

基于改进斑点鬣狗优化算法的红外图像分割

李唐兵¹, 胡锦涛², 周求宽¹

(1. 国网江西省电力公司电力科学研究院, 江西 南昌 330096; 2. 国网上海浦东供电公司, 上海 200122)

摘要: 针对斑点鬣狗优化算法 (spotted hyena optimizer, SHO) 容易陷入局部最优解、求解质量低等缺点, 本文提出使用 Lévy 飞行和单纯形搜索算法改进 SHO (spotted hyena optimizer based on simplex method and Lévy flight, Lévy_SM_SHO)。将 Lévy_SM_SHO 与 Lévy 飞行斑点鬣狗优化算法 (spotted hyena optimizer based on Lévy flight, Lévy_SHO)、单纯形搜索斑点鬣狗优化算法 (spotted hyena optimizer based on simplex method, SM_SHO) 和 SHO 在测试函数上结果进行对比, 实验证明改进算法能够取得较好的优化结果。并将 Lévy_SM_SHO 算法用于红外图像阈值分割问题, 通过与粒子群算法 (particle swarm optimization, PSO) 分割结果对比, 证明 Lévy_SM_SHO 算法能够取得较好的阈值分割结果。

关键词: Lévy 飞行; 单纯形搜索算法; Lévy_SM_SHO; Lévy_SHO; SM_SHO

中图分类号: TN219 **文献标识码:** A **文章编号:** 1001-8891(2021)10-0994-09

Infrared Image Segmentation Based on Improved Spotted Hyena Optimizer

LI Tangbing¹, HU Jinhong², ZHOU Qiukuan¹

(1. Power Science Research Institute of State Grid Jiangxi Electric Power Company, Nanchang 330096, China;

2. State Grid Shanghai Pudong Power Supply Company, Shanghai 200122, China)

Abstract: Based on the shortcomings of the spotted hyena optimizer (SHO), falling into a local optimal solution or a low-quality solution is easy. In this study, the Lévy flight and simplex method are proposed to improve the SHO(Lévy_SM_SHO). Comparing Lévy_SM_SHO to Lévy flight spotted hyena optimizer (Lévy_SHO), simplex method spotted hyena optimizer (SM_SHO), and spotted hyena optimizer (SHO) on the test function, the experiment proves that the improved algorithm can achieve better optimization results. Finally, the Lévy_SM_SHO algorithm is applied to the infrared image threshold segmentation problem. By crosschecking the segmentation results with the particle swarm optimization algorithm (PSO), we proved that the Lévy_SM_SHO algorithm can achieve better threshold segmentation results.

Key words: lévy flight, simplex search algorithm, Lévy_SM_SHO, Lévy_SHO, SM_SHO

0 引言

在过去几十年里,现实生活问题和工程问题的复杂性增加,使得人们需要更好技术来解决这些实际需求。元启发算法应运而生,它与其他现有的经典技术相比,由于其效率高和复杂性低,逐渐变得十分流行。

元启发优化算法大致可分为 3 类,如基于进化的、基于物理的和基于群智能的优化算法。基于进化的优化算法,它是受到生物进化的启发,如繁殖、突变、重组和选择。如遗传算法 (GA)^[1-2]、进化策略 (ES)^[3]、生物地理算法 (BBO)^[4]和差分进化算法

(DE)^[5]。基于物理的优化算法是模拟自然界的物理现象,随机产生的个体根据物理规则在空间内进行寻优,如模拟退火 (SA)^[6]、引力搜索算法 (GSA)^[7]、和声搜索算法 (HS)^[8]、中心力优化算法 (CFO)^[9]和水波优化算法 (WWO)^[10]。基于群智能的优化算法是模拟自然界中动物群体的行为,通过种群的信息共享和合作行为实现优化,如蚁群优化 (ACO)^[11]、粒子群优化 (PSO)^[12]、人工蜂群算法 (ABC)^[13]、萤火虫算法 (FA)^[14]、磷虾群算法 (KH)^[15]和灰狼优化算法 (GWO)^[16]等。

斑点鬣狗优化算法 (spotted hyena optimizer,

收稿日期: 2020-11-23; 修订日期: 2021-01-25.

作者简介: 李唐兵 (1983-), 男, 高级工程师, 研究方向电力设备故障诊断。E-mail: 63463723@qq.com.

基金项目: 国网江西省电力公司科技项目 (52182016001S)。

SHO) 是 Dhiman 等受到斑点鬣狗的狩猎行为启发, 提出了一种新的元启发式算法^[17]。SHO 算法具有易操作、参数少、稳定性好、求解精度高等优点, 已经被应用于解决现实约束工程优化设计问题^[18]、非线性连续优化问题^[19]和图像阈值分割问题^[20]。该算法通过摇摆因子 \vec{B} 和收敛因子 \vec{E} 调整全局搜索能力和局部搜索能力, 而且算法中包围猎物和狩猎行为两个阶段种群位置更新公式相似, SHO 算法也具有与其他智能算法类似的容易陷入局部最优解、求解质量低等缺点。而在大多数复杂优化问题中, 要求优化算法遍历全局最优解可能的分布范围以寻求全局最优。

文献[20]中使用混沌初始化策略、非线性收敛因子调整策略、莱维飞行策略以及精英反向学习策略等多种策略来改进鬣狗优化算法, 这就会导致算法计算量的增大。本文采用一种判断算法是否陷入局部最优的机制来决定是否对种群使用 Lévy 飞行和单纯形搜索法进行更新, 该机制能够减少种群计算量和能够寻找全局最优的特点, 在测试函数中对改进算法进行验证最后将其应用到红外图像阈值分割中。

