

# 基于 NW 小世界邻居的粒子群多阈值分割算法

陈自郁 何中市 张程  
(重庆大学计算机学院 重庆 400044)

**摘要** 针对多阈值分割问题,提出了一种新的多阈值分割算法。此算法采用相对类内方差代替传统 Otsu 算法中的绝对类内方差,改善了传统 Otsu 对小对象分割不理想的弱点;采用 NW 小世界模型作为粒子群优化的社会认知拓扑结构,具有较好的全局寻优能力和较快的收敛速度。实验结果显示此算法具有好的性能。

**关键词** NW 小世界模型,粒子群优化,多阈值分割,类方差

**中图法分类号** TP18 **文献标识码** A

## Image Multilevel Thresholding Based on PSO with NW Small World Neighborhood

CHEN Zi-yu HE Zhong-shi ZHANG Cheng  
(College of Computer Science, Chongqing University, Chongqing 400044, China)

**Abstract** As for image multilevel thresholding, a novel algorithm was proposed. Adopting relative intra-class variance instead of absolute intra-class variance, the algorithm improved the segmentation result for small objects with Otsu method. Introducing NW small world model into particle swarm optimization, the algorithm gained better optimization performance. Experimental results show that the proposed algorithm was promising and outperformed some existing techniques.

**Keywords** NW small world model, Particle swarm optimization (PSO), Multilevel thresholding, Class variance

图像分割是数字图像处理中最重要的技术之一,目的是依据图像的特征将图像划分成不同的区域。在众多分割技术中,阈值分割由于其简单、快速,已得到了最广泛的应用。

阈值分割技术有不同的分类,在研究中我们发现其实质是一个优化问题,所以我们将阈值分割过程分为 3 部分:(1)图像特征模型;(2)数学模型和优化(判别)标准;(3)优化算法。根据这 3 部分的不同选择,形成了不同的阈值分割算法。

图像特征模型是阈值分割的基础。在现有的阈值分割技术中,大多是以图像的灰度分布直方图作为特征模型,例如 Otsu 方法<sup>[1]</sup>、ME 阈值法<sup>[2]</sup>等;也有采用空间关系作为图像的特征模型<sup>[3,4]</sup>和采用图像局部特征作为分割依据的局部特征模型<sup>[5,6]</sup>。数学模型和优化标准是在图像特征模型的基础上获得阈值的方法,是阈值分割的关键。例如采用高斯模型模拟直方图或直接使用直方图的形状特征将其转化为数学问题,再采用优化标准,例如类方差最小<sup>[1,7]</sup>、最大熵或交叉熵<sup>[8,9]</sup>、聚类<sup>[4,10]</sup>等方法最终得到结果。然而,随着阈值级别的增加,无法在有限时间内完成通过穷举的方法得到多个阈值,所以需要采用优化算法来得到最佳阈值。目前有很多优化算法已应用于多阈值分割中,例如:遗传算法、模拟退火等<sup>[11]</sup>。粒子群算法<sup>[12]</sup>是其中较新的优化算法,由于其思想简单、操作方便,近几年受到了人们广泛的关注。

本文以 Otsu 方法为基础,利用相对类方差最小作为优化

标准,使用 NW 小世界模型作为粒子群的社会认知结构,实现了一个新的多阈值分割算法。此算法解决了 Otsu 类方差法不能准确定位谷值,对小对象的分割效果差的缺点;较好地改进了传统粒子群算法易陷入局部最优的弱点,但仍保持了较好的收敛能力。

## 1 类内方差法

### 1.1 Otsu 方法

Otsu 方法关于最佳阈值的定义是使类内方差的加权和最小的阈值。设  $\sigma_w^2(t)$  是类内方差的加权和;  $\sigma_1^2(t)$  是值小于或等于  $t$  (称为第一类)的类方差,  $\sigma_2^2(t)$  是值大于  $t$  (称为第二类)的类方差;  $q_1(t)$  是值小于或等于  $t$  的类概率,  $q_2(t)$  是值大于  $t$  的类概率;  $\mu_1(t)$  是第一类的均值,  $\mu_2(t)$  是第二类的均值。类内方差的加权和  $\sigma_w^2$  定义为:

