_{ij} - x_{kj}) \quad (10)$$

式中,  $V_{ij}$  为新的蜜源位置;  $x_{ij}$  是蜜源 i 的第 j 维位置;  $x_{kj}$  代表随机选取的不等于 i 的蜜源 k 的第 j 维位置;  $R_{ij}$  为[-1,1]间的随机数,接下来计算新位置  $V_{ij}$  的适应度函数值。

步骤3 根据  $x_i$  的适应度函数值  $f_j$ , 计算概率

$P_j$  :

$$P_j = \frac{fit_j}{\sum_{j=1}^N fit_j} \quad (11)$$

$$fit_j = \begin{cases} \frac{1}{1+f_j} & f_j \geq 0 \\ 1+abs(f_j) & f_j < 0 \end{cases} \quad (12)$$

步骤4 跟随蜂根据从蜜源处引领蜂获得的花蜜信息,依概率  $P_j$  按轮盘赌方式选择蜜源,并计算该位置下适应度函数值,在其附近按式(10)搜索新蜜源。

步骤5 比较引领蜂和跟随蜂搜索的蜜源的花蜜数量大小,其中引领蜂、蜜源位置由花蜜数量较优的 N 个位置来表示,其余为跟随蜂位置。

步骤6 判断是否有放弃的位置,若存在则相应引领蜂变成侦查蜂,按式(9)随机产生新蜜源。

步骤7 将新确定的蜜源、跟随蜂、引领蜂位置从步骤 2 重新进行搜索,直到达到最大迭代次数,输出最佳蜜源位置,即最优分割阈值(s, t)。

### 3 改进的人工蜂群算法在图像分割中的应用

传统人工蜂群算法由于容易使算法易陷入局部最优,而且在进化的过程中,每代的最差解会参与新解的产生,这样并不能很好的找到全局最优解,即最佳的分割阈值,因此为了提高算法的搜索精度,本文在

以下两个方面做了改进。

#### 3.1 选择蜜源方式的调整

在ABC算法中,跟随蜂选择蜜源时依据的轮盘赌方式是一种较贪婪的选择方式,很容易使算法陷入提前停滞和过早收敛的情况。在自由搜索算法中通过灵敏度与信息素(与优化问题的目标函数的解成正比)配合选择区域,个体之间使用信息素进行间接通讯。理论上个体可以搜索任何区域,可以在其自身的当前最佳分割阈值邻域周围,也可在其他个体或群体发现的当前最佳分割阈值邻域周围这就在很大程度上避免了过早收敛<sup>[9]</sup>。这种区域选择的方式可以看成是一种新的方法。具体过程如下:

$$nf(i) = \begin{cases} \frac{f(i) - f_{\min}}{f_{\max} - f_{\min}} & f_{\max} \neq f_{\min} \\ 0 & \text{其他} \end{cases} \quad (13)$$

式中,  $f(i)$  为个体的适应度值;  $nf(i)$  为第 i 个蜜源的信息素;要求  $nf(i) \leq S(i)$ ,  $S(i)$  为第 i 个跟随蜂的灵敏度。其中改进人工蜂群的适应度函数利用二维Otsu来设计如上式(7)来表示,其中s表示灰度值,t表示领域均值。

#### 3.2 缺陷蜜源的替换

在人工蜂群算法中,引领蜂和跟随蜂主要依靠交叉操作指导种群的进化,这种方式存在许多不足。算法的收敛速度则在一定程度上收到了影响。于是可以用某种方法产生新蜜源代替缺陷蜜源。从文献 [10] 可以看出:采用产生候选解的相对点取代原始解的反向学习策略(OBL)与随机产生点来取代原始解的方法相比具有更加明显的效果,不但避免了过早的陷入局部最优,而且在很大程度上使搜索精度得到了提高。于是采用文献[11]提出的反向学习策略产生新蜜源来取代缺陷蜜源。具体方式如下:

引用 OBL 后的新蜜源位置为  $X'_b$ , 则新位置的 第 j 维  $X'_{bj}$  为

$$X'_{bj} = X_{jL} + X_{jH} + rand \times X_{bj} \quad (14)$$

判断新位置 and 原位置对应的蜜源的好坏来决定现在蜜源的位置即分割阈值。为了提高抗噪性能,本文利用文献[12]中所提出的分割判别法:

$$b(m,n) = \begin{cases} 1 & 0 \leq f(m,n) + g(m,n) \leq T+S \\ 0 & f(m,n) + g(m,n) > T+S \end{cases} \quad (15)$$

其中  $f(m,n)$  为像素点  $(m,n)$  的灰度级,  $g(m,n)$  为  $(m,n)$  点的邻域平均灰度级,  $b(m,n)$  为分割后

的二值图像. 图 2 给出了算法的分割流程图.