1 斑点鬣狗优化算法

斑点鬣狗依靠它们之间的社会关系和协作行为, 能够迅速有效地捕食猎物。用搜索猎物、包围猎物、狩猎行为和攻击猎物 4 个基本步骤来模拟斑点鬣狗优化算法, 并建立这 4 个步骤的数学建模。

1.1 包围猎物

斑点鬣狗可以熟悉猎物的位置并包围它们。为了对斑点鬣狗的社会等级进行数学建模, 认为当前最佳包围对象是目标猎物或者接近最优的目标, 因为搜索空间不是先验的。其他个体会尝试更新自己的位置, 在确定了最佳包围对象后, 对最佳包围对象进行猎捕, 这种行为的数学模型由下面的方程表示:

$$\vec{D}_h = |\vec{B} \cdot \vec{P}_p(x) - \vec{P}(x)| \quad (1)$$

$$\vec{P}(x+1) = \vec{P}_p(x) - \vec{E} \cdot \vec{D}_h \quad (2)$$

式中: \vec{D}_h 为猎物与斑点鬣狗的距离; x 为当前迭代; \vec{B} 是为摇摆因子; \vec{E} 是收敛因子; $\vec{P}_p(x)$ 为当前猎物的位置向量; $\vec{P}(x)$ 是当前斑点鬣狗的位置向量, $\vec{P}(x+1)$ 是下次斑点鬣狗的位置向量。||和·分别是绝对值和向量点乘。

\vec{B} 和 \vec{E} 的计算公式如下:

$$\vec{B} = 2 \cdot r d_1 \quad (3)$$

$$\vec{E} = 2 \vec{h} \cdot r d_2 - \vec{h} \quad (4)$$

$$\vec{h} = 5 - (\text{Iteration} * (5 / \text{Max}_{\text{Iteration}})) \quad (5)$$

式中: \vec{h} 是控制因子, 为了使目标函数的搜索和开发平衡, 在最大迭代次数的过程中, \vec{h} 从 5 减小到 0。 $r d_1$ 和 $r d_2$ 为[0,1]之间随机数。

1.2 狩猎行为

斑点鬣狗通常群居狩猎, 依靠一群可信赖的朋友和识别猎物位置的能力。为了精确地定义斑点鬣狗的行为, 假设最佳的搜索个体, 无论哪个是最优的, 都知道猎物的位置。其他搜索个体组成一个集群, 信任的朋友群, 向最好的搜索个体, 并保存到目前为止获得的最好的狩猎方案来更新他们的位置。狩猎行为的数学模型为:

$$\vec{D}_h = |\vec{B} \cdot \vec{P}_h - \vec{P}_k| \quad (6)$$

$$\vec{P}_k = \vec{P}_h - \vec{E} \cdot \vec{D}_h \quad (7)$$

$$\vec{C}_h = \vec{P}_k + \vec{P}_{k+1} + \cdots + \vec{P}_{k+N} \quad (8)$$

式中: \vec{P}_h 是斑点鬣狗的第一个最佳位置; \vec{P}_k 表示其他斑点鬣狗的位置; N 表示斑点鬣狗的数量, \vec{C}_h 是 N 个最优解的集群。其中 N 的计算如下:

$$N = \text{Counts}_{\text{nos}}(\vec{P}_h, \vec{P}_{h+1}, \vec{P}_{h+2}, \cdots, \vec{P}_h + M) \quad (9)$$

M 是一个随机向量[0.5,1]; nos 是小于 $\vec{P}_h + M$ 的可行解得数量并计算所有可行解, 与搜索空间的最优解相似; \vec{C}_h 是 $N+1$ 个最优解的集合。

1.3 攻击猎物

攻击猎物可以减少向量 \vec{h} 的值, 在仿真模拟过程中, 向量 \vec{E} 也随着 \vec{h} 的改变向量而改变。攻击猎物的数学公式描述如下:

$$\vec{P}(x+1) = \frac{\vec{C}_h}{N} \quad (10)$$

1.4 寻找猎物

斑点鬣狗算法主要是根据斑点鬣狗群的位置来寻找猎物。它们互相远离, 寻找和攻击猎物。当收敛因子 $|\vec{E}| > 1$ 时, 斑点鬣狗将分散开, 远离当前猎物。这种机制允许 SHO 算法进行全局搜索。 \vec{B} 包含随机值, 提供了猎物的随机权重, 有助于 SHO 算法搜索和避免局部最优解, 使 SHO 算法满足终止条件。

2 改进鬣狗优化算法 (Lévy_SM_SHO)

通过查阅智能算法的相关文献, 越来越多的算法

研究者们将 Lévy 飞行应用于智能算法中,以此来改善搜索和开发能力的平衡性。本文提出用 Lévy 飞行^[21-25]和单纯形搜索法^[26-29]提高算法的搜索和开发能力。

2.1 Lévy 飞行

在智能算法高维空间中引入 Lévy 飞行,增大空间搜索能力,能更好地对空间进行开发。Lévy 飞行增强了群体的多样性,避免智能算法陷入局部最优解,从而获得真正的全局最优。

Lévy 飞行本质上是一个随机步长服从 Lévy 分布的随机行走, Lévy 飞行的方差随时间呈现指数的关系, Lévy 分布为:

$$\text{Lévy}(\beta) \sim u = t^{-\beta}, \quad (1 < \beta < 3) \quad (11)$$

式(11)计算 Lévy 飞行步长 s :

$$s = \frac{\phi \times u}{|v|^{1/\beta}} \quad (12)$$

式中: u 和 v 服从标准正态分布; ϕ 为正态分布的方差; Γ 是一个标准的伽玛函数, $\beta=1.5$, ϕ 为:

$$\phi = \left[\frac{\Gamma(1+\beta) \times \sin(\pi \times \beta / 2)}{\Gamma[(1+\beta)/2] \times \beta \times 2^{(\beta-1)/2}} \right] \quad (13)$$