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t) \tag{1}$$

其中  $q_1(t) = \sum_{i=1}^t P(i)$ ,  $q_2(t) = \sum_{i=t+1}^I P(i)$

$$\mu_1(t) = \frac{\sum_{i=1}^t iP(i)}{q_1(t)}, \mu_2(t) = \frac{\sum_{i=t+1}^I iP(i)}{q_2(t)}$$

$$\sigma_1^2(t) = \frac{\sum_{i=1}^t [i - \mu_1(t)]^2 P(i)}{q_1(t)}, \sigma_2^2(t) = \frac{\sum_{i=t+1}^I [i - \mu_2(t)]^2 P(i)}{q_2(t)}$$

利用简单顺序搜索所有可能的值,得到使  $\sigma_w^2(t)$  最小的最佳值:  $t^* = \arg \min_t \{\sigma_w^2(t)\}$ 。然而 Otsu 方法并不能很好地

到稿日期:2008-08-11 本文受国家 863 计划课题(2007AA01Z423),重庆市自然科学基金(2007BB2134),重庆大学研究生科技创新基金(200701Y1A0280214)资助。

陈自郁(1976-),女,博士生,讲师,CCF 会员,研究方向为计算机视觉与模式识别, E-mail:chenziyu@cqu.edu.cn;何中市(1965-),男,博士,教授,博士生导师,研究方向为人工智能、自然语言处理、数据挖掘与机器学习;张程(1977-),男,博士生,研究方向为 Web 智能。

找到直方图的谷,它选择的阈值偏向与方差大的类,也就是说 Otsu 方法不能处理小对象的分割。

### 1.2 修改的 Otsu 方法

本文采用了修改的 Otsu,使其不仅保留原有 Otsu 的特点,并能够较准确地发现直方图的谷,较好地解决小对象的分割。修改后的  $\sigma_w^2$  定义为:

$$\sigma_w^2(t) = \sigma_1^2(t) + \sigma_2^2(t) = \sum_{i=1}^l [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)} + \sum_{i=l+1}^l [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)} \quad (2)$$

注意在 Otsu 方法中

$$\begin{aligned} \sigma_w^2(t) &= q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t) \\ &= q_1(t) \sum_{i=1}^l [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)} + q_2(t) \sum_{i=l+1}^l [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)} \\ &= \sum_{i=1}^l [i - \mu_1(t)]^2 P(i) + \sum_{i=l+1}^l [i - \mu_2(t)]^2 P(i) \quad (3) \end{aligned}$$

从式(2),式(3)可以看到 Otsu 算出的结果是绝对类方差和,修改后的 Otsu 对于每个类方差都除以它的类概率,得到的是相对类方差和,更准确地描述了两个类之间的关系。

### 1.3 两种方法比较

为了说明修改后的 Otsu 方法对于小对象分割的有效性,并比较其与 Otsu 的阈值分割的不同结果,本文用这两种方法分别对经典图像 F14 进行了阈值分割,如图 1 所示。

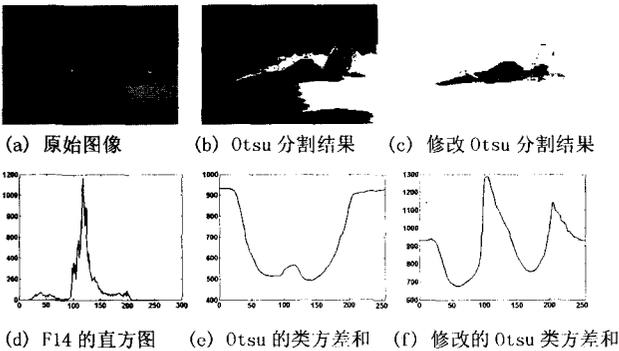


图 1 F14

从图像 F14 的直方图(图 1. d)中可以看到其直方图大体分两类,然而这两类的面积差距很大,直方图谷的位置大约在 70 的位置。在计算类内方差和时,如果按照 Otsu 的方法,不考虑每个类本身所占的概率,那么所算得的结果将偏向于类方差大的一方。图 1. e 和图 1. f 分别显示了这两种方法所有可能阈值的类方差和,从图中可看出修改的 Otsu 方法得到的类方差和图更符合 F14 直方图的规律。从分割结果来看,修改后的 Otsu 较准确地将原图中的 F14 飞机分割出来,而 Otsu 方法( $t^* = 137$ )分割效果较差,而且修改后的 Otsu 所得到的阈值( $t^* = 59$ )更接近直方图谷的位置。

