

基于莱维飞行樽海鞘群优化算法的多阈值图像分割

邢致恺¹ 贾鹤鸣¹ 宋文龙¹

摘要 针对 Otsu 算法用于多阈值图像分割中存在运算时间长和精度低的不足, 利用群智能优化算法对图像分割算法进行优化. 本文首先应用莱维飞行算法对樽海鞘群优化算法进行改进, 将多阈值 Otsu 函数作为优化算法的适应度函数, 利用改进后的 LSSA 寻找适应度函数的最大值, 同时获得相对应的多阈值. 其次, 通过对几幅基本图像、伯克利大学图像分割库中的图像和实际油污图像进行多阈值 Otsu 分割研究, 在最佳适应度值、PSNR、SSIM 指标以及算法耗时方面进行对比分析. 实验结果表明本文提出的算法可以获得更为准确的分割阈值和更高的分割效率.

关键词 多阈值图像分割, 最大类间方差法, 樽海鞘群优化, 莱维飞行

引用格式 邢致恺, 贾鹤鸣, 宋文龙. 基于莱维飞行樽海鞘群优化算法的多阈值图像分割. 自动化学报, 2021, 47(2): 363–377

DOI 10.16383/j.aas.c180140



开放科学(资源服务)标识码(OSID):

Levy Flight Trajectory-based Salp Swarm Algorithm for Multilevel Thresholding Image Segmentation

XING Zhi-Kai¹ JIA He-Ming¹ SONG Wen-Long¹

Abstract Aiming at the shortcoming of long operation time and low precision in multi-threshold image segmentation using Otsu algorithm, image segmentation algorithm is optimized by using group intelligent optimization algorithm. In this paper, firstly, the optimized algorithm of thaliacea scabbard group is improved by using the Levy flight algorithm. The multi-threshold Otsu function is taken as the fitness function of the optimized algorithm, and the improved LSSA is used to find the maximum fitness function, and the corresponding multi-threshold value is obtained. Secondly, the multi-threshold Otsu segmentation study was conducted on several basic images, images in Berkeley university image segmentation database and actual images of pollution oil, and the comparative analysis was conducted on the optimal fitness value, PSNR, SSIM index and algorithm time. Experimental results show that the proposed algorithm can obtain more accurate segmentation thresholds and higher segmentation efficiency.

Key words Multi-threshold image segmentation, Otsu method, salp swarm optimization, Levy flight

Citation Xing Zhi-Kai, Jia He-Ming, Song Wen-Long. Levy flight trajectory-based salp swarm algorithm for multilevel thresholding image segmentation. *Acta Automatica Sinica*, 2021, 47(2): 363–377

在图像处理领域中, 图像分割是一项基础工作, 但是图像分割却又是一项极具挑战的研究课题, 图像分割主要方法有基于阈值进行分割^[1]、基于聚类进行分割^[2]、基于区域分割^[3] 以及人工神经网络^[4] 等方法. 闫成新等^[5] 提出了一种过渡区提取的多阈值图像分割方法, 利用局部熵和滤波器获得图像的多个阈值, 可以稳定地找到图像的阈值. 刘仲民等^[6]

提出了基于稀疏矩阵的聚类方法, 利用图像的信息建立稀疏相似度矩阵, 提高了分割的准确性和鲁棒性. 张军国等^[7] 提出了利用复合梯度提取图像信息, 最终利用分水岭实现图像分割, 能够有效地监控树林中虫害区域. 陈鸿翔^[8] 构建并实现了卷积神经网络和反卷积层神经网络相结合的深度神经网络结构, 直接对图像在像素水平上进行预测其所属的语义类别, 在医学图像和物体的分割有着较好的效果. 上文分析了几种常用的图像分割方法, 本文主要研究多阈值分割方法, 其核心的问题就是阈值的选取, 单阈值处理复杂图像面临较大的挑战, 故许多学者研究将单阈值推广到多阈值对图像进行分割, 以求获得更加准确的阈值.

本文研究多阈值图像分割问题, 通过将传统的 Otsu 单阈值分割推广到多阈值图像分割, 并且解决多阈值图像分割算法中存在计算精度不高和运算

收稿日期 2018-03-12 录用日期 2018-11-08
Manuscript received March 12, 2018; accepted November 8, 2018

国家自然科学基金(31470714), 中央高校基本科研业务费专项资金(2572014BB03, 2572019BF04) 资助
Supported by National Natural Science Foundation of China (31470714), Fundamental Research Funds for the Central Universities (2572014BB03, 2572019BF04)

本文责任编辑 左旺孟
Recommended by Associate Editor ZUO Wang-Meng

1. 东北林业大学机电工程学院 哈尔滨 150040
1. School of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin 150040

时间较长等问题. 为了克服上述问题, 许多学者引入元启发式优化算法对多阈值的适应度函数进行寻优. Oliva 等^[9] 提出了一种改进鲸鱼算法应用在多阈值图像分割之中, 该方法能够有效地提高鲸鱼算法的全局搜索能力, 并且能够找到更加优秀的适应度值, 以及更为精确的分割阈值. 张新雨等^[10] 提出了改进的人工鱼群算法应用到硅单晶直径检测图像阈值分割, 该方法具有较高的搜索精度, 有效地解决了该工程问题. Agarwal 等^[11] 利用蜘蛛优化算法进行多阈值分割优化, 进行直方图的双模和多模态阈值分割, 有效地提升计算时间和实现最优阈值的寻优. Erdman 等^[12] 提出利用改进的萤火虫算法进行多阈值分割, 通过模拟三个阈值级别的场景对其进行分割, 结果更加接近手动分割的效果. 上述研究将元启发式算法应用到多阈值分割中具有一定的效果, 但大多数算法均是由全局的粗搜索到局部的细搜索, 且算法设置参数较多, 影响了算法的学习效率和工程实用性, 因此从工程应用角度出发, 探索形式更简化且耗时更少的解决多阈值图像分割问题的优化算法是今后研究的重要方向之一.

由于各种元启发式算法适应解决的实际工程问题不同, 元启发式算法主要分为两种^[13], 分别为进化算法和群智能优化算法. 进化算法就是模仿自然界进化的概念. 这类算法中最经典且最受欢迎的算法是模仿达尔文进化理论的遗传算法 (Genetic algorithm, GA). 在 GA 中优化开始针对特定问题提供一套随机解决方案, 通过目标函数的评估后, 基于适应度值来解决方案的变量更新. 还有其他几种常用的演化算法, 比如差异演化、进化论策略、生物地理学优化等. 群智能优化算法源于生物的集体行为, 模仿大的种群如何寻找食物的方法. 蚁群优化算法 (Ant colony optimization, ACO)^[14] 和粒子群优化算法 (Particle swarm optimization, PSO)^[15] 是这类算法中较为经典的两类方法. ACO 模仿蚂蚁寻找最短路径的社会行为, PSO 算法模拟鸟群在航行和狩猎中的行为. 其他群优化算法有: 2015 年由 Karaboga 等提出的人工蜂群算法 (Artificial bee colony algorithm, ABCA)^[16], 通过模仿蜜蜂采花蜜的行为进行寻优, 该算法能够适应环境的变换; 萤火虫算法 (Firefly algorithm, FA)^[17] 由 Yang 等提出, 通过模仿萤火虫的趋光性, 向光源移动进行寻优; 蝙蝠优化算法 (Bat algorithm, BA)^[18] 是一种有效地全局搜索方法, 该算法通过迭代搜索最优解, 并且在最优解附近进行飞行产生局部新解, 加强了局部搜索能力; 2016 年 Mirjalili 等提出的鲸鱼优化算法 (Whale optimization algorithm, WOA)^[19] 是一种模仿座头鲸捕食的算法, 该算法在全局搜索及局部寻优均具有较强的能力. 虽然进化优化与群体

优化之间存在差别, 但其共同点是能够在限定域中找到最优值. 尽管各类算法都具有各自的优点, 但 No-free-lunch (NFL)^[20] 已经证明没有一种算法能够解决所有的优化问题, 本文研究并应用的樽海鞘群优化算法能够有效解决单目标优化学习问题, 具有较强的搜索学习能力.

本文的主要研究内容如下: 首先, 利用莱维飞行轨迹对 SSA 算法进行改进, 使得该算法具有更加优秀的全局搜索能力和更强的收敛能力, 获得更加优秀的适应度值. 其次, 将本文算法应用于多阈值图像分割领域当中, 对经典图像以及油污图像进行处理, 观察本算法与其他算法分割后的效果, 并且对相关参数指标进行分析, 说明本算法具有较强的全局搜索能力, 并且能够很好地解决实际的图像分割问题. 最后, 通过与其他图像分割算法进行比较, 验证本文所提出的图像分割算法有着较高的精度.