本文,在斑点鬣狗个体位置更新后,使用下式对斑点鬣狗执行一次 Lévy 飞行:

$$\vec{P}(x+1) = \vec{P}(x+1) + \text{random} \oplus \text{Lévy}(\beta)(\vec{P}(x+1) - \vec{P}_{\text{best}}(x)) \quad (14)$$

式中: random 表示一个随机数; M_i^t 为第 i 只个体第 t 次的位置; $\text{Lévy}(\beta)$ 为随机搜索; $\vec{P}_{\text{best}}(x)$ 上次斑点鬣狗最优个体。

2.2 单纯形搜索法

在斑点鬣狗一次迭代完成之后,利用单纯形法搜

索策略,选择 K 个位置较差的斑点鬣狗进行优化。单纯形法是指在一个空间中构造一个多面体,求出多面体各个顶点的适应值并作比较,找出最优点、次优点以及最差点,通过反射、压缩、扩张等操作更新最差点,形成一个新的多面体。它是一种局域搜索方法,具有简单易用、适用范围广、收敛速度快的特点。单纯形搜索法搜索过程如图 1 所示。

求 n 维问题的最小值, N_1, N_2, \dots, N_{n+1} 为 n 维空间中的 $n+1$ 个顶点,描述单纯形法的步骤如下:

步骤 1: 计算所有点的适应值,找出最优点 x_b , 目标函数值记作 $f(x_b)$; 次优点 x , 目标函数值记作 $f(x)$; x_c 记作 x_b 和 x 的中心位置, 目标函数值记作 $f(x_c)$ 。中心位置: $x_c = (x_b + x) / 2$ 。

步骤 2: 找出其中的最差点, 记作 x_s , 目标函数值记作 $f(x_s)$, 对 x_s 进行反射操作得到反射点 x_r , 目标函数值记作 $f(x_r)$ 。反射操作: $x_r = x_c + \delta(x_c - x_s)$, δ 为反射系数, 通常 $\delta=1$ 。

步骤 3: 如果 $f(x_r) < f(x_b)$, 则反射方向正确, 执行扩张操作得到扩张点 x_e , 目标函数值记作 $f(x_e)$ 。扩张操作: $x_e = x_c - \phi(x_c - x_r)$, ϕ 为扩张系数, 通常 $\phi=2$; 如果 $f(x_e) < f(x_r)$, $x_s = x_e$; 否则为 $x_s = x_r$ 。

步骤 4: 如果 $f(x_b) \leq f(x_r) \leq f(x)$, 则为 $x_s = x_r$ 。

步骤 5: 如果 $f(x_r) > f(x)$, 则反射方向不正确,

$f(x_h) = \min\{f(x_r), f(x_s)\}$, 其中 $x_h \in \{x_r, x_s\}$, 执行压缩

操作得到压缩点 x_t , 目标函数值记作 $f(x_t)$ 。压缩操作:

$x_t = x_c - \psi(x_h - x_c)$, ψ 为压缩系数, 通常 $\psi \in (0, 1)$ 。

步骤 6: 如果 $f(x_t) \leq f(x_h)$, $x_s = x_t$, 否则对 x_s 和 x 执行收缩操作得到 x_w 和 x_{sw} 为 $x_w = x + \phi(x_b - x)$, $x_{sw} = x_s + \phi(x_b - x_s)$, $\phi=0.5$, 目标函数值记作 $f(x_{sw})$ 和 $f(x_w)$ 。

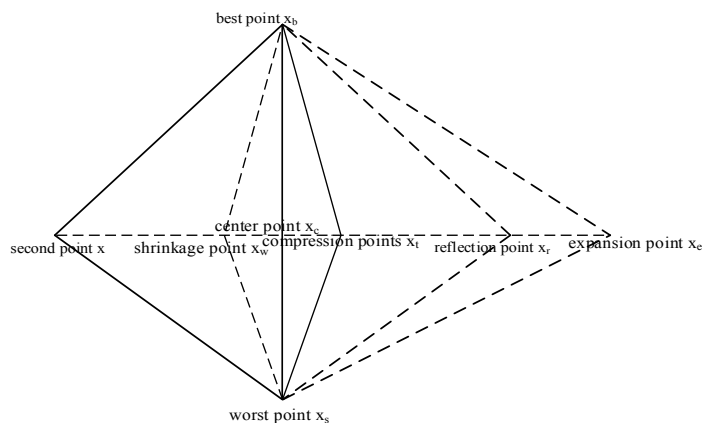


图1 单纯形搜索法

Fig.1 Simplex search method

2.3 Lévy_SM_SHO 算法步骤

相关参数设置及其含义,设种群规模为 N ,当前迭代次数为 i ,其初始值设置为 $i=0$,迭代总次数为 I ,计数器为 S ,初始值 $S=0$,阈值 $T=5$,算法中未设置的参数根据待解决实际问题进行设置;计数器变量 S 用于记录陷入局部最优的次数,阈值 T 表示触发改进策略的临界值,换言之当 $S>T$ 则触发改进策略,否则使用标准 SHO 算法的更新规则进行搜索。

步骤 1: 算法参数设置和随机生成种群规模为 N 的斑点鬣狗位置。

步骤 2: 使用标准 SHO 算法更新规则来搜索最优个体,当前迭代次数 i 自增 1 次;

步骤 3: 若当前迭代次数 $i\geq 2$,则执行步骤 4,否则执行步骤 2;