## 2 NW 小世界邻居的粒子群算法

### 2.1 基本粒子群算法

粒子群优化算法(Particle Swarm Optimization, PSO)<sup>[12]</sup>是由 Eberhart 博士和 Kennedy 博士于 1995 年提出的一种新的进化计算技术。基本原理如下:

粒子群由  $M$  个运动的粒子组成,每个粒子  $i$  在  $t+1$  时刻

的运动速度、位置、历史最佳位置以及邻最佳位置为:

$$V_i(t+1) = \omega V_i(t) + c_1 \times Rand_1() \times (pBest_i(t) - X_i(t)) + c_2 \times Rand_2() \times (lBest_i(t) - X_i(t)) \quad (4)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (5)$$

$$pBest_i(t+1) = \begin{cases} pBest_i(t), & \text{如果 } f(X_i(t+1)) \geq f(pBest_i(t)) \\ X_i(t+1), & \text{如果 } f(X_i(t+1)) < f(pBest_i(t)) \end{cases} \quad (6)$$

$$lBest_i(t+1) = \underset{n \in Neighbor(i)}{\operatorname{argmin}} f(pBest_n(t)) \quad (7)$$

其中  $f(X)$  是适应度函数,即判断位置  $X$  的优劣标准,这里以此函数的值最小的  $X$  为最优解。

从式(4)可以看出,粒子通过 3 个方面来改变自己的位置:自身的速度  $V_i(t)$ ;自己的历史最佳位置  $pBest_i(t)$ ,即自我认知;邻居的最佳位置  $lBest_i(t)$ ,即社会认知。

而社会认知结构直接影响到整个粒子群的寻优能力和收敛性。全互连结构(记为 gBest,如图 2 所示)和环形结构(记为 lBest,如图 3 所示)是最常用的粒子群社会认知结构。全互连结构限制了粒子的多样性,很容易陷于局部最优,但一般收敛较快;环形结构虽保持了粒子的多样性,不易陷入局部最优,但收敛能力较差。

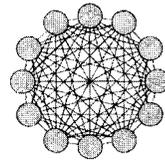


图 2 gBest 社会认知结构

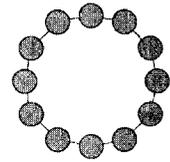


图 3 lBest 社会拓扑结构

### 2.2 NW 小世界邻居的粒子群算法

作为从完全规则网络向完全随机图的过渡,Watts 和 Strongtz 于 1998 年首次引入了小世界模型,称为 WS 小世界模型<sup>[13]</sup>。然而,WS 小世界模型构造算法中的随机化过程有可能破坏网络的连通性。因此,Newman 和 Watts 后来提出另一个小世界模型,称为 NW 小世界模型<sup>[14]</sup>。

NW 小世界模型是通过用“随机化加边”取代 WS 小世界模型构造中的“随机化重连”而得到的。具体构造算法如下:

(1) 从规则图开始:考虑一个含有  $N$  个点的最近邻耦合网络,它们围成一个环,其中每个节点都与它左右相邻的各  $K/2$  节点相连, $K$  是偶数。

(2) 随机化加边:以概率  $p$  在随机选取的一对节点之间加上一条边。其中,任意两个不同的节点之间至多只能有一条边,并且每一个节点都不能有边与自身相连。

图 4 显示了 NW 小世界模型的演化过程。在 NW 小世界模型中, $p=0$  对应于原来的最近邻耦合网络, $p=1$  则对应于全局耦合网络。

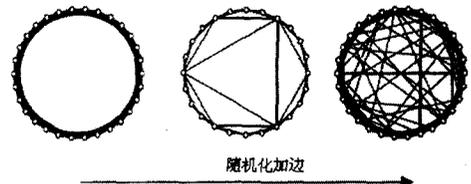


图 4 NW 小世界模型

从另一个方面我们发现:当  $K=2, p=0$  时,NW 小世界

模型中的最近邻耦合网络,就是本文上面所提到的粒子群环形社会认知拓扑结构(lBest);当  $p=1$  时得到的全局耦合网络即为粒子群的全互连社会认知拓扑结构(gBest);而整个NW小世界模型的演化过程,恰恰就是环形结构向全互连结构的演变过程。对于粒子群算法来说,如果开始能较大幅度地保留粒子的多样性,而后逐渐加快其收敛的速度,即可得到一个不易陷入局部最优又有较好收敛能力的平衡优化算法。而NW小世界模型正好符合这个设想。另外,前面所讲到的环形和全互连社会认知结构都是静态结构,而NW小世界模型是一种动态寻找邻居的过程,我们可以通过控制  $p$  的取值来调节其形成的过程和收敛的速度。