1 多阈值 Otsu 分割

Otsu 法是根据类间方差最大的准则选取图像最佳阈值的方法^[21]. 假设阈值 t 将灰度范围为 $[0, 1, \dots, L-1]$, 大小为 $M \times N$ 的图像分为前景和背景两部分, $h(m)$ 为灰度级 m 在图像中所占的比例, 设图像分割后的前景、背景占图像的比例分别为 ϖ_0, ϖ_1 , 平均灰度分别为 μ_0, μ_1 , 其公式如下:

$$\begin{cases} \varpi_0 = \sum_{m=0}^{L-1} h(m) \\ \varpi_1 = \sum_{m=t+1}^{L-1} m \times h(m) \end{cases} \quad (1)$$

$$\begin{cases} \mu_0 = \frac{\sum_{m=0}^{L-1} m \times h(m)}{\varpi_0} \\ \mu_1 = \frac{\sum_{m=t+1}^{L-1} m \times h(m)}{\varpi_1} \end{cases} \quad (2)$$

类间方差为:

$$\sigma_B^2(t) = \varpi_0(\mu_0 - \mu_T)^2 + \varpi_1(\mu_1 - \mu_T)^2 \quad (3)$$

其中, μ_T 为图像的平均灰度, $\mu_T = \sum_{m=0}^L m \times h(m)$, 使 $\sigma_B^2(t)$ 达到最大时 t^* 即为最大阈值. 将该公式推广至多阈值, 设阈值组合为 $[t_1, t_2, \dots, t_{K-1}]$, 将图像分割成 K 个类别, 此时类间方差表示为:

$$\sigma_B^2(t_1, t_2, \dots, t_{K-1}) = \sum_{j=0}^{K-1} \varpi_j \times (\mu_j - \mu_T)^2 \quad (4)$$

使 $\sigma_B^2(t_1, t_2, \dots, t_{K-1})$, 达到最大值的一组阈值 $[t_1, t_2, \dots, t_{K-1}]$, 即为所求的最优阈值, 传统的

多阈值 Otsu 算法中阈值的搜索过程是对灰度空间中的所有解进行穷举法计算, 需要计算 C_L^{K-1} 种组合, 每种组合内需要计算 L 次, 总计算量近似为 $O(L^K)$, 随着 K 的增加, 计算量指数级增长.

2 基于莱维飞行轨迹的改进樽海鞘群的图像分割算法

樽海鞘是地球上最大的海洋生物群体, 属于纽鳃樽科, 而且具有透明的身体, 形态与水母十分相似^[22]. 本算法有效地模拟了它在海洋中游弋和觅食的过程. 它们的移动方式与水母也十分相似, 而且很难在实验环境中饲养研究, 其中最有趣的就是它们的群居觅食行为. 在深海中樽海鞘经常形成一种樽海鞘链的运动行为, 如图 1 所示. 产生上述行为的主要生体信息原因尚不明确, 但一些研究人员已经研究并发现这是为了更好地利用快速协调变化来实现更好的运动轨迹, 实现觅食的优化行为.

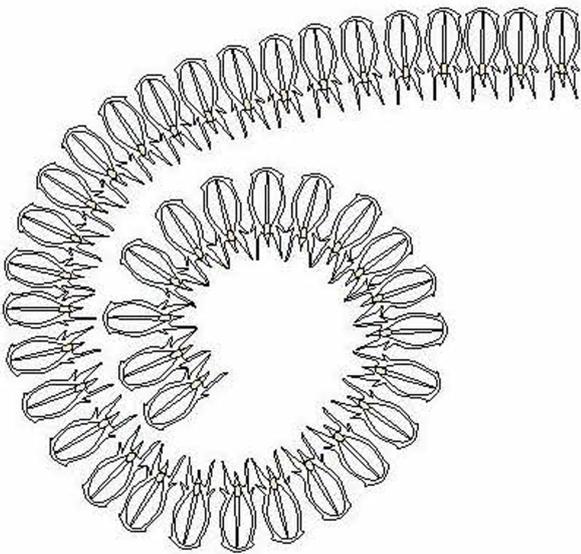


图 1 樽海鞘链
Fig.1 Salp chain

为了建立樽海鞘群运动觅食的轨迹模型, 首先将种群分为领导者和追随者两类. 领导者是在樽海鞘链的顶端, 而其他的樽海鞘视为追随者. 正如其命名的含义一样, 樽海鞘的领导者负责领导群体运动, 而追随者则进行跟随. 该模型与其他群体智能优化技术一样, 樽海鞘的位置是在 n 维搜索空间中定义的, 其中 n 是所求问题给定的变量. 因此, 所有樽海鞘的位置都存储在 x 的二维矩阵中. 在搜索空间中设置 F 为食物源 (即为需要求解的适应度函数), 作为樽海鞘搜寻食物的目标.

樽海鞘链的觅食运动轨迹的数学模型如下:

$$\mathbf{x}_j^1 = \begin{cases} \mathbf{F}_j + c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) \times c_2 + \mathbf{lb}_j), & c_3 \geq 0 \\ \mathbf{F}_j - c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) \times c_2 + \mathbf{lb}_j), & c_3 < 0 \end{cases}$$

$$\mathbf{x}_j^i = \frac{1}{2}(\mathbf{x}_j^i - \mathbf{x}_j^{i-1}), \quad i \geq 2 \quad (5)$$

其中, \mathbf{X}_j^1 表示第一个樽海鞘的在 j 维度位置, $i \geq 2$, \mathbf{X}_j^i 表示第 i 个樽海鞘追随者在第 j 维度的位置, \mathbf{F}_j 表示在 j 维度食物源的位置, \mathbf{ub}_j 为搜索空间的向量, \mathbf{lb}_j 为搜索空间的最小向量, c_1, c_2, c_3 为随机数. 式 (5) 表明, 领导者只是更新其对食物源的位置. 其中参数 c_1 是本算法中最重要的参数, 定义如下:

$$c_1 = 2e^{-\left(\frac{l}{L}\right)^2} \quad (6)$$

其中, l 为算法当前的迭代次数, L 为算法的总迭代次数. c_2, c_3 为 $[0, 1]$ 之间的随机数. 实际上 c_2, c_3 决定了樽海鞘在第 j 个维度的下一个位置应该是正无穷大还是负无穷大以及樽海鞘前进的步长.

2.1 莱维飞行

实际上莱维飞行是一种随机步长描绘莱维分布的方法^[23-24]. 大量的研究表明, 许多动物和昆虫的行为表现是莱维飞行的经典特征. 莱维飞行是一种特殊的随机步长的方法, 如图 2 是关于莱维飞行轨迹的模拟图像, 它的步长总是很小, 但是偶尔也会出现大的跳动.

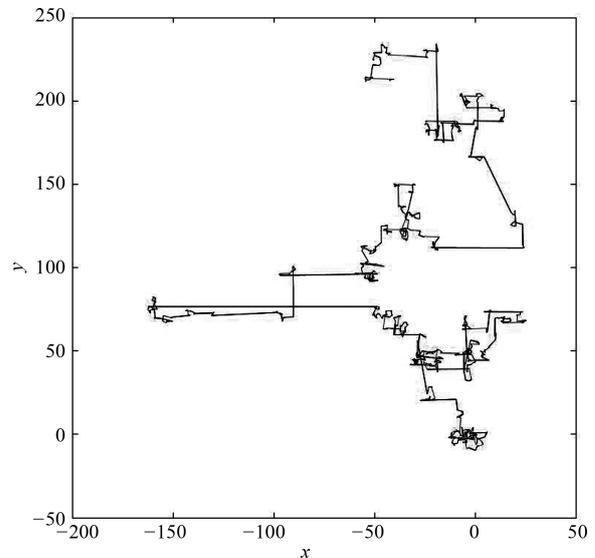


图 2 莱维飞行轨迹
Fig.2 Levy's fight path

莱维飞行的公式如下:

$$\text{Levy} - u = t^{-\lambda}, \quad 1 < \lambda \leq 3 \quad (7)$$

莱维飞行的步长公式为:

$$s = \frac{\mu}{|v|^{\frac{1}{\beta}}} \quad (8)$$

其中, 参数 $\beta = 1.5$, $\mu = N(0, \sigma_\mu^2)$ 和 $v = N(0, \sigma_v^2)$ 都表示 gamma 函数.

参数的方差为:

$$\sigma_\mu = \left[\frac{\Gamma(1 + \beta) \times (\sin \pi \times \frac{\beta}{2})}{\Gamma[\frac{(1+\beta)}{2}] \times \beta \times 2^{\frac{(\beta-1)}{2}}} \right]^{\frac{1}{\beta}}, \sigma_v = 1 \quad (9)$$

由于莱维飞行具有增加种群多样性和扩大搜索范围的特性, 将其加入式 (5) 中, 更好地完成樽海鞘领导者位置更新, 使其可以更容易跳出局部最优问题.

2.2 基于莱维飞行的樽海鞘群优化算法

樽海鞘优化算法 (由 Mirjalili 提出) 可以简单、高效地解决低维单模优化问题. 然而在处理高维和复杂的图像分割时, 传统 SSA 所得到的解并不十分理想. 为了提高 SSA 的全局搜索能力, 本文提出了一种改进的莱维飞行樽海鞘群优化算法. 莱维飞行可以最大限度地实现搜索域的多样化, 这样能够保证算法高效地搜索食物源的位置, 实现局部最优. 这一发现表明, 利用莱维飞行有助于 SSA 获得更好的寻优结果, 因此将式 (7) 结合式 (5) 进行改进, 可用如下数学公式进行表达:

$$\mathbf{X}_j^1 = \begin{cases} \mathbf{F}_j + c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) + \mathbf{lb}_j) \times \text{Levy}, & c_3 \geq 0 \\ \mathbf{F}_j - c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) + \mathbf{lb}_j) \times \text{Levy}, & c_3 < 0 \end{cases} \quad (10)$$

式 (10) 中 c_1 由式 (6) 求得, 该参数能够随迭代次数的增加使优化算法进行收敛, 更加靠近适应度函数的最佳值. 由参数 c_1 和莱维飞行结合, 既加强了 SSA 算法的全局搜索能力, 又能够及时收敛, 确保找到最优值. 这种方法不仅提高了 SSA 的搜索强度, 也提高了算法的多样化. 通过优化算法保证该法能够找到算法最优值并且避免陷入局部最优, 通过增强多样性使该算法具有更好的全局搜索能力. 可以更好地解决多阈值图像分割问题具有较好的效果.