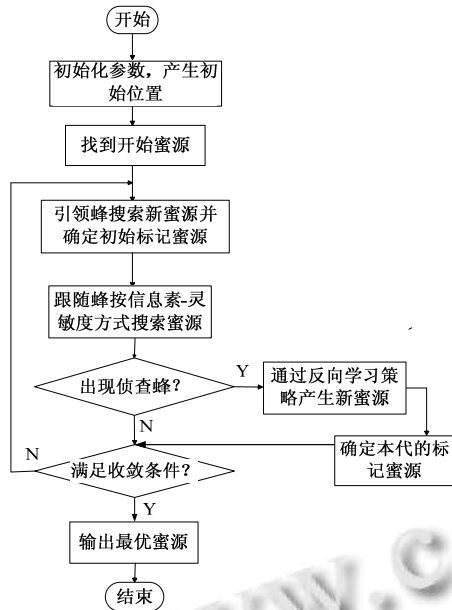


图2 算法分割流程图

为了验证改进算法的性能, 本文选取了以下几个典型的基准测试函数.

1) Sphere 函数

$$f_1(x) = \sum_{i=1}^n x_i^2 \quad (16)$$

$$-5.12 \leq x_i \leq 5.12 (x = 1, 2, \dots, n)$$

2) Rosenbrock 函数

$$f_2(x) = \sum_{i=1}^n [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \quad (17)$$

$$-2.048 \leq x_i \leq 2.048 (x = 1, 2, \dots, n)$$

3) Schaffer 函数

$$f_3(x) = 0.5 + \frac{\sin^2(\sqrt{x_1^2 + x_2^2}) - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2} \quad (18)$$

$$-5.12 \leq x_i \leq 5.12 (x = 1, 2, \dots, n)$$

为了验证新算法的有效性, 将基准函数的维度扩大, 与文献[13]进行对比, 参数设置为: 蜂群总数 200, 即引领蜂个数为 100, Limit 为 100, 循环次数 50, Rosenbrock 函数的定义域设为[-10 10], 取均值和标准差如表 1 所示.

表1 与文献[13]的实验结果比较

函数	维	迭代数	标准ABC算法		改进ABC算法	
			均值	标准差	均值	标准差
Sphere	10	5000	5.1721e-017	1.32015e-017	4.1854e-017	1.1247e-017
	20	7500	2.4363e-016	3.52147e-017	2.0121e-016	4.0123e-017
Rosenbrock	5	3500	0.00435864	0.00324758	5.1024e-012	1.3412e-011
	10	5000	0.00495203	0.00275482	2.4789e-010	4.4785e-010
Schaffer	2	350	1.2457e-005	1.74485e-005	1.4005e-013	5.7487e-013

通过反复实验, 不难看出本文算法在高维函数优化中也有很好的效果, 尤其在Schaffer、Rosenbrock函数中搜索精度和稳定性方面获得较高提升, 如图3和图4所示. 而在Sphere函数则与传统的ABC算法效果接近.

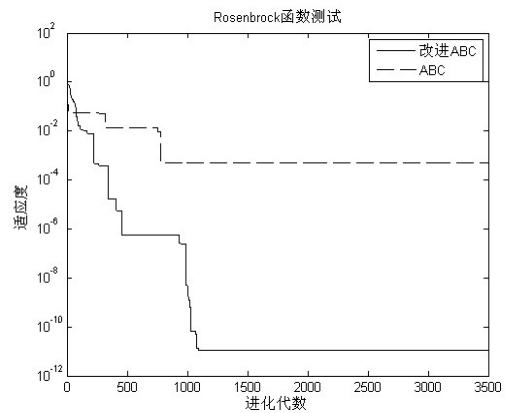


图3 Rosenbrock(5D)结果

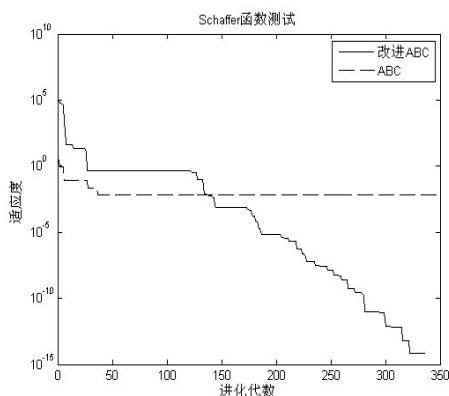


图4 Schaffer(2D)结果

为了进一步研究改进后的蜂群算法在种群增加后对最优值的影响,分别给出了三种测试函数的最优值曲线对比图,如图5,图6,图7所示.为更清晰地显示效果,y轴采用对数坐标,x轴为相对应的循环次数.