为了适应粒子群优化算法,本文对NW小世界模型构造算法做了修改,算法如下:

(1) 将  $M$  个粒子先形成环形,其中每个粒子都与它左右相邻的各  $K/2$  粒子相连。

(2) 对于每个粒子  $i$  在当前循环次数  $c\_iter$  时:当  $p_2 > p_1$ ,此粒子在它的非邻居粒子中选择一个作为邻居;否则,不增加邻居。其中,  $p_1$  为限制概率,  $p_2$  为随机概率。

通过对  $p_1$  的控制可以调整粒子增加邻居的速度。考虑到粒子群的特点,本文采用先发挥每个粒子寻找位置的能力,然后逐步收敛。限制概率  $p_1$  的控制公式如下:

$$p_1 = (\log_2(itters/c\_iter) / \log_2(itters))^{1/\log_2 K} \quad (8)$$

其中  $itters$  是预定的循环总次数,  $c\_iter$  为当前时刻的循环次数,  $K$  为初始的邻居数。图5显示了  $p_1$  随着循环次数和  $K$  的变化情况,横坐标表示  $c\_iter$ ,纵坐标表示概率  $p_1$ 。从图5可以看到,开始  $p_1$  的值很大,每个粒子增加邻居的可能性比较小;随着循环次数的增加,  $p_1$  逐渐减小,每个粒子的邻居逐渐增多;当  $p_1$  接近 0 时,粒子群的社会拓扑接近于全互连结构。由于本文还设定了随机概率  $p_2$ ,因此使得每个粒子的邻居变化不会完全相同,这也是为了在变化中保持粒子的多样性。另外,考虑到初始邻居个数对粒子群全局寻优能力和收敛性的影响,利用式(8)可以约束初始邻居数大的结构增长粒子的速度。

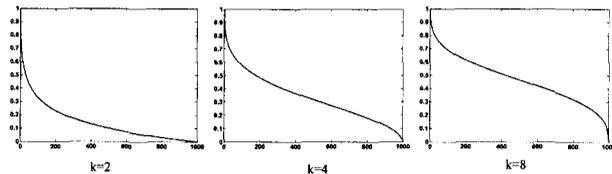


图5  $p_1$  变化图

### 3 NW小世界邻居的粒子群多阈值分割算法

首先,将修改后的Otsu推广到多阈值,其判别公式如下:

$$\sigma_w^2(\vec{T}) = \sum_{i=1}^L \sigma_i^2 = \sum_{i=1}^L \left( \sum_{j=T_{i-1}+1}^{T_i} [i - \mu_i(T_i)]^2 \frac{P(i)}{q_i(T_i)} \right) \quad (9)$$

$$\vec{T}^* = \arg \min_T \{ \sigma_w^2(\vec{T}) \} \quad (10)$$

表1 gBest, lBest 和 swBest 寻优性能比较

| 阈值级别 L | 4              | 5                   | 6                        | 7                            | 8                                 | 9                                      | 10   |
|--------|----------------|---------------------|--------------------------|------------------------------|-----------------------------------|--|--|
| 最优阈值   | (75, 119, 182) | (73, 111, 143, 184) | (73, 109, 137, 166, 189) | (65, 86, 112, 138, 166, 189) | (65, 86, 112, 137, 164, 183, 199) | (65, 85, 111, 133, 149, 167, 184, 199) | (60, 75, 90, 112, 133, 149, 167, 184, 199) |
| 最优评判值  | 633.5974       | 505.6988            | 436.4526                 | 373.9081                     | 338.5541                          | 305.7069                               | 280.5025                                   |

$$\text{其中, } q_i(T_i) = \sum_{j=T_{i-1}+1}^{T_i} P(j), \mu_i(T_i) = \sum_{j=T_{i-1}+1}^{T_i} \frac{jP(j)}{q_i(T_i)}$$

$$\vec{T} = (T_1, T_2, \dots, T_{L-1}), T_0 < T_1 < T_2 < \dots < T_{L-1} < T_L,$$

$$T_0 = 0, T_L = 255, L \text{ 为阈值级别。} \quad (11)$$

NW小世界邻居的粒子群多阈值分割算法过程如下:

(a) 初始化粒子群  $M$ :

$X_M^{-1} = \text{rand int}(M, L-1, [0, 255]), V_M^{-1}$  为零,其中  $M$  是粒子数,  $L$  是阈值级别;

(b) 对  $X_M^{-1}$  排序,使其符合式(11)的约束条件;

(c) 设置  $pBest_i = X_M^{-1}$ ;

(d) 依据修改的NW小世界构造算法,构造  $M$  个粒子的邻居结构  $SWN(M)$ ;

(e) 按照  $SWN(M)$  和式(7), (9), (10) 找到每个粒子  $i$  的邻居最佳位置  $lBest_i$  (即评判值最小的位置);

(f) 按照式(4), (5) 修改粒子的速度和位置;

(g) 对新位置  $X_M^{-1}$  排序,使其符合式(11)的约束条件;

(h) 按照式(6), (7), (9), (10) 重新计算每个粒子的  $pBest_i$  和  $lBest_i$ ;

(i) 如果达到最大循环次数或  $V_M^{-1}$  趋于零,到(k),否则到(j);

(j) 根据当前的循环次数,按照式(8) 计算限制概率  $p_1$ , 并随机产生每个粒子的随机概率  $p_2$ , 转到(d);

(k) 找到最佳的  $lBest_i$ , 即为所得结果。

### 4 实验及结果分析

实验以经典图像 Lena 为例,分别用全互连结构粒子群(gBest)、环形结构粒子群(lBest)和NW小世界邻居粒子群(swBest,  $K=2$  和  $K=4$ ) 对该图像进行从 4 到 10 级别的多阈值分割,参数设置为  $M=20, \omega=0.72984, c_1=c_2=1.5$ , 循环总次数  $itters=1000$ , 每个算法运行 100 次。表 1 和表 2 分别显示了它们在寻优能力和收敛性上的性能,其中表 1 的最优阈值是这 3 种结构得到的共同最优值。

在表 1 中随着阈值级别的增加(即维数的增加), lBest 具有最高的命中率, swBest 其次, gBest 命中率最低;表 2 则显示出 lBest 具有最差的收敛能力,这与前面分析的结果相同,然而可以看到 gBest 和 swBest 具有很接近的收敛性能,从未收敛次数来看 swBest 还优于 gBest。表 3 展示了这 3 种结构的综合性能:与 gBest 相比, swBest ( $K=2$ ) 以增加 3.66% 的循环次数为代价,换取了增加 17% 的命中率;与 lBest 相比, swBest ( $K=2$ ) 虽然降低了 13% 的命中率,但减少了 42.26% 的循环次数;而且从表 3 中,可以看到在循环 1000 次下, swBest 基本都收敛,具有最好的收敛能力。对于不同的  $K$ , 实验结果显示 swBest 保持较稳定的性能。

从实验中,可以观察到采用NW小世界模型作为社会拓扑结构的粒子群算法不仅结合了环形和全互连结构的粒子群算法的优点,而且在收敛性上具有更好的性能,是一个具有好的全局寻优能力且收敛快的平衡优化算法。