2.3 基于莱维飞行的樽海鞘群的图像分割算法

为解决多阈值 Otsu 图像分割算法运算时间长和分割精度低的问题, 对多阈值的搜索过程进行优

化, 可用如下公式表达 LSSA 应用于多阈值 Otsu 图像分割算法的数学模型:

$$\mathbf{x}_j^1 = \begin{cases} \mathbf{F}_j + c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) \times \text{Levy} + \mathbf{lb}_j), & c_3 \geq 0 \\ \mathbf{F}_j - c_1 \times ((\mathbf{ub}_j - \mathbf{lb}_j) \times \text{Levy} + \mathbf{lb}_j), & c_3 < 0 \end{cases}$$

$$\mathbf{x}_j^i = \frac{1}{2} \mathbf{x}_j^i - \mathbf{x}_j^{i-1}, \quad i \geq 2$$

$$\mathbf{F}_j = \sum_{j=0}^{K-1} \varpi_j \times (\mu_j - \mu_T)^2 \quad (11)$$

本文采用樽海鞘优化算法对阈值的搜索过程进行求解最优值, 将对图像进行多阈值分割的问题转化为对目标函数进行最优解求取的问题. 所以, 本文将式 (4) 作为樽海鞘优化算法的适应度函数, 即为式 (11) 中的 \mathbf{F}_j , 作为樽海鞘领导者的食物源, 通过樽海鞘群对食物捕食, 更加快速找到式 (4) 的最大值, 此时, 得到的 $[t_1, t_2, \dots, t_{K-1}]$ 即为图像的多阈值.

由于樽海鞘群算法结合多阈值 Otsu 算法时存在容易陷入局部最优的问题, 导致分割的结果存在过分割现象, 影响分割的质量. 故对其加入莱维飞行增强算法的全局搜索能力, 通过式 (7) 对式 (5) 进行改进, 进而获得式 (10), 使樽海鞘群能够跳出局部最优, 更好地找到全局最优解. 下面给出本文阈值分割算法的整体实现流程图, 如图 3 所示.

3 多阈值图像分割的实验与分析

为验证本文算法的分割能力, 分别选择经典图像、伯克利大学图像分割库中的图像和无人机巡航飞行采集拍摄的油污图像进行图像分割研究, 并应用本文算法与 WOA、SSA、PSO 和声搜索算法 (Harmony search algorithm, HSA) 和花授粉算法 (Flower pollination algorithm, FPA) 几种经典的优化算法对多阈值分割方法进行寻优, 进行实验验证与对比分析. 由无人机拍摄的油污图像, 存在噪声大、目标与背景区域差别不大及运算速度慢等实际问题. 实验硬件条件为 MATLAB2016b 环境下, 在 2.7 GHz 和 8 GB 内存微处理器上进行的.

3.1 经典图像多阈值分割实验

本节对 Lena 图、Baboon 图、Moon 图、Camera 图、Plane 图和 Tank 图进行多阈值分割, 采用优化算法对最佳阈值分割点进行寻优, 获得每张图像的最佳阈值, 本节选取图像如图 4 所示.

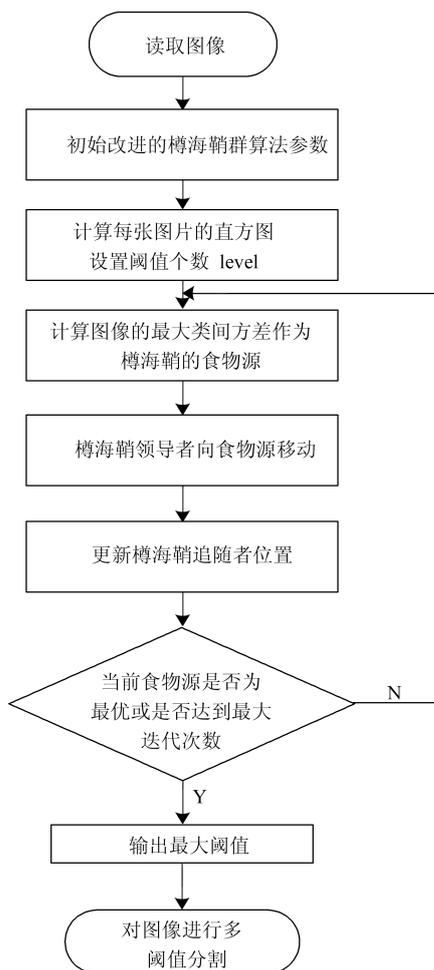


图 3 图像分割算法流程图

Fig. 3 Image segmentation algorithm flow chart

为了验证本文算法优秀地搜寻最佳阈值能力, 将本文的优化算法与 WOA、SSA、PSO、HSA 和 FPA 经典算法进行比较, 更为准确地分析各算法的优劣, 将每种算法的最大迭代次数设置为 1000, 初始种群设置为 30. 每种算法的基本参数如表 1 所示.

对所有的实验图片选取的阈值个数 K 为: 2, 3, 4, 5. 为了验证本算法在多阈值分割上的优秀能力, 本节通过比较每个算法在经典图像上进行多阈值分割的时间和最佳适应度值, 实验结果如表 2~4 所示. 当阈值个数 $K = 2$ 时, 所有的算法取值几乎相同, LSSA 适应度值上有所提升, 而 PSO 和 FPA 算法适应度值最差. 当阈值个数 $K = 3$ 时, LSSA、WOA 和 SSA 的值基本是相同的, PSO、FPA 和 HSA 的数值则相对较差. 当阈值个数 $K = 4, 5$ 时, LSSA 的效果有明显的提升, 通过最佳适应度值以及分割后的图像对比效果能够看出 LSSA 比其他算法都优秀. 从分割阈值和最佳适应度值这个两个指标能够看出虽然搜索后的结果浮动不大, 但是 LSSA 算法对所测试的所有图片均能够

进行有效的寻优, 在处理更加复杂的图片上, LSSA 的优化性能更能得到体现.



图 4 经典图像

Fig. 4 Classic images

表 1 元启发式算法的参数

Table 1 Parameter of the heuristic algorithm

算法	参数	取值
WOA	a	[0, 2]
	b	1
	l	[-1, 1]
SSA	c_2	rand
	c_3	rand
PSO	c_1	1.5
	V	[0, 1]
HSA	$HMCR$	0.7
	PAR	0.3
	PAR_{min}	0.3
	PAR_{max}	0.9
	bw_{min}	0.2
FPA	bw_{max}	0.5
	P	0.5
LSSA	莱维参数	1.5

表 2 各算法所用时间 (s)

Table 2 The time of each algorithm (s)

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Lena	2	2.0194	1.1280	2.3447	2.151	2.2726	1.2192
	3	2.1376	1.3220	2.5441	3.5147	3.6616	1.4381
	4	2.3473	1.5417	2.7315	5.4181	5.4681	1.6668
	5	2.5166	1.7181	3.0027	7.5123	7.8495	1.9037
Moon	2	1.8974	1.3124	2.2654	2.1011	2.1043	1.1867
	3	1.9821	1.5134	2.5297	3.5741	3.6502	1.4522
	4	2.0864	1.8076	2.8298	5.1547	5.5808	1.6469
	5	2.1856	2.0871	3.9691	7.7145	7.9073	1.8979
Baboon	2	1.6136	1.3094	2.3135	2.1421	2.1442	1.1925
	3	1.8123	1.5775	2.5898	3.5741	3.7015	1.4971
	4	2.0219	1.7995	2.7694	5.6241	5.6584	1.7795
	5	2.2238	2.0742	3.0042	7.8145	7.9067	2.0095
Camera	2	1.5753	1.0552	2.0963	1.4321	1.5753	1.2465
	3	1.7649	1.2233	2.2171	1.9451	1.7649	1.4252
	4	2.3980	1.4006	3.4061	2.5124	2.3980	1.6983
	5	2.4131	1.5882	3.6251	2.6412	2.4131	1.8411
Plane	2	2.3117	1.6997	2.5147	2.5421	2.5117	1.6197
	3	2.4719	1.6959	2.8412	3.1741	2.8749	1.6759
	4	2.6711	1.7831	3.2145	4.1241	2.9711	1.7731
	5	2.9613	2.1693	3.8417	5.1145	3.1633	1.9693
Tank	2	2.1714	1.8784	2.6415	2.4321	2.2714	1.5784
	3	2.3113	1.9103	2.9451	2.9451	2.6123	1.7103
	4	2.4024	2.1094	3.3171	3.5124	2.9024	1.8094
	5	2.8117	2.3817	3.9541	4.6412	3.2117	1.8817

3.2 伯克利图像多阈值分割实验

本节对伯克利大学图像分割库中的图像进行实验分析,限于篇幅,本文只列出了 Test1~Test8 图进行多阈值分割,如图 5 所示.采用优化算法对阈值的搜索过程进行寻优,求得每幅图像的最佳阈值.本节通过对复杂的图像进行实验,验证了本文改进后的算法能够有效地解决 SSA 算法优化求解过程中存在的局部最优问题.

由表 5 中最佳适应度函数值可以看出,随着阈值个数的增加,适应度函数值逐步增加,而当群智能算法陷入局部最优的时候,适应度函数值可能不会增长,导致选取的阈值比较分散,无法精准地将图像分割为多个区域.由表 5 中 SSA 对 Test2 和 Test4 两幅图像在阈值个数 $K = 5$ 时获得的最佳阈值可以看出,这 5 个阈值比较集中并且接近最大灰度值,同时,适应度函数值反而比在阈值个数 $K = 4$ 时降低,说明 SSA 对复杂的图像进行寻优的时候容易陷入局部最优,而 LSSA 算法有效地解决了这个问题,从数据上看,寻找的阈值较为稳定,而且适应度函数值均表现较为优秀.综上所述,本文提出的 LSSA 算法能够更好地解决多阈值 Otsu 算法中阈值搜寻的问题,并且有效地避免陷入局部最优,更好地找到适

应度函数的最大值,同时给出图像的最佳阈值.