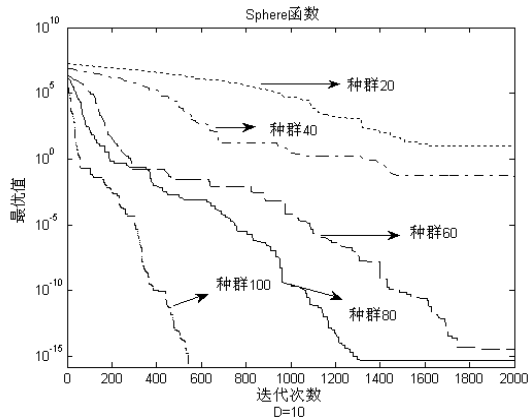


图5 Sphere 函数不同种群对比图

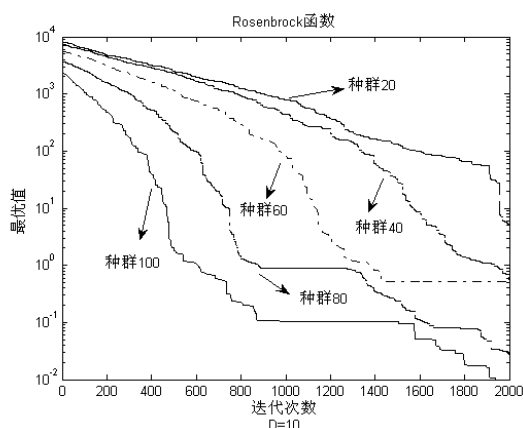


图6 Rosenbrock 函数不同种群对比

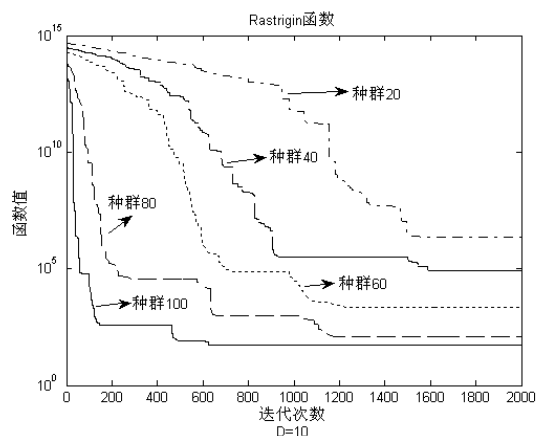
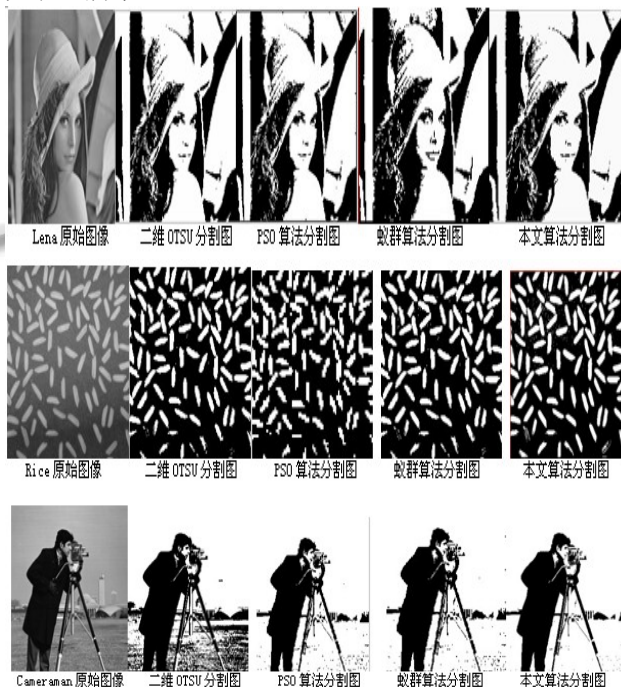


图7 Rastrigin 函数不同种群对比

从图中可以看出改进后的算法对不同的测试函数进行测试,在种群数目增加时搜索精度和稳定性方面均有一定提高,值得提出的是,增加种群数目的同时,也伴随着计算时间的明显增加.

### 4 仿真实验结果

在 matlab 环境下将改进的算法对不同图像进行仿真实验.图像分别采用了 Lena, Rice, Cameraman 和 气门摇臂.实验结果如图 8 所示,四幅图的直方图曲线如图 9 所示,同时与其他算法进行了比较,其结果如表 2 所示.