步骤 4: 判断此次迭代和前一次迭代相比适应度值是否更新,若适应度值更新了,说明没有陷入局部最优解,则继续使用标准 SHO 算法的更新规则进行搜索,执行步骤 2。若适应度值没有更新,则计数器 S 自增 1 次;

步骤 5: 判断计数器 S 的值是否达到触发改进策略的阈值 T ,若达到,则说明适应度值已经多次未更新,可能陷入了局部最优解无法跳出,执行步骤 6,若未达到,则执行步骤 2,继续使用标准 SHO 算法的更新规则进行搜索;

步骤 6: 将当前种群个体利用改进策略进行更新

得到改进后的种群;重新计算种群个体适应度值;将适应度值最小的个体作为新的最优个体替换掉原始种群中的最优个体;计数器归零 $S=0$;

步骤 7: 判断当前迭代次数 i 是否达到设置的总迭代次数 I ,若达到,则搜索过程结束,输出 I 次迭代中获取的最优解和最小适应度值。若迭代尚未完成,则执行步骤 2,继续进行搜索。

3 算法实验与分析

为了验证改进鬣狗优化算法 (Lévy_SM_SHO) 的有效性,使用文献[17]中的 F_5 、 F_6 、 F_{13} 、 F_{16} 、 F_{17} 和 F_{20} 的测试函数对其测试,其中测试函数 F_5 和 F_6 为单峰测试函数, F_{13} 为多峰测试函数, F_{16} 、 F_{17} 和 F_{20} 为固定维多峰函数,将其与 SM_SHO、Lévy_SHO 和 SHO 测试结果进行对比。每种算法单独运行 50 次,记录每次实验结果,对 50 次实验的适应度函数值进行分析。测试函数如表 1 所示。

3.1 算法实验

实验搭建的运行环境为 Win7, I5 3.20 GHz 处理器,8 G 内存,所有算法代码均用 Matlab 2016b 编程实现。

3.2 结果分析

结合图 2 中 4 种算法运行 50 次的适应度函数箱型图和表 2 结果可知,Lévy_SM_SHO 算法在 F_5 、 F_6 、

表 1 测试函数
Table 1 Test functions

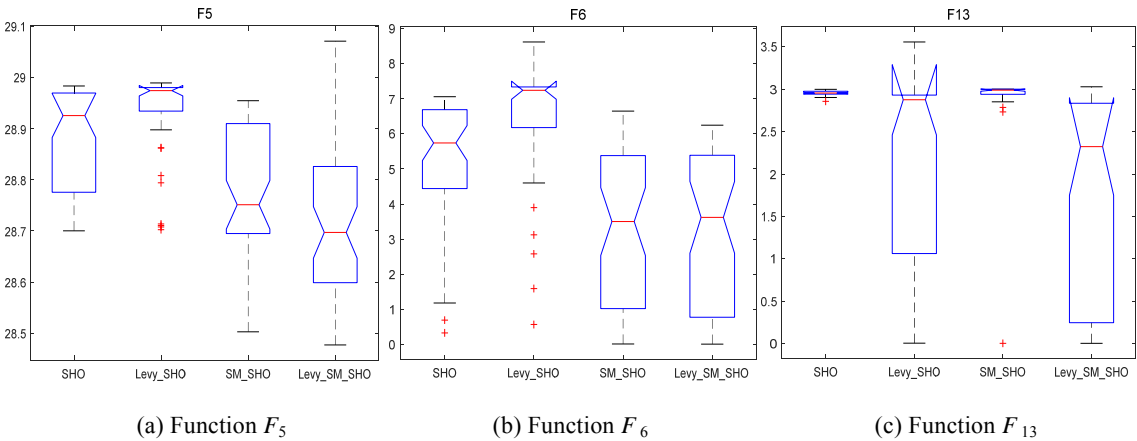
Function	Expression	Dimension	Search range	Minimum
F_5	$f_5(x)=\sum_{i=1}^n[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	30	$[-30,30]$	0
F_6	$f_6(x)=\sum_{i=1}^n(x_i+0.5)^2$	30	$[-100,100]$	0
F_{13}	$f_{13}(x)=0.1\{\sin^3(3\pi x_1)+\sum_{i=1}^n(x_i-1)^2[1+\sin^2(3\pi x_i)]$ $+ (x_n-1)^2[1+\sin^2(2\pi x_n)]\}+\sum_{i=1}^nu(x_i,5,100,4)$ $u(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, & x_i>a \\ 0, & -a\leq x_i\leq a \\ k(-x_i-a)^m, & x_i<-a \end{cases}$	30	$[-50,50]$	0
F_{16}	$f_{16}(x)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	2	$[-5,5]$	-1.0316
F_{17}	$f_{17}(x)=(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6)^2+10(1-\frac{1}{8\pi})\cos x_1+10$	2	$[-5,5]$	0.398
F_{20}	$f_{20}(x)=-\sum_{i=1}^4c_i\exp(-\sum_{j=1}^3a_{ij}(x_j-p_{ij})^2)$	6	$[0,1]$	-3.3

F_{13} 能取得较好的平均值和最小值,但是标准差较大,在固定维多峰函数 F_{16} 、 F_{17} 和 F_{20} 能取得明显比其他3种算法较好的适应度函数值,有较小的标准差值,说明 Lévy_SM_SHO 算法的稳定性较好。

Lévy_SM_SHO 算法的平均值比其他算法好,说明其有较好的求解精度。实验证明 Lévy_SM_SHO 算法取得较好的改进效果,能为解决复杂工程优化问题提供方法。