(a) Test1



(b) Test2



(c) Test3



(d) Test4



(e) Test5



(f) Test6



(g) Test7



(h) Test8

图 5 伯克利图像

Fig. 5 Berkeley images

3.3 本文算法与其他算法实验

为了更好地验证本文提出的算法在图像分割领域中的效果,故将本文算法与其图像分割算法进行对比,分别应用文献 [25] 提出改进的 FCM 算法、传统 Otsu 算法、文献 [26] 提出的模糊熵算法、文献 [27] 提出的基于区域生长的图像分割方法 (Multi-scale region growing, MSRG)、文献 [28] 提出的基于贝叶斯区域统计的自适应图像分割模型 (Region competition, RC) 和本文提出的 LSSA 算法 (选择阈值个数 $K = 5$) 对 4 幅图像进行实验分析.本节选取上文中实验的图片进行实验分析,分割后的图像如图 6~9 所示.

表 3 各优化算法的最佳分割阈值
Table 3 Optimal segmentation threshold of each optimization algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Lena	2	93 151	93 151	94 152	94 152	93 151	93 151
	3	80 126 171	80 126 171	84 128 168	84 128 168	80 138 168	80 126 171
	4	75 114 146 181	75 114 146 181	67 113 142 191	69 110 143 193	73 113 142 179	75 114 146 181
	5	73 109 136 160 188	6 593 121 149 182	55 102 143 173 205	59 107 140 179 205	72 111 139 163 182	65 93 120 148 182
Moon	2	57 152	57 152	57 152	57 152	56 151	57 152
	3	41 110 178	41 110 178	42 111 178	42 111 178	39 112 174	41 110 178
	4	3 493 142 198	3 492 141 198	34 103 146 198	39 105 142 198	33 116 140 175	34 93 142 198
	5	2 878 117 153 204	2 877 116 153 204	2 162 104 138 205	2 064 101 138 205	30 78 120 146 207	2 876 115 152 204
Baboon	2	98 150	98 150	97 149	97 149	97 150	98 150
	3	85 125 161	85 125 161	67 108 160	67 108 160	84 124 160	85 125 161
	4	72 106 137 168	72 106 137 168	61 105 132 161	69 102 132 161	69 107 138 166	72 106 137 168
	5	6 799 125 150 175	6 697 123 148 174	64 102 126 156 184	66 104 124 156 184	70 102 128 149 174	6 799 125 150 175
Camera	2	70 144	70 144	70 144	70 144	70 144	70 144
	3	59 119 156	59 119 156	59 119 156	59 119 156	54 109 152	59 119 156
	4	4 396 140 170	4 295 140 170	41 92 140 170	4 295 140 170	47 106 140 170	4 293 140 170
	5	3 683 122 149 173	3 682 122 149 173	35 83 122 149 173	3 683 122 149 173	3 885 103 141 171	3 683 122 149 173
Plane	2	26 179	126 179	126 179	126 179	126 179	126 179
	3	85 142 174	84 141 178	85 143 175	85 144 175	85 141 173	85 145 178
	4	82 144 171 183	81 142 171 182	79 138 171 182	84 141 174 185	84 141 174 185	84 141 174 185
	5	81 132 163 171 181	83 132 163 171 183	81 130 156 172 181	77 122 151 161 185	82 131 160 174 185	82 131 160 174 185
Tank	2	96 134	96 134	96 134	96 134	96 134	96 134
	3	82 121 142	87 121 144	81 102 134	89 112 134	88 129 154	88 122 144
	4	71 102 138 138	76 106 131 148	76 118 152 189	78 108 132 149	98 109 142 159	78 108 132 149
	5	6 593 119 137 143	6 592 116 132 146	6 791 107 114 138	6 693 117 134 148	4 673 127 154 188	6 693 117 134 148

表 4 各优化算法的最佳适应度值
Table 4 Optimum fitness value of each optimization algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Lena	2	1 962.9788	1 962.9788	1 962.7213	1 962.7841	1 962.9788	1 962.9788
	3	2 129.5118	2 129.5118	2 121.1345	2 127.0668	2 107.4125	2 129.5118
	4	2 193.1658	2 193.1658	2 175.0012	2 176.0021	2 192.0198	2 193.1658
	5	2 219.0787	2 217.7731	2 175.9015	2 184.9068	2 215.3949	2 219.0787
Moon	2	4 488.1684	4 488.1684	4 488.1684	4 488.1684	4 487.9365	4 488.1684
	3	4 635.5642	4 635.5642	4 631.1234	4 635.5614	4 633.7889	4 636.7907
	4	4 705.9054	4 705.8884	4 688.5124	4 699.5635	4 655.2737	4 705.9054
	5	4 736.3511	4 736.3721	4 710.0314	4 727.0397	4 731.9581	4 736.3277
Baboon	2	1 559.9789	1 559.9789	1 559.8912	1 559.8972	1 559.9723	1 559.9789
	3	1 651.5989	1 651.5989	1 613.8143	1 631.8262	1 651.5354	1 651.3989
	4	1 705.5302	1 725.5302	1 701.3112	1 701.3283	1 704.0964	1 705.5302
	5	1 731.7870	1 731.6784	1 712.5131	1 722.5437	1 730.9155	1 731.7877
Camera	2	3 651.5573	3 651.5573	3 651.5573	3 651.5573	3 651.5573	3 651.5573
	3	3 726.9841	3 726.9841	3 726.9841	3 726.9841	3 724.7608	3 726.9841
	4	3 782.0173	3 872.0395	3 781.0311	3 782.0391	3 778.4987	3 782.0395
	5	3 813.3775	3 813.3769	3 813.3775	3 813.3775	3 799.5781	3 813.3775
Plane	2	408.3514	411.1475	416.1014	411.0124	400.1204	416.3834
	3	427.5317	437.2471	457.2478	447.2474	424.1478	457.5537
	4	438.5733	459.2357	469.3457	457.1247	438.2044	469.5933
	5	454.8162	464.1789	474.6457	469.8751	475.5751	474.8202
Tank	2	611.4421	611.1247	631.1240	611.2407	601.2104	631.4447
	3	641.2147	652.5789	671.3745	641.3047	641.3578	671.2437
	4	680.4747	671.5478	690.4577	680.4701	670.4528	690.4043
	5	690.1457	693.0124	703.1045	691.1023	693.9014	703.1382

表 5 各优化算法的最佳适应度值
Table 5 Optimum fitness value of each optimization algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Test1	2	7 891.2936	7 891.2936	7 891.2936	7 891.2936	7 891.2936	7 891.2936
	3	8 087.8384	8 087.8384	8 014.3372	8 024.3372	8 084.3372	8 087.8384
	4	8 172.9962	8 172.9962	8 072.9653	8 122.9653	8 172.9653	8 172.9962
	5	8 215.5841	8 215.5841	8 005.2923	8 139.8311	8 199.8311	8 215.4419
Test2	2	326.2732	326.2732	326.1873	321.1873	326.1873	326.2732
	3	375.7722	375.7722	365.6812	370.6812	375.6812	375.7722
	4	408.6649	408.6993	398.2891	401.2891	408.2891	408.6945
	5	422.1188	407.3658	404.9742	414.9742	424.9742	425.8495
Test3	2	1 437.6581	1 437.6581	1 437.5427	1 437.5427	1 437.5427	1 437.6581
	3	1 562.7039	1 562.7039	1 460.9753	1 510.9753	1 550.9753	1 562.7039
	4	1 627.4583	1 627.456	1 525.0731	1 585.0731	1 625.0731	1 627.4273
	5	1 664.2046	1 664.1218	1 521.0817	1 619.2961	1 649.2961	1 664.0444
Test4	2	1 355.9394	1 355.9394	1 355.9394	1 355.9394	1 355.9394	1 355.9394
	3	1 448.3263	1 448.3263	1 397.9881	1 407.9881	1 447.9881	1 448.3263
	4	1 491.6077	1 491.6077	1 427.9804	1 437.9804	1 487.9804	1 491.5784
	5	1 492.1746	1 478.1969	1 483.6063	1 493.6063	1 513.6063	1 483.8625
Test5	2	3 102.2450	3 122.2145	3 022.1045	3 012.0124	3 112.2748	3 122.2871
	3	3 209.2104	3 249.1245	3 149.2786	3 149.4577	3 189.2741	3 249.0745
	4	3 287.1247	3 297.8161	3 217.2014	3 247.2547	3 207.3045	3 317.8161
	5	3 312.1274	3 312.5482	3 252.5758	3 302.7885	3 302.1278	3 352.5482
Test6	2	821.1024	831.1042	801.2371	801.3457	831.0245	831.8096
	3	905.2785	935.5921	905.0543	915.0214	915.5781	955.5921
	4	985.7852	1 001.2456	927.0578	987.2787	997.3857	1 007.4113
	5	1 001.2457	1 018.4278	948.3542	1 018.0122	1 008.4527	1 028.6463
Test7	2	515.3857	565.4257	505.0248	505.2788	525.0245	565.1769
	3	587.0245	608.0125	571.2578	578.3942	575.6781	608.1853
	4	606.5728	616.1410	595.2452	606.2015	606.2015	626.7067
	5	616.4527	626.4527	601.0276	616.3782	616.4578	636.4758
Test8	2	704.3458	716.6664	684.1275	701.0217	671.2573	716.6664
	3	718.9524	768.4527	715.3857	748.8377	718.0214	768.3891
	4	751.4205	781.0124	752.7821	771.0215	751.5789	791.4562
	5	795.4527	800.1045	783.0274	795.3781	795.2452	805.5394

由 Lena 图、Baboon 图和伯克利图像分割后的图像可以看出本文提出的 LSSA 算法相比其他 3 种方法目标区域分割更加清晰, 从 Lena 图和 Test6 图中可以看出将人物的脸部轮廓以及背景的一些细节分割的更加清晰, 而其他方法中存在原始图像信息的缺失或过分割现象. 在伯克利图像中 Test7 图可以明显地将图像中的飞机从图像中分割, 分割后的图像展现的信息更为清晰, 而其他算法仍存在原始图像信息的丢失现象. 因此说明本文提出的算法更加高效, 能够成功分割不同图像, 得到较为完整的目标区域.