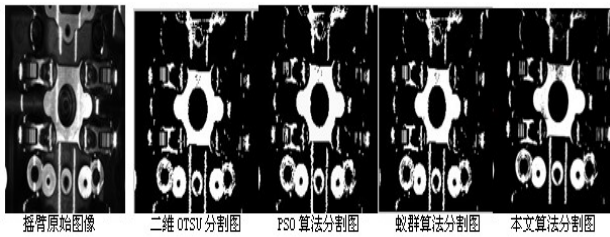


图 8 多种算法分割对比图

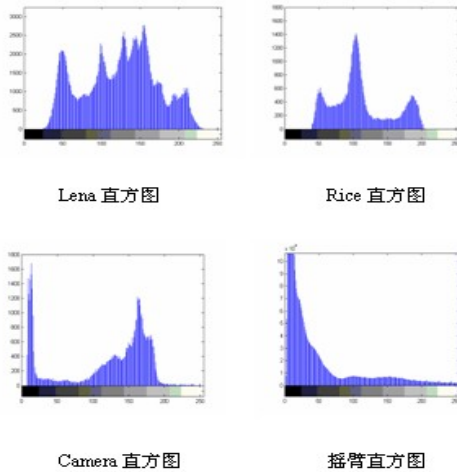


图 9 四种图像直方图

表 2 不同算法阈值选取和运行时间

图像	二 维 OTSU 分割	PSO 算法	蚁 群 算 法	本 文 算 法
Lena	115/18.0 7s	118/0.84s	122/0. 94s	116/1. 12s
Rice	121/23.6 7s	128/1.14s	128/0. 45s	133/0. 81s
Cam eraman	133/17.4 2s	116/0.95s	116/0. 74s	97/0.4 7s
摇臂	128/20.4 7s	123/1.04s	112/0. 64s	130/0. 97s

由图 8 效果可以看出, 经过加入本文算法分割后的图像效果要比其他算法分割的更好. 其原因在于对人工蜂群算法的采蜜聚群行为进行精确地分类, 可以有效地搜索到最佳阈值, 使分割效果最好.

从表 2 给出的最优阈值和计算时间可以知道在相同条件下, 通过本文算法计算的时间明显快于二维阈值, 表明了处理速度方面本文算法的优势, 为后续图像处理奠定了良好的基础.

### 5 结束语

由于传统二维Otsu阈值图像分割计算量大等特点,

本文应用改进人工蜂群算法对二维阈值图像分割进行了提升,通过分析仿真结果可以看出改进蜂群算法优化后的二维阈值分割不但使计算量大大减少,而且有效地搜索到全局最优二维阈值,改善了图像分割的效果,提高了运行效率.

### 参考文献

- 1 杨怀义. 图像分割中算法的应用研究. 计算机仿真, 2012,29(2):229-232.
- 2 肖超云,朱伟兴. 基于Otsu准则及图像熵的阈值分割算法. 计算机工程,2007,33(7):188-189,209
- 3 尤建洁,周则明,王平安. 基于增强型EM模型重叠区域图像分割算法. 微电子学与计算机,2013,30(2):157-160.
- 4 Wang YP, Soh VCM, Ban KW, et al. Improved canny edges using ant colony optimization. Proc. of the 5th International Conference on Computer Graphics, Imaging and Visualization. Washington DC. IEEE Computer Society. 2008. 197-202.
- 5 伊力哈木,亚尔买买提. 基于改进的自适应分水岭图像分割方法研究. 计算机仿真,2013,30(2):373-377.
- 6 Karabogad. An idea based on honey bee swarm for numerical optimization, TR06. Kangser, iTurkey. Erciyes Universtiy, 2005.
- 7 暴励,曾建潮. 自适应搜索空间的混沌蜂群算法. 计算机应用研究,2010,27(4):1331-1335.
- 8 Liu XB, Cai ZX. Artificial bee colony programming made faster. Natural Computation, 2009, (8): 14-16.
- 9 周晖,李丹美,邵世煌等. 一种新的群集智能算法—自由搜索. 东华大学学报,2007,33(05):579-583.
- 10 Rahnamayan S, Tizhoosh HR, Salama MMA. Opposition versus randomness in soft computing technique. Applied Soft Computing, 2008, 8(2): 906-918.
- 11 Rahnamayan S, Tizhoosh HR, Salama MMA. Opposition-based differential evolution. IEEE Trans. on Evolutionary Computation, 2008, 12(1): 64-79.
- 12 潘喆,吴一全. 二维 Otsu 图像分割的人工鱼群算法. 光学学报,2009,29(8):2115-2121.
- 13 王慧颖,刘建军,王全洲. 改进的人工蜂群算法在函数优化问题中的应用. 计算机工程与应用,2012,48(19):36-39.