表 2 4 种算法在测试函数上的测试结果

Table 2 Test results of 4 algorithms on test functions					
Function	Indicators	SHO	Lévy_SHO	SM_SHO	Lévy_SM_SHO
F_5	Maximum	28.9838	29.6218	29	29.0713
	Minimum	28.7005	28.7027	28.5028	28.4771
	Average	28.8738	28.9371	28.7862	28.7253
	Standard	0.1025	0.0799	0.0947	0.1582
F_6	Maximum	7.06	7.02	7.50	6.19
	Minimum	0.326	0.0421	0.0182	0.003723
	Average	5.23	4.710	3.26	3.19
	Standard	1.93	2.29	2.92	2.25
F_{13}	Maximum	2.99	3.2709	3	3.0271
	Minimum	2.86	0.0046	2.7835	0.0023
	Average	2.95	1.8008	2.8493	1.6320
	Standard	0.0302	1.1731	0.0559	1.1147
F_{16}	Maximum	-0.11	-1.0101	0	-1.0094
	Minimum	-1.03	-1.0316	-1.0316	-1.0316
	Average	-0.94	-1.0252	-0.9882	-1.0304
	Standard	0.147	0.0057	0.1444	0.0031
F_{17}	Maximum	3.597	0.7162	0.51318	0.4690
	Minimum	0.398	0.3982	0.3979	0.3979
	Average	0.642	0.4282	0.4060	0.4031
	Standard	0.484	0.0528	0.0201	0.0108
F_{20}	Maximum	-1.6	-2.0591	-2.7087	-2.9141
	Minimum	-3.1	-3.0991	-3.3031	-3.3047
	Average	-2.6	-2.7685	-3.167	-3.1990
	Standard	0.3	0.2243	0.145831	0.09829



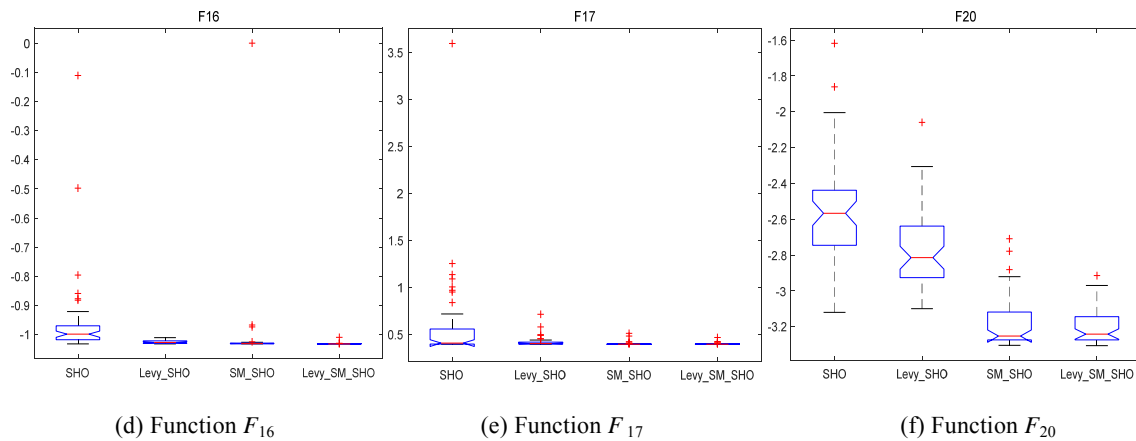


图2 四种算法对测试函数上的箱型图

Fig.2 Box diagrams of the four algorithms on the test function

4 红外图像分割

将 Lévy_SM_SHO 算法应用于电力系统红外图像分割中,选择断路器瓷套点状温度分布异常和断路器静触头发热两幅图像进行分割。为验证该对红外图像阈值分割效果,选择与粒子群算法 (particle swarm optimization, PSO) 分割结果对比。

4.1 适应度函数选择

选择最大类间方差 (Otsu) 公式作为 Lévy_SM_SHO 算法和 PSO 算法的适应度函数。

4.2 图像阈值分割评价

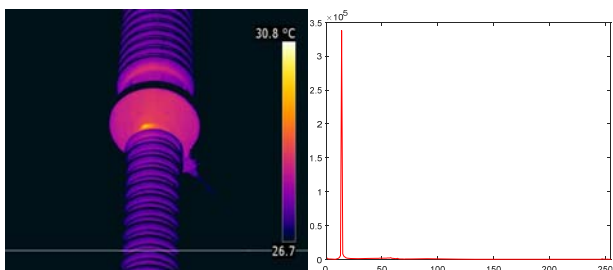
为了评计阈值分割后的图像质量,本文采用峰值信噪比 (peak signal to noise ratio, PSNR) 和结构相似度 (structural similarity index measurement, SSIM) 两个参数。PSNR 和 SSIM 值越大,阈值分割效果越好。

4.2.1 PSNR

峰值信噪比是图像分割的重要性能指标之一。

PSNR 的表达式定义如下:

$$\text{PSNR} = 20 \times \lg\left(\frac{255}{\text{MSE}(x, y)}\right) \quad (15)$$



(a) 断路器瓷套点状温度分布异常原图和灰度图

(a) The original and grayscale drawings of the abnormal point-like temperature distribution of the porcelain sleeve of the circuit breaker

$$\text{MSE} = \sqrt{\frac{\sum_{i=1}^X \sum_{j=1}^Y (I(i, j) - I'(i, j))^2}{X \times Y}} \quad (16)$$

式中: $X \times Y$ 表示图像的大小; I 表示原图; I' 表示阈值分割后的图。

4.2.2 SSIM

SSIM 指数是评价原始图像与分割图像之间的相似性, 公式为:

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

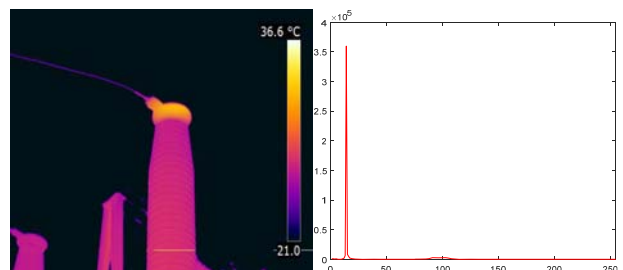
式中: μ_I 、 $\mu_{I'}$ 分别代表图像 I 和 I' 的平均值; σ_I^2 、 $\sigma_{I'}^2$

分别代表图像 I 和 I' 的方差; $c_1 = (k_1 D)^2$ 、 $c_2 = (k_2 D)^2$, $D = 255$, $k_1 = 0.01$, $k_2 = 0.03$ 。

$$\text{MSSIM} = \sum_{i=1}^X \sum_{j=1}^Y \text{SSIM}(I, I') \quad (18)$$

4.3 分割结果分析

图 3 为 2 幅图像的原图像及其灰度直方图, 图 4 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法的断路器瓷套二阈值分割结果, 图 5 基于 PSO-Otsu 和 Lévy_



(b) 断路器静触头发热原图和灰度图

(b) The original and grayscale drawings of the circuit breaker static contact hair hot

图3 测试图像原图和灰度直方图

Fig.3 Original test images and gray histograms

SM_SHO-Otsu 算法的断路器静触头二阈值分割结果。表 3 为 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法对红外图像分割阈值,表 4 为两种算法适应度函数值,表 5 为两种算法 PSNR 和 SSIM 值。

从表 4 与表 5 以及图 4 与图 5 分割的结果分析

得知:①Lévy_SM_SHO-Otsu 比 PSO-Otsu 取得更好的适应度函数值;②以 PSNR 和 SSIM 评价图像阈值分割的好坏时,Lévy_SM_SHO-Otsu 算法比 PSO-Otsu 算法取得更大的 PSNR 和 SSIM 值,说明图像失真度较小,分割效果较优。

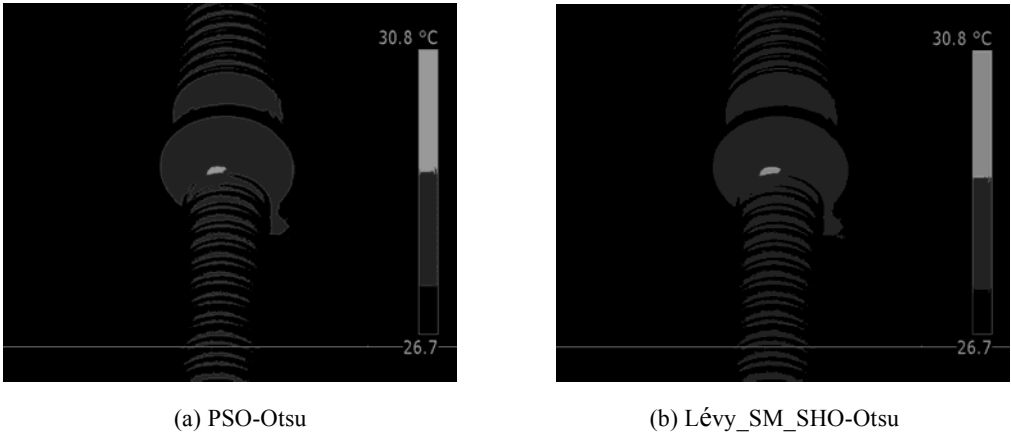


图 4 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法的断路器瓷套二阈值分割结果

Fig.4 Two threshold segmentation results of ceramic sleeve of circuit breaker based on PSO-Otsu and Lévy_SM_SHO-Otsu algorithms

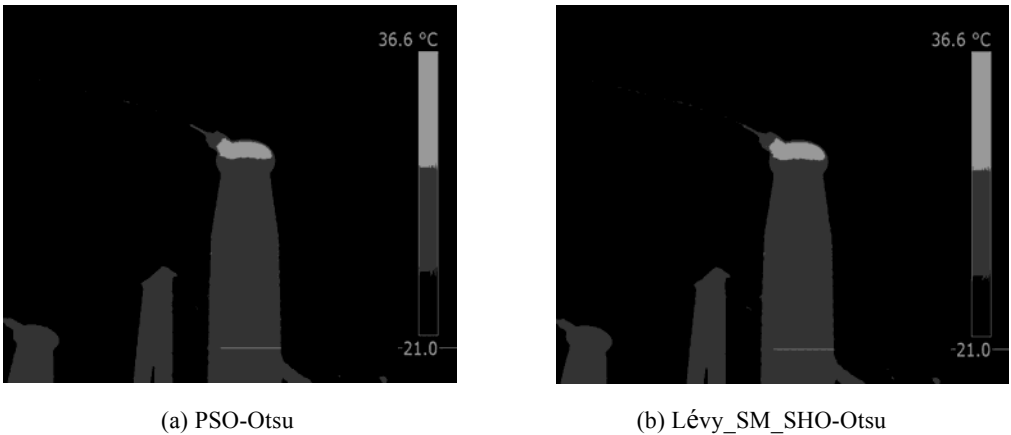


图 5 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法的断路器静触头二阈值分割结果

Fig.5 Two threshold segmentation results of circuit breaker static contacts based on PSO-Otsu and Lévy_SM_SHO-Otsu algorithms