为了更好地验证本文提出的算法的优秀性能, 在多种评价方法中, 本章选用了较为常见的客观评

价标准, 并对本次的实验结果进行了分析评价. 下面对本文应用的评价方法进行简单介绍.

1) 香农熵

香农熵可以衡量一幅图像中信息量的多少, 香农熵结果越大, 说明包含的信息量越多, 分割的精确度越高. 香农熵可以表示为:

$$H = -P_0 \times \log_2(P_0) - P_1 \times \log_2(P_1) \quad (12)$$

由算法分割结果图像为二值图, 只包含 0 和 1. 公式中 P_0 表示分割后的图像 Y 输出的是 0 的概率, P_1 表示分割后的图像 Y 输出的是 1 的概率.

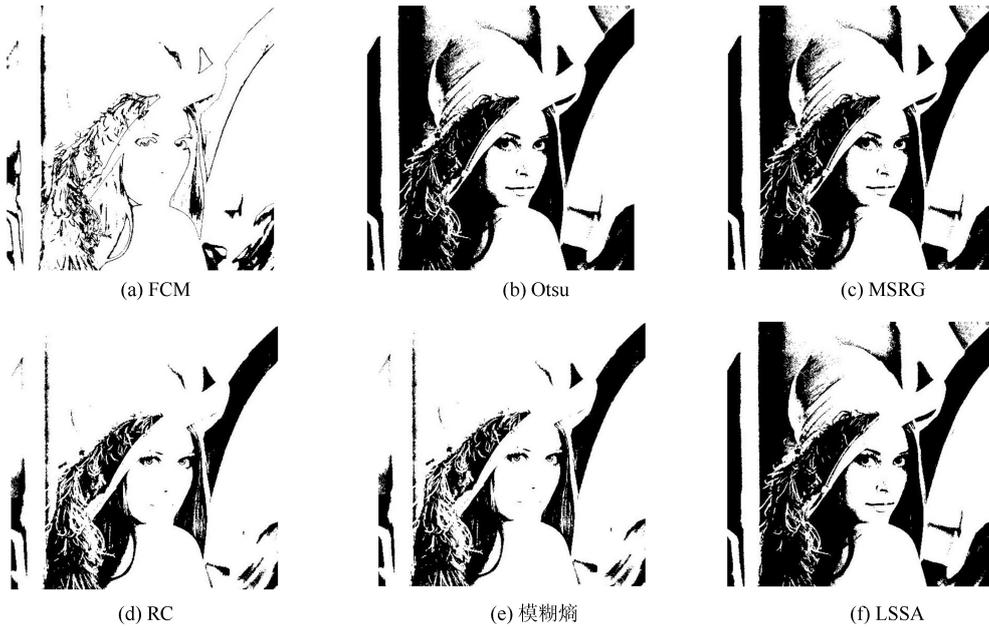


图 6 Lena 图仿真

Fig.6 Lena figure simulation

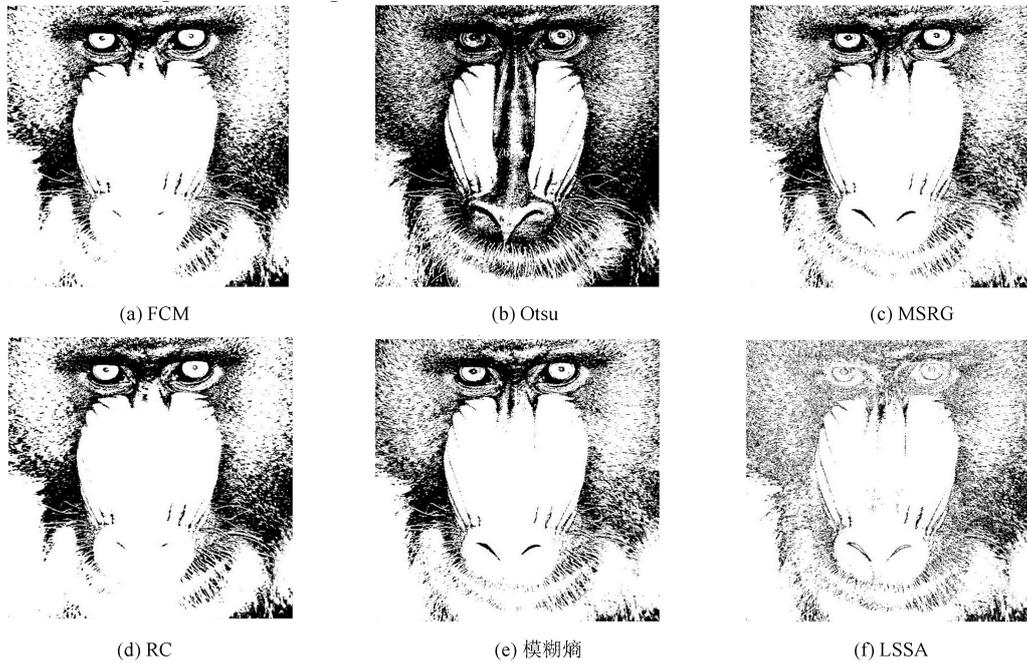


图 7 Baboon 图仿真

Fig.7 Baboon figure simulation

2) 区域一致性评价

图像分割由规定的标注进行分割, 把图像划分为不同的多个相邻域. 每个子区域都是根据这个特点标准划分的, 所以相邻域之间存在一定的相似性. 区域一致性 U 就是用来验证各个子区域内相似性的程度, 当 U 越大, 表示同一区域内所有像素的相似性越大. 该评价指标能较好地评价图像分割的结果,

区域一致性 U 的公式如下:

$$U = 1 - \frac{\sigma_0^2 + \sigma_1^2}{C_1} \tag{13}$$

式中: $\sigma_i^2 = \sum_{(x,y) \in R_i} (f(x,y) - \psi_i)^2$, $\psi_i = \sum_{(x,y) \in R_i} f(x,y) / A_i$, A_i 为图像中分割区域的像

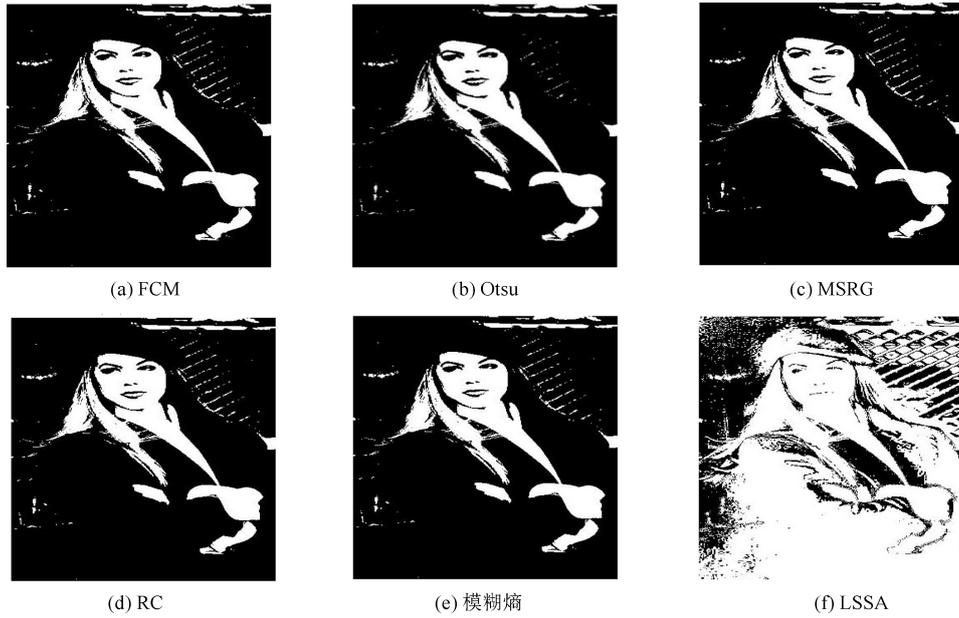


图 8 Test6 图仿真

Fig. 8 Test6 figure simulation

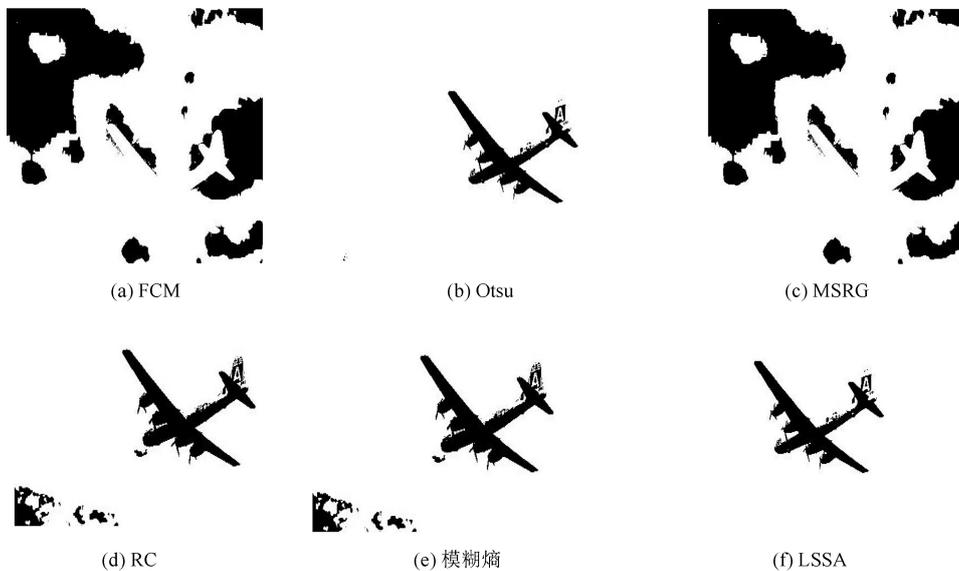


图 9 Test7 图仿真

Fig. 9 Test7 figure simulation

素, C_1 表示整幅图像的像素.