|               |       |                          |                          |                     |                      |                     |                      |                      |
|---------------|-------|--------------------------|--------------------------|---------------------|----------------------|---------------------|----------------------|----------------------|
| gBest         | 最差评判值 | 633.5974                 | 505.6988                 | 505.6988            | 441.0914             | 373.9081            | 348.3008             | 338.3675             |
|               | 评判值范围 | 633.5974±<br>1.7139e-012 | 505.6988±<br>7.9982e-013 | 439.8173±<br>9.7514 | 381.4091±<br>17.5730 | 344.2447±<br>9.9673 | 316.5552±<br>12.0272 | 292.9450±<br>14.2468 |
|               | 命中率   | 100%                     | 100%                     | 55%                 | 76%                  | 27%                 | 20%                  | 21%                  |
| lBest         | 最差评判值 | 633.5974                 | 505.6988                 | 441.2780            | 373.9218             | 348.3008            | 339.8879             | 292.9489             |
|               | 评判值范围 | 633.5974±<br>1.7139e-012 | 505.6988±<br>7.9982e-013 | 436.9411±<br>1.4532 | 373.9084±<br>0.0019  | 339.3726±<br>2.1868 | 307.6778±<br>4.2859  | 280.8580±<br>1.7224  |
|               | 命中率   | 100%                     | 100%                     | 73%                 | 98%                  | 64%                 | 43%                  | 58%                  |
| swBest<br>K=2 | 最差评判值 | 633.5974                 | 505.6988                 | 441.2780            | 401.0433             | 348.5782            | 339.8879             | 305.7187             |
|               | 评判值范围 | 633.5974±<br>1.7139e-012 | 505.6988±<br>7.9982e-013 | 438.1735±<br>2.3115 | 374.1803±<br>2.7134  | 340.7226±<br>3.2989 | 311.2397±<br>6.3029  | 283.9821±<br>6.0816  |
|               | 命中率   | 100%                     | 100%                     | 62%                 | 95%                  | 44%                 | 29%                  | 37%                  |
| swBest<br>K=4 | 最差评判值 | 633.5974                 | 505.6988                 | 441.4024            | 441.2780             | 373.9288            | 344.0550             | 313.0450             |
|               | 评判值范围 | 633.5974±<br>1.7139e-012 | 505.6988±<br>7.9982e-013 | 438.1914±<br>2.3292 | 376.1533±<br>11.5564 | 340.8635±<br>5.4430 | 311.3949±<br>7.4752  | 283.3614±<br>6.2559  |
|               | 命中率   | 100%                     | 100%                     | 58%                 | 91%                  | 41%                 | 28%                  | 41%                  |

表2 gBest, lBest 和 swBest 的收敛性能比较

|               |          | 阈值级别 L |     |     |     |     |     |     |
|---------------|----------|--------|-----|-----|-----|-----|-----|-----|
|               |          | 4      | 5   | 6   | 7   | 8   | 9   | 10  |
| gBest         | 命中平均循环次数 | 26     | 35  | 49  | 67  | 72  | 160 | 118 |
|               | 收敛平均循环次数 | 161    | 165 | 196 | 211 | 230 | 276 | 265 |
|               | 未收敛次数    | 0      | 0   | 1   | 6   | 5   | 12  | 15  |
| lBest         | 命中平均循环次数 | 40     | 61  | 132 | 124 | 159 | 337 | 332 |
|               | 收敛平均循环次数 | 184    | 208 | 311 | 362 | 477 | 583 | 575 |
|               | 未收敛次数    | 0      | 0   | 76  | 1   | 69  | 65  | 65  |
| swBest<br>K=2 | 命中平均循环次数 | 31     | 42  | 74  | 70  | 91  | 107 | 129 |
|               | 收敛平均循环次数 | 161    | 177 | 223 | 222 | 239 | 259 | 271 |
|               | 未收敛次数    | 0      | 0   | 4   | 0   | 1   | 0   | 2   |
| swBest<br>K=4 | 命中平均循环次数 | 31     | 39  | 80  | 80  | 79  | 109 | 120 |
|               | 收敛平均循环次数 | 157    | 175 | 211 | 218 | 240 | 275 | 271 |
|               | 未收敛次数    | 0      | 0   | 4   | 1   | 1   | 0   | 2   |