表 3 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 的最佳阈值

Table 3 The optimal thresholds based on PSO-Otsu and Lévy_SM_SHO-Otsu algorithms

Infrared image	The number of threshold	PSO-Otsu	Lévy_SM_SHO-Otsu
		threshold	
Abnormal temperature distribution in porcelain sleeve of circuit breaker	1	70	70
	2	44,145	46,143
	3	73,74,255	70,76,255
	4	46,138,173,255	46,73,144,255
	5	69,88,181,255,255	46,54,132,139,255
	6	58,142,167,181,255,255	53,65,76,144,255,255
The circuit breaker still touches the hair to heat	1	63	63
	2	51,147	57,146
	3	44,72,255	63,71,255
	4	38,80,123,255	37,63,121,255
	5	58,70,135,150,255	55,62,72,255,255
	6	54,71,97,186,246,255	58,72,82,138,245,255

表4 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法的适应度函数

Table 4 Fitness function based on PSO-Otsu and Lévy_SM_SHO-Otsu algorithms			
Infrared image	The number of threshold	PSO-Otsu	Lévy_SM_SHO-Otsu
		The value of fitness functions	
Load switch	1	851.7254	851.7254
	2	1096.1	1096.2
	3	1631.5	1633.5
	4	1746.8	1877
	5	2096.3	2197.8
	6	2255.8	2614.6
Load switch	1	1282.8	1282.8
	2	1486.6	1487
	3	2305.5	2362.6
	4	2521.7	2565.1
	5	2763.8	3437.9
	6	3182.6	3635.9

表5 基于 PSO-Otsu 和 Lévy_SM_SHO-Otsu 算法的 PSNR 和 SSIM 值

Table 5 PSNR and SSIM values based on PSO-Otsu and Lévy_SM_SHO-Otsu algorithms					
Infrared image	The number of threshold	PSO-Otsu		Lévy_SM_SHO-Otsu	
		PSNR	SSIM	PSNR	SSIM
Load switch	1	18.6778	0.0981	18.6778	0.0981
	2	21.0977	0.1753	21.1041	0.1794
	3	18.7711	0.0969	18.9324	0.1011
	4	21.7095	0.1741	22.3763	0.1806
	5	21.3552	0.1266	21.5929	0.1760
	6	21.3714	0.1370	21.9312	0.1580
Load switch	1	19.1335	0.1567	19.1335	0.1567
	2	20.7128	0.1686	21.2433	0.1720
	3	20.0060	0.1738	20.1006	0.1741
	4	22.5955	0.1882	22.6117	0.2747
	5	22.6896	0.1892	22.7556	0.1908
	6	24.0231	0.1900	24.1233	0.1964

5 结语

鬣狗优化算法具有算法结构简单、收敛速度快等优点，同时也存全局搜索能力差、易早熟等缺点。通过使用 Lévy 飞行和单纯形搜索算法来改进鬣狗算法，在测试函数上验证了改进后能取得较优的效果。并通过对电力系统红外图像分割测试，验证改进后算法能够取得更好的分割效果。进一步表明 Lévy_SM_SHO-Otsu 具有较强的实用性、高效性和可行性。

参考文献:

[1] Maryam M, Majid H, Fatemeh N. Air condition's PID controller

fine-tuning using artificial neural networks and genetic algorithms[J]. *Computers*, 2018, 7(2): 32.

[2] Ayala H V H, Coelho L D S. Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator[J]. *Expert Systems with Applications*, 2012, 39(10): 8968-8974.

[3] Beyer H G, Schwefel H P. Evolution strategies——a comprehensive introduction[J]. *Natural Computing*, 2002, 1: 3-52.

[4] ZHU W, DUAN H. Chaotic predator-prey biogeography-based optimization approach for UCAV path planning[J]. *Aerospace Science & Technology*, 2014, 32(1): 153-161.

[5] Ghosh A, Das S, Chowdhury A, et.al. An improved differential evolution algorithm with fitness-based adaptation of the control parameters[J]. *Information Sciences*, 2011, 181(18): 3749-3765.