3) 区域对比度评价

区域对比度是指不同区域间的差异应该尽可能大, 这种区域间的对比度由区域对比度函数进行评价, 此指标能够较好地评测图像分割效果, 公式如下:

$$C = \frac{|f_0 - f_1|}{f_0 + f_1} \quad (14)$$

其中, f_0 表示目标区域的平均灰度值, f_1 表示背景区域的平均灰度值.

由表 6 中的香农熵值可知, FCM、Otsu 和 MSRSG 在该指标上较低, 反映两种算法分割后图像的信息较少, 未能对目标区域进行有效分割. 模糊熵和 RC 优于 FCM, 数值有所提升但不是很明显. LSSA 算法的香农熵均高于其他对比算法, 说明该算法分割后图像的包含目标区域的信息较多, 成功地将目标区域从背景中进行了分割.

由表 7 可以知道本节实验的各算法求得区域一致性值. FCM、Otsu 和 MSRSG 从数值上看相差不大, 说明分割后的目标区域一致性较差, 分割的目标

区域不完整, 存在欠分割现象. LSSA 算法模型的数值较高, 说明分割后的目标区域较为完整, 成功地将目标区域从背景中进行了分割.

表 6 各算法的香农熵值

Table 6 The Shannon entropy of each algorithm

图像	FCM	Otsu	模糊熵	MSRG	RC	LSSA
Lena	0.8157	0.8021	0.8112	0.8524	0.8614	0.9514
Baboon	0.8251	0.8154	0.8354	0.8414	0.8517	0.9317
Test6	0.8314	0.8231	0.8047	0.8358	0.8421	0.8821
Test7	0.8318	0.8057	0.8012	0.8304	0.8407	0.8907

表 7 各算法区域一致性值

Table 7 The regional consistency value of each algorithm

图像	FCM	Otsu	模糊熵	MSRG	RC	LSSA
Lena	0.8325	0.8514	0.8654	0.8514	0.8681	0.9414
Baboon	0.8258	0.8412	0.8517	0.8415	0.8631	0.9217
Test6	0.8197	0.8357	0.8481	0.8517	0.8758	0.8921
Test7	0.8314	0.8341	0.8617	0.8718	0.8811	0.8907

区域对比度数据记录在表 8 中, 从表中可以看到各算法的区域对比度相差不是很大, 但是在 Test6 和 Test7 这两幅复杂图像的结果中, FCM 和 Otsu 算法的分割结果较差, 分割精度不高. LSSA 在分割图像时, 分割区域对比度值优于其他算法, 并且在分割 Test6 和 Test7 这两幅复杂图像时区域对比度值最优.

表 8 各算法区域对比值

Table 8 Ratio of each algorithm region

图像	FCM	Otsu	模糊熵	MSRG	RC	LSSA
Lena	0.4024	0.4024	0.4131	0.4021	0.4231	0.4214
Baboon	0.4124	0.4124	0.4258	0.4317	0.4428	0.4117
Test6	0.4258	0.4025	0.4189	0.4257	0.4527	0.4421
Test7	0.4028	0.4157	0.4358	0.4318	0.4612	0.4707

从表 9 中可知, 本文算法在分割实验图像所用时间相比其他算法更快, 尤其比传统 Otsu 算法运行有着明显的提升, 成功地解决了多阈值 Otsu 分割算法中存在计算时间慢的问题, 因此说明本文提出的算法在保证了分割精度的同时减少了运算时间.

从以上四个指标可以看出本文提出的 LSSA 算法在图像分割中的应用较为稳定, 能够精确地分割图像的目标区域. 通过对比经典的图像分割算法分割后的效果图, 并且计算各算法的香农熵、区域一致性值、CPU 运行时间和区域对比度值, 明显地发现通过 LSSA 优化算法改进的多阈值 Otsu 算法在

图像分割应用中有着明显地提升. 从实验结果可知, LSSA 算法在分割复杂图像时的能力优于其他较为常用的分割算法, 可以处理复杂的图像分割任务.

表 9 各算法所用时间

Table 9 The time of each algorithm

图像	FCM	Otsu	模糊熵	MSRG	RC	LSSA
Lena	3.5166	17.5778	3.6457	3.8457	4.1266	1.9037
Baboon	4.2238	17.0147	3.4527	3.7527	5.3438	2.0095
Test6	5.9131	15.4457	3.8131	5.1131	6.9671	1.8411
Test7	4.5613	15.5789	3.4527	5.4517	6.5613	1.9693

4 油污图像多阈值分割实验

为验证本文算法在实际情况中的应用效果, 将其与 WOA、SSA、PSO、HSA 和 FPA 算法进行对比仿真实验, 并且给出每种算法的实验结果以及相关数据. 为解决油污图像分割的实际问题, 选取的油污图像是通过大疆精灵 4 对大庆油田采油二厂地区进行巡检实地拍摄采集的油污图片, 作为本次实验数据样本. 首先对采集图像进行预处理, 对有效图片进行自适应霍夫曼压缩, 将 4000×2250 的待处理图像压缩至 400×225 , 并且进行归一化处理, 对图像采用中值平均滤波的方法进行优化, 提高样本的图像质量. 实验表明, 本方法对不同的油污图像进行分割都十分有效, 可以根据不同的图像寻找合适的参数, 从而得到满意的分割后的油污图像. 限于篇幅, 仅给出 4 幅典型油污图像的研究结果, 如图 10 所示.

从图 10 中可以明显地看出 Oil1 和 Oil2 两张图中的油污区域与背景有明显的差异, 而 Oil3 和 Oil4 则与背景差异不大, 这样使算法的寻优能力受到限制, 并且会增加算法的计算时间. 为了更好地解决油污图像分割问题, 通过以下 4 个指标对各优化算法的性能进行评判:

1) 各算法的运行时间.

2) 每个算法的适应度函数最佳值.

3) 为了更直观地评判本文算法的优劣, 以信噪比准则作为定量分析算法性能的指标计算分割后图像与原图像的 PSNR 值. PSNR (dB) 公式定义如下:

$$\text{PSNR} = 20 \lg \left(\frac{255}{\text{RMSE}} \right) \quad (15)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N \sum_{j=1}^N (\mathbf{I}(i, j) - \hat{\mathbf{I}}(i, j))^2}{M \times N}} \quad (16)$$

其中 $\mathbf{I}(i, j)$, $\hat{\mathbf{I}}(i, j)$ 分别表示尺寸为 $M \times N$ 的原图

像和分割后的图像。

4) 通过计算图像的结构相似性, 评判分割后的图像和原图像的相似性. SSIM 公式如下:

$$SSIM(I, \hat{I}) = \frac{(2\mu_I\mu_{\hat{I}} + c_1) \times (2\sigma_{I,\hat{I}} + c_2)}{(\mu_I + \mu_{\hat{I}} + c_1) \times (\sigma_I + \sigma_{\hat{I}} + c_2)} \quad (17)$$

其中, μ_I 为原图像的平均值, $\mu_{\hat{I}}$ 为分割后图像的平均值, σ_I 是原图像的方差, $\sigma_{\hat{I}}$ 是分割后图像的方差, $\sigma_{I,\hat{I}}$ 为原图像和分割后图像的协方差. c_1, c_2 是两个常量, 本文取值为: $c_1 = 6.4025$ 和 $c_2 = 6.4025$. SSIM 的值越高说明该算法在多阈值分割应用上的性能越好. 本实验各算法的参数与上节中设置的参数一样.

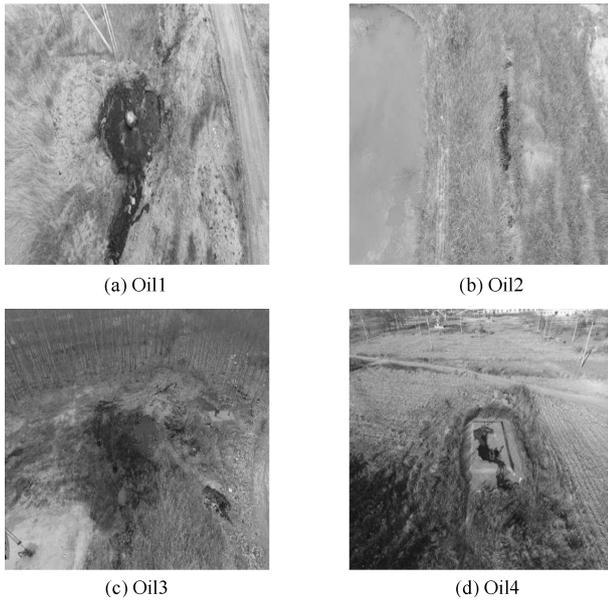


图 10 污油图像

Fig. 10 Dirty of oil images

从表 4 和表 5 中的数据可以看到, 当阈值个数 $K = 2, 3$ 时, 各元启发式算法的最佳适应度值基本上是一样的, 因为计算难度是与阈值个数的增加成正比的, 所以在阈值选取较少的时候, 各算法都顺利地找到最佳阈值. 当阈值个数 $K = 4, 5$ 时, LSSA 的优势展现出来, 能够找到较其他算法更优的最佳适应度值, 有效地提高了多阈值分割的效果. 通过分析能够证明 LSSA 算法不仅适用于结构简单的图像分割, 并且在复杂图像上的分割有更好的效果.

为了更好地对该算法进行评价, 表 10 给出了各算法在处理污油图像所用的时间, 由于选取的图像具有复杂的背景, 给图像分割带来较大的困难, 相应的计算时间会增加, 而 LSSA 在时间上优于 HSA、FPA、PSO 算法, 并且与 WOA、SSA 算法时间所用相近, 在保证实验结果的准确性的同时, 提升了算法的运算效率, 说明 LSSA 算法优于本文

中进行试验的其他典型的元启发式算法.