表3 gBest, lBest 和 swBest 综合性能比较

|               |          | 阈值级别 L |      |     |     |     |     |     | 合计(平均)    |
|---------------|----------|--------|------|-----|-----|-----|-----|-----|-----------|
|               |          | 4      | 5    | 6   | 7   | 8   | 9   | 10  |           |
| gBest         | 命中率      | 100%   | 100% | 55% | 76% | 27% | 20% | 21% | 399(57%)  |
|               | 收敛平均循环次数 | 161    | 165  | 196 | 211 | 230 | 276 | 265 | 1504(215) |
|               | 未收敛次数    | 0      | 0    | 1   | 6   | 5   | 12  | 15  | 39(6)     |
| lBest         | 命中率      | 100%   | 100% | 73% | 98% | 64% | 43% | 58% | 539(77%)  |
|               | 收敛平均循环次数 | 184    | 208  | 311 | 362 | 477 | 583 | 575 | 2700(386) |
|               | 未收敛次数    | 0      | 0    | 76  | 1   | 69  | 65  | 65  | 276(39)   |
| swBest<br>K=2 | 命中率      | 100%   | 100% | 62% | 95% | 44% | 29% | 37% | 467(68%)  |
|               | 收敛平均循环次数 | 161    | 177  | 223 | 222 | 239 | 259 | 271 | 1559(223) |
|               | 未收敛次数    | 0      | 0    | 4   | 0   | 1   | 0   | 2   | 7(1)      |
| swBest<br>K=4 | 命中率      | 100%   | 100% | 58% | 91% | 41% | 28% | 41% | 459(66%)  |
|               | 收敛平均循环次数 | 157    | 175  | 211 | 218 | 240 | 275 | 271 | 1547(221) |
|               | 未收敛次数    | 0      | 0    | 4   | 1   | 1   | 0   | 2   | 8(1)      |

**结束语** 图像多阈值分割不仅能帮助实现对象的识别,而且可用于图像压缩等技术,具有很好的研究意义和实用价值。Otsu 是阈值分割中最流行实用的方法。但是 Otsu 方法利用绝对类方差得到的阈值往往偏向于类方差大的对象,所以并不能准确地找到直方图中谷的位置,尤其对于小对象的分割效果很不理想。本文采用相对类方差计算图像的阈值,能够较好地找到谷值,对小对象的分割有较好的结果。在多阈值优化算法中,本文首次采用 NW 小世界模型作为粒子群的社会邻居,此模型通过动态地扩充邻居,达到了全局寻优能力和收敛速度的较好平衡,得到了比目前流行的粒子群拓扑结构算法更好的优化结果。

## 参考文献

- [1] Otsu N. A threshold selection method from gray-level histograms [J]. IEEE Transactions on Systems Man Cybernet, 1979, 9(1):62-66
- [2] Kittler J, Illingworth J. Minimum error thresholding [J]. Pattern Recognition, 1986, 19(1):41-47
- [3] Abutaleb A S. Automatic thresholding of grey-level pictures using two-dimensional entropy [J]. Computer Vision, Graphics and Image Process, 1989, 47(1):22-32
- [4] Soon H K. Threshold selection based on cluster analysis [J]. Pattern Recognition Letters, 2004, 25(9):1045-1050
- [5] Huang Qingming, Gao Wen, Cai Wenjian. Thresholding technique with adaptive window selection for uneven lighting image [J]. Pattern Recognition Letters, 2005, 26(6):801-808
- [6] Hemachander S, Verma A, Arora S, et al. Locally adaptive block thresholding method with continuity constraint [J]. Pattern Recognition Letters, 2007, 28(1):119-124
- [7] Qiao Yu, Hua Qingmao, Qian Guoyu, et al. Thresholding based on variance and intensity contrast [J]. Pattern Recognition, 2007, 40(2):596-608
- [8] Pal N R, Pal S K. Entropic thresholding [J]. Signal Processing, 1989, 16(2):97-108
- [9] Brink A D, Pendock N E. Minimum cross-entropy threshold selection [J]. Pattern Recognition, 1996, 29(1):179-188
- [10] Agus Z A, Akira A. Image segmentation by histogram thresholding using hierarchical cluster analysis [J]. Pattern Recognition Letters, 2006, 27(13):1515-1521
- [11] Cao Li, Bao Paul, Shi Zhongke. The strongest schema learning GA and its application to multilevel thresholding [J]. Image and Vision Computing, 2008, 26(5):716-724
- [12] Kennedy J, Eberhart R C. Particle swarm optimization [C]// Proc. IEEE Int. Conf. Neural Networks. Perth; IEEE press, 1995; 1942-1948
- [13] Watts D J, Strogatz S H. Collective dynamics of 'small-world' network [J]. Nature, 1998, 393(6684):440-442
- [14] Newman M E J, Watts D J. Renormalization group analysis of the small-world network model [J]. Physics Letters A, 1999, 263(4-6):341-346