[6] 孙云霞, 刘兆刚, 董灵波. 基于模拟退火算法逆转搜索的森林空间

- 经营规划[J]. 林业科学, 2019, **55**(11): 52-62.
- SUN Yunxia, LIU Zhaogang, DONG Lingbo. Spatial forest management planning based on reversion search technique of simulated annealing algorithm[J]. *Scientia Silvae Sinicae*, 2019, **55**(11): 52-62
- [7] Rashedi E, Nezamabadi-Pour H, Saryazdi S. BGSA: Binary gravitational search algorithm[J]. *Natural Computing*, 2010, **9**(3): 727-745.
- [8] PAN Q K, WANG L, GAO L. A Chaotic Harmony Search Algorithm for the Flow Shop Scheduling Problem with Limited Buffers[M]. Elsevier Science Publishers B. V., 2011.
- [9] Shaikh N F, Doye D D. An adaptive central force optimization (ACFO) and feed forward back propagation neural network (FFBNN) based iris recognition system[J]. *Journal of Intelligent and Fuzzy Systems*, 2016, **30**(4): 2083-2094.
- [10] 金旭旸. 基于莱维飞行的水波优化算法[J]. 科技创新与生产力, 2019(5): 66-68.
- JIN Xuyang. Water wave optimization algorithm based on Lévy flight[J]. *Sci-tech Innovation and Productivity*, 2019(5): 66-68.
- [11] HUANG H, YANG X, HAO Z, et al. A novel ACO algorithm with adaptive parameter[C]//*International Conference on Intelligent Computing on Lecture Notes in Computer Science*, 2006, **4115**: 12-21.
- [12] NIU B, ZHU Y, HU K, et al. A novel particle swarm optimizer using optimal foraging theory[C]//*International Conference on Intelligent Computing on Computational Intelligence and Bioinformatics*, 2006, **4115**: 61-71.
- [13] Akay B, Karaboga D. A modified Artificial Bee Colony algorithm for real-parameter optimization[J]. *Information Sciences*, 2012, **192**(1): DOI:10.1016/j.ins.2010.07.015.
- [14] QIANG Z, LI H, LIU C, et al. A new extreme learning machine optimized by firefly algorithm[C]//*Proceedings of the 2013 Sixth International Symposium on Computational Intelligence and Design of IEEE*, 2013: <https://ieeexplore.ieee.org/search/searchresult.jsp?newsearch=true&queryText=A%20new%20extreme%20learning%20machine%20optimized%20by%20firefly%20algorithm>.
- [15] Mukherjee A, Mukherjee V. A solution to optimal power flow with DC link placement problem using chaotic krill herd algorithm[C]//*International Conference on Emerging Technological Trends of IEEE*, 2016: DOI: 10.1109/ICETT.2016.7873756
- [16] JIANG T, ZHANG C. Application of grey wolf optimization for solving combinatorial problems: job shop and flexible job shop scheduling cases[J]. *IEEE Access*, 2018, **6**: 26231- 26240.
- [17] Dhiman G, Kumar V. Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications[J]. *Advances in Engineering Software*, 2017, **114**: 48-70.
- [18] Dhiman G, Kaur A. Spotted hyena optimizer for solving engineering design problems[C]// *2017 International Conference on Machine Learning and Data Science (MLDS) of IEEE*, 2017: 114-119.
- [19] Dhiman G, Kumar V. Spotted Hyena Optimizer for Solving Complex and Non-linear Constrained Engineering Problems[M]//*Harmony Search and Nature Inspired Optimization Algorithms, Advances in Intelligent Systems and Computing*, Singapore: Springer Singapore, 2019, **741**: 857-867.
- [20] 贾鹤鸣, 姜子超, 彭晓旭, 等. 基于改进鬣狗优化算法的多阈值彩色图像分割[J]. 计算机应用与软件, 2020, **37**(5): 261-267.
- JIA Heming, JIANG Zichao, PENG Xiaoxu, et al. Multi-threshold color image segmentation based on improved hyena optimization algorithm[J]. *Computer Applications and Software*, 2020, **37**(5): 261-267.
- [21] 孙悦, 何同祥. 基于莱维飞行的改进蚁群算法的 PID 参数优化[J]. 仪器仪表用户, 2019, **26**(6): 83-85.
- SUN Yue, HE Tongxiang. Optimization of PID parameters based on improved ant colony algorithm for Lévy flight[J]. *Electronic Instrumentation Customers*, 2019, **26**(6): 83-85.
- [22] 张烈平, 何佳洁, 于滢琳, 等. 基于蚁群算法优化的布谷鸟搜索算法[J]. 微电子学与计算机, 2018, **35**(12): 27-32.
- ZHANG Lieping, HE Jiajie, YU Yanlin, et al. A cuckoo search algorithm based on ant colony algorithm optimization[J]. *Microelectronics & Computer*, 2018, **35**(12): 27-32.
- [23] 张新, 李珂, 严大虎, 等. 改进入侵杂草算法求解柔性作业车间调度问题[J]. 系统仿真学报, 2018, **30**(11): 446-453.
- ZHANG Xin, LI Ke, YAN Dahu et al. Improved intrusion weed algorithm for solving flexible job shop scheduling problem[J]. *Journal of System Simulation*, 2018, **30**(11): 446-453.
- [24] 赵洪, 李伟鹏, 刘铁军. 基于改进莱维飞行的狼群算法及其在翼型气动优化设计中的应用[J]. 科学技术与工程, 2019, **19**(18): 315-323.
- ZHAO Hong, LI Weipeng, LIU Tiejun. An improved Lévy flight based grey wolf optimization algorithm for aerodynamic design problem[J]. *Science Technology and Engineering*, 2019, **19**(18): 315-323.
- [25] 肖石林. 基于 Lévy 飞行的树种优化算法及在图像分割中的应用[D]. 南宁: 广西民族大学, 2019.
- XIAO Shilin. Tree Seed Optimization Algorithm Based on LévyFlight and Its Application in Image Segmentation[D]. Nanning: Guangxi University for Nationalities, 2019.
- [26] 莫愿斌, 郑巧燕, 马彦追. 单纯形法的布谷鸟搜索算法及其在约束优化问题中的应用[J]. 计算机与应用化学, 2015(2): 213-218.
- MO Yuanbin, ZHENG Qiaoyan, MA Yanzhui. Cuckoo search based on simplex method and its application on constrained optimization problems[J]. *Computers and Applied Chemistry*, 2015(2): 213-218.
- [27] 莫愿斌, 马彦追, 郑巧燕, 等. 单纯形法的改进萤火虫算法及其在非线方程组求解中的应用[J]. 智能系统学报, 2014(6): 747-755.
- MO Yuanbin, MA Yanzhui, ZHENG Qiaoyan, et al. Improved firefly algorithm based on simplex method and its application in solving non-linear equation groups[J]. *CAAI Transactions on Intelligent Systems*, 2014(6): 747-755.
- [28] 张红霞, 罗毅, 师瑞峰. 基于单纯形法的改进型人工鱼群算法[J]. 计算机应用, 2011(5): 1321-1323.
- ZHANG Hongxia, LUO Yi, SHI Ruifeng. Artificial fish swarm algorithm based on simplex method[J]. *Journal of Computer Applications*, 2011(5): 1321-1323.
- [29] 肖辉辉. 基于单纯形法和自适应步长的花朵授粉算法[J]. 计算机工程与科学, 2016, **38**(10): 2126-2133.
- XIAO Huihui. A flower pollination algorithm based on simplex method and self-adaptive step[J]. *Computer Engineering & Science*, 2016, **38**(10): 2126-2133.