表 11 给出了对所有实验后分割的图像计算 PSNR 值的结果. 当阈值个数 $K = 2, 3$ 时, 所有算法的 PSNR 值基本一样, 获得效果图没有明显差别, 但是当阈值个数 $K = 4, 5$ 时, LSSA 在多阈值分割上的性能优于其他算法, 对 Oil1 和 Oil2 这种目标与背景区域差别较大的图像, 数值有所提升; 对 Oil3 和 Oil4 这种目标和背景区域差别较小的图像, 数值有明显的提升, 并且从分割效果角度进行分析, 能够有效地将油污区域分割出来. 表明该算法不仅能够适应多维度的计算, 还可处理复杂图像, 有效地解决实际工程问题.

表 10 各算法所用时间

Table 10 The time of each algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Oil1	2	1.5813	1.3501	10.8044	1.4131	1.5813	1.2690
	3	1.8284	1.4181	11.0703	1.8145	1.8284	1.4165
	4	1.9591	1.6535	11.5977	1.9874	1.9591	1.6501
	5	2.1831	1.8763	12.0414	2.2514	2.1831	1.8584
Oil2	2	1.6183	1.2056	11.0774	1.6124	1.6183	1.2093
	3	1.8057	1.4853	11.2452	1.8421	1.8057	1.5087
	4	1.9716	1.7181	11.5426	1.9587	1.9716	1.6898
	5	2.1823	1.9337	11.9453	2.2151	2.1823	1.9916
Oil3	2	1.5915	1.1830	14.4043	1.5467	1.5915	1.2289
	3	1.7729	1.4196	14.3581	1.7951	1.7729	1.4492
	4	1.9602	1.7168	14.8511	1.9641	1.9602	1.6507
	5	2.1482	1.9474	15.0878	2.1054	2.1482	1.9004
Oil4	2	1.6269	1.2696	1.7499	1.6844	1.6269	1.1873
	3	1.8117	1.5481	2.0056	1.9541	1.8117	1.4476
	4	2.0206	1.8283	2.2152	2.1547	2.0206	1.8023
	5	2.2196	2.0961	2.4264	2.5123	2.2196	1.9617

表 12 显示所有优化算法的 SSIM 值, 随着阈值的增加, 分割后的图像与原图像的相似性在逐步提高, 说明多阈值图像分割方法在图像处理上有着明显的作用, 能够有效地将图像分割为几个区域, 更好地为后续的图像处理提供支持. 本文的 LSSA 算法分割后的图像与原图的相似度最高, 说明该算法有效地完成了区域分割任务, 为后续进行辨别油污区域奠定了基础.

为了更好地选取油污区域, 对本文算法在阈值个数 $K = 5$ 时获得的结果图像进一步处理. 将每个独立、完整的油污区域分割出来, 需要对多阈值分割后的图像进行形态学操作和填洞等处理. 具体操作步骤如下:

1) 将污油图像分割结果图中阈值个数 $K = 5$ 的分割后图片进行处理, 将分割后的阈值区间从小

表 11 各算法的 PSNR 值
Table 11 PSNR value of each algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Oil1	2	15.4928	15.4928	15.4129	15.7529	15.492	15.4928
	3	17.8281	17.828	13.6561	17.7283	17.8926	17.8281
	4	19.5611	19.5811	18.8672	19.8867	19.4862	19.5651
	5	20.8643	21.0431	16.6292	21.3415	20.7961	20.9544
Oil2	2	12.2654	12.2654	12.2654	13.3473	12.0936	12.2654
	3	16.7488	16.7488	18.7909	16.9435	16.7466	16.7488
	4	18.6808	18.9998	16.2054	20.9003	18.0668	18.9129
	5	21.4732	21.6717	19.7711	23.0068	21.4817	21.4732
Oil3	2	15.1441	15.1441	15.8394	15.1441	15.1441	15.1441
	3	18.1237	18.1237	10.5491	18.1456	18.1606	18.1237
	4	20.1087	20.1087	14.6293	19.8004	19.6802	20.1087
	5	22.2648	22.4092	9.6173	20.2118	22.0937	22.2648
Oil4	2	15.6292	15.6292	15.5639	15.6292	15.6292	15.6292
	3	17.9204	17.9204	14.8809	18.2441	18.0359	17.9204
	4	19.6612	19.6612	15.4654	18.9829	19.8258	19.6612
	5	20.7293	20.8209	13.4597	20.6193	20.5241	20.9209

表 12 各算法的 SSIM 值
Table 12 SSIM value of each algorithm

图像	K	WOA	SSA	PSO	HSA	FPA	LSSA
Oil1	2	0.4492	0.5529	0.4492	0.4492	0.4492	0.5529
	3	0.6019	0.6441	0.6019	0.6019	0.6016	0.6741
	4	0.6927	0.7389	0.6927	0.6927	0.6617	0.7489
	5	0.7518	0.7786	0.7516	0.7518	0.7526	0.7986
Oil2	2	0.3241	0.5608	0.3241	0.3241	0.3241	0.5608
	3	0.4888	0.7191	0.4888	0.4888	0.5073	0.7291
	4	0.5903	0.7455	0.5903	0.5945	0.5945	0.7755
	5	0.6491	0.8292	0.6559	0.6683	0.6616	0.8492
Oil3	2	0.3349	0.4455	0.3349	0.3349	0.3393	0.4655
	3	0.4831	0.6082	0.4831	0.4831	0.4884	0.6182
	4	0.5868	0.7036	0.5868	0.5868	0.5602	0.7036
	5	0.6701	0.7677	0.6701	0.6701	0.6663	0.7777
Oil4	2	0.3924	0.4759	0.3924	0.3924	0.3924	0.4759
	3	0.5412	0.6061	0.5412	0.5412	0.5139	0.6061
	4	0.6389	0.6586	0.6389	0.6389	0.6287	0.6886
	5	0.6974	0.7302	0.6974	0.6995	0.6907	0.7402

到大依次设置灰度值为 $0, 1, 2, \dots, O$ (O 为最大阈值个数, 本文为 5), 得到灰度图像.

2) 对步骤 1) 获得的图像中灰色部分转换成黑色, 即将图像中灰度值大于 0 的点的灰度值赋值为 1, 得到二值化图像.

3) 将步骤 2) 获得的图像进行膨胀和闭运算操作, 将油污区域完整地区分出来, 腐蚀掉与油污无关的区域.

4) 对步骤 3) 处理后的图像进行填洞操作, 使油污区域填充完整, 得到填洞后的图像.

5) 对步骤 4) 获得的图像取反, 并且求得最大区域, 最大区域即为油污区域. 将该二值矩阵与原图相乘可获得油污区域, 如图 11 所示.

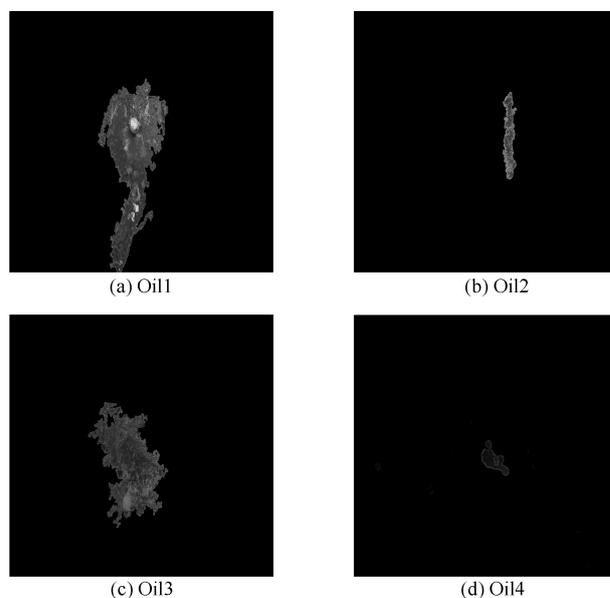


图 11 油污的分割图像
Fig. 11 The segmentation image of dirty oil

通过观察图 11 可知, 能够有效地将油污区域分割出来, 说明多阈值图像分割在油污问题处理上有着较大的作用. 本文的 LSSA 算法在油污图像分割效果上优于其他的元启发式算法, 为确定油污区域提供很大帮助. 所以, LSSA 算法不仅能够以较快地运行时间结束寻优, 同时能够更好地进行全局搜索, 比 SSA 算法获得的值更优, 能够胜任图像背景更加复杂的环境分割问题, 更加准确地找到最优阈值.

5 结论

本文提出了基于莱维飞行的樽海鞘群优化算法来确定多阈值分割的最优阈值, 并将其作为优化问题进行研究. 为了解决多阈值分割中阈值个数增加, 计算难度增加, 寻优精度不高等问题, 通过应用改进的樽海鞘群优化算法能够有效地进行图像分割, 并且通过与 WOA、SSA、PSO、HSA 和 FPA 等算法进行对比, 发现本文算法的全局搜索能力, 以及最优值搜索能力更强, 可以完成复杂图像的有效分割, 为解决油污图像的分割提供了有效的方法. 但是本文算法在时间上, 比 SSA、WOA 算法运行时间相对较慢. 在未来, 将会研究本算法在更高的阈值寻优上的能力, 并尝试在动态多阈值问题中使用该算法, 以求获得更好的分割效果. 以及解决大规模处理图像时处理速度问题, 将会研究多核 CPU/众核 CPU 技术^[29-30] 提高运算速度, 在保证分割精度的情况下进一步减少图像分割的运算时间.

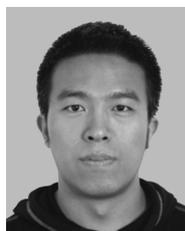
References

- Zhao Feng, Zheng Yue, Liu Han-Qiang, Wang Jun. Multi-population combined multi-objective evolutionary adaptive threshold image segmentation algorithm. *Computer Application Research*, 2018, **35**(6): 1858–1862
(赵凤, 郑月, 刘汉强, 王俊. 多种群联合的多目标进化自适应阈值图像分割算法. *计算机应用研究*, 2018, **35**(6): 1858–1862)
- Xu Qiu-Ye, Li Yu, Lin Wen-Jie, Zhao Quan-Hua. Remote sensing image segmentation based on information clustering. *Journal of China University of Mining & Technology*, 2017, **46**(1): 209–214
(徐秋晔, 李玉, 林文杰, 赵泉华. 基于信息聚类的遥感图像分割. *中国矿业大学学报*, 2017, **46**(1): 209–214)
- Zhang Fan, Zhang Xin-Hong. Segmentation algorithm of distance regularization level set based on dislocation theory. *Acta Automatica Sinica*, 2018, **44**(5): 943–952
(张帆, 张新红. 基于位错理论的距离正则化水平集图像分割算法. *自动化学报*, 2018, **44**(5): 943–952)
- Tian Juan-Xiu, Liu Guo-Cai, Gu Shan-Shan, Ju Zhong-Jian, Liu Jin-Guang, Gu Dong-Dong. Research and challenge of deep learning methods for medical image analysis. *Acta Automatica Sinica*, 2018, **44**(3): 401–424
(田娟秀, 刘国才, 谷珊珊, 鞠忠建, 刘劲光, 顾冬冬. 医学图像分析深度学习方法研究与挑战. *自动化学报*, 2018, **44**(3): 401–424)
- Yan Cheng-Xin, Sang Nong, Zhang Tian-Xu. Multi-threshold image segmentation based on transitional zone extraction. *Journal of Huazhong University of Science and Technology (Natural Science Edition)*, 2005, **33**(1): 65–67
(闫成新, 桑农, 张天序. 基于过渡区提取的多阈值图像分割. *华中科技大学学报 (自然科学版)*, 2005, **33**(1): 65–67)
- Liu Zhong-Min, Li Zhan-Ming, Li Bo-Hao, Hu Wen-Jin. Spectral clustering image segmentation algorithm based on sparse matrix. *Journal of Jilin University (Engineering and Technology Edition)*, 2017, **47**(4): 1308–1313
(刘仲民, 李战明, 李博皓, 胡文瑾. 基于稀疏矩阵的谱聚类图像分割算法. *吉林大学学报 (工学版)*, 2017, **47**(4): 1308–1313)
- Zhang Jun-Guo, Feng Wen-Zhao, Hu Chun-He, Luo You-Qing. Composite gradient watershed algorithm for image segmentation of forest insect pests in aerial photography by uav. *Transactions of the Chinese Society of Agricultural Engineering*, 2017, **33**(14): 93–99
(张军国, 冯文钊, 胡春鹤, 骆有庆. 无人机航拍林业虫害图像分割复合梯度分水岭算法. *农业工程学报*, 2017, **33**(14): 93–99)
- Chen Hong-Xiang. Image Semantic Segmentation Based on Convolution Neural Network [Master thesis]. Zhejiang University, China, 2016
(陈鸿翔. 基于卷积神经网络的图像语义分割 [硕士学位论文]. 浙江大学, 中国, 2016)
- Oliva D, Aziz M A E, Hassanien A E. Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. *Applied Energy*, 2017, **200**(1): 141–154
- Zhang Xin-Yu, Liu Ding, Yang Wen, Yang Yan-Xi. Detection of single crystal silicon diameter based on artificial fish group hoff transformation. *Chinese Journal of Scientific Instrument*, 2014, **35**(4): 940–947
(张新雨, 刘丁, 杨文, 杨延西. 基于人工鱼群霍夫变换的单晶硅直径检测. *仪器仪表学报*, 2014, **35**(4): 940–947)
- Agarwal P, Singh R, Kumar S, Bhattacharya M. Social spider algorithm employed multi-level thresholding segmentation approach. In: *Proceedings of the 1st International Conference on Information and Communication Technology for Intelligent Systems*. Berlin, Germany: Springer, 2016: 249–259
- Erdmann H, Wachs-Lopes G, Gallao C, Ribeiro M P, Rodrigues P S. A study of a firefly meta-heuristics for multithreshold image segmentation. In: *Proceedings of Developments in Medical Image Processing and Computational Vision*. Berlin, Germany: Springer, 2015: 279–295
- Xiao Hui-Hui, Wan Chang-xuan, Duan Yan-Ming, Tan Qian-Lin. Flower pollination algorithm based on gravity search mechanism. *Acta Automatica Sinica*, 2017, **43**(4): 576–594
(肖辉辉, 万常选, 段艳明, 谭黔林. 基于引力搜索机制的花朵授粉算法. *自动化学报*, 2017, **43**(4): 576–594)
- Ilie S, Badica C. Multi-agent approach to distributed ant colony optimization. *Science of Computer Programming*, 2013, **78**(6): 762–774
- Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: *Proceedings of the 6th International Symposium on Micro Machine and Human Science*. New York, USA: IEEE, 2002: 39–43
- Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 2007, **39**(3): 459–471

- 17 Yang X S, He X. Firefly algorithm: Recent advances and applications. *International Journal of Swarm Intelligence*, 2013, **1**(1): 36–50
- 18 Yang X S. A new metaheuristic bat-inspired algorithm. *Computer Knowledge & Technology*, 2010, **284**(1): 65–74
- 19 Mirjalili S, Lewis A. The whale optimization algorithm. *Advances in Engineering Software*, 2016, **95**(1): 51–67
- 20 Wolpert D H, Macready W G. No free lunch theorems for optimization. *IEEE Trans on Evolutionary Computation*, 1997, **1**(1): 67–82
- 21 Onumanyi A J, Onwuka E N, Aibinu A M, et al. A modified Otsu's algorithm for improving the performance of the energy detector in cognitive radio. *AEU - International Journal of Electronics and Communications*, 2017, **79**(1): 53–63
- 22 Li Qing, Tang Huan, Chi Jian-Nan, Xing Yong-Yue, Li Hua-Tong. Study on hand gesture segmentation based on improved maximum between-group variance method. *Acta Automatica Sinica*, 2017, **43**(4): 528–537
(李擎, 唐欢, 迟健男, 邢永跃, 李华通. 基于改进最大类间方差法的手势分割方法研究. *自动化学报*, 2017, **43**(4): 528–537)
- 23 Martine Chevrollier. Radiation trapping and Lévy flights in atomic vapours: an introductory review. *Contemporary Physics*, 2012, **53**(3): 227–239.
- 24 Imkeller P, Pavlyukevich I. Lévy flights: transitions and meta-stability. *Journal of Physics A General Physics*, 2006, **39**(15): 237–246
- 25 Tang Tu-Sheng, Chen Xuan-Qing. Research and implementation of hyperfuzzy entropy ULPCNN binary image segmentation algorithm. *Laser Magazine*, 2016, **37**(1): 113–116
(唐土生, 陈绚青. 基于超模糊熵 ULPCNN 二值图像分割算法研究与实现. *激光杂志*, 2016, **37**(1): 113–116)
- 26 Li Chang-Xing, Xue Xin-Wei, Wu Cheng-Mao. An improved fuzzy clustering image segmentation algorithm. *Journal of Xi'an University of Posts and Telecommunications*, 2017, **22**(5): 28–36
(李昌兴, 薛新伟, 吴成茂. 一种改进的模糊聚类图像分割算法. *西安邮电大学学报*, 2017, **22**(5): 28–36)
- 27 Xiao Ming-Yao, Li Xiong-Fei, Zhang Xiao-Li, Zhang Liu. Image segmentation algorithm based on multi-scale regional growth. *Journal of Jilin University (Engineering and Technology Edition)*, 2017, **47**(5): 1591–1597
(肖明尧, 李雄飞, 张小利, 张刘. 基于多尺度的区域生长的图像分割算法. *吉林大学学报 (工学版)*, 2017, **47**(5): 1591–1597)
- 28 Wang Lin-Juan, Wang Xi-Li. Fast image segmentation algorithm based on level set based on regional competition. *Journal of Computer Applications*, 2008, **28**(10): 2628–2632
(王琳娟, 汪西莉. 基于区域竞争的水平集快速图像分割算法. *计算机应用*, 2008, **28**(10): 2628–2632)
- 29 Zhou Y, He F, Hou N, Qiu Y. Parallel ant colony optimization on multi-core SIMD CPUs. *Future Generation Computer Systems*, 2017, **79**(2): 473–483
- 30 Zhou Y, He F, Qiu Y. Dynamic strategy based parallel ant colony optimization on GPUs for TSPs. *Science China (Information Sciences)*, 2017, **60**(6): 1–3



邢致恺 东北林业大学机电工程学院硕士研究生. 主要研究方向为智能优化与图像处理. E-mail: kai230606@163.com
(**XING Zhi-Kai** Master student at the School of Mechanical and Electrical Engineering, Northeast Forestry University. His research interest covers intelligent optimization and image processing.)



贾鹤鸣 东北林业大学机电工程学院副教授. 主要研究方向为智能优化与图像处理, 非线性控制理论与应用. 本文通信作者.

E-mail: jiaheminglucky99@126.com
(**JIA He-Ming** Associate professor at the School of Mechanical and Electrical Engineering, Northeast Forestry University. His research interest covers intelligent optimization and image processing, nonlinear control theory and application. Corresponding author of this paper.)



宋文龙 东北林业大学机电工程学院教授. 主要研究方向为智能检测与控制技术. E-mail: wlsong139@126.com
(**SONG Wen-Long** Professor at the School of Mechanical and Electrical Engineering, Northeast Forestry University. His research interest covers intelligent detection and control technology.)