

# 基于 Lévy 飞行的改进飞蛾扑火算法优化红外图像分割

李唐兵<sup>1</sup>, 胡锦涛<sup>2</sup>, 周求宽<sup>1</sup>

(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:** 针对使用传统阈值分割方法对电力设备故障诊断效率低、精度低的问题, 使用智能算法优化 Otsu 算法对红外图像进行阈值分割再进行故障诊断。根据基本飞蛾扑火 (Moth-Flame Optimization, MFO) 算法缺点提出改进飞蛾扑火算法 (Improved Moth-Flame Optimization Algorithm, IMFO) 并将其应用红外图像分割中, 通过对比粒子群算法 (Particle Swarm Optimization, PSO)、生物地理算法 (Biogeography-Based Optimization, BBO)、基本飞蛾扑火算法红外图像分割效果, 表明改进算法取得成功。提出一种通过温度区域对红外图像进行多阈值分割的方法, 能够准确确定每个部分的温度范围, 从而保证设备的正常运行。

**关键词:** 红外图像; IMFO; 故障诊断; 多阈值

中图分类号: TN219

文献标识码: A

文章编号: 1001-8891(2020)09-0846-09

## Improved Moth-Flame Optimization Algorithm Based on Lévy Flight to Optimize Infrared Image Segmentation

LI Tangbing<sup>1</sup>, HU Jinhong<sup>2</sup>, ZHOU Qiukuan<sup>1</sup>

(1. Power science research institute of state grid Jiangxi electric power company, Jiangxi 330096, China;

2. State grid Shanghai Pudong power supply company, Shanghai 200122, China)

**Abstract:** To solve the problem of low efficiency and accuracy of power equipment fault diagnosis using the traditional threshold segmentation method, an intelligent algorithm, the optimized Otsu algorithm was used for threshold segmentation of infrared images for fault diagnosis. According to the shortcomings of the basic moth-flame optimization, the improved moth-flame optimization algorithm is proposed. It was applied to the infrared image segmentation. By comparing its infrared image segmentation results with those of the particle swarm optimization, biogeography-based optimization, and moth-flame optimization algorithms, it was shown that the improved algorithm is successful. A multithreshold segmentation method for infrared images through the temperature region is proposed. It can accurately determine the temperature range of each part and ensure normal operation of the equipment.

**Key words:** infrared image, IMFO, fault diagnosis, multilevel thresholding

## 0 引言

图像分割是将图像分割成若干个段或区域, 并提取有意义的和感兴趣的对象, 这是图像处理和图像分析的关键步骤。目前将基本图像处理算法 Otsu、KSW 以及基本算法和群智能算法相结合的算法用于图像分割计算的文献已达上千种, 阈值分割对图像分割性能较好, 此法被广泛应用。图像分割在医学成像、物体检测、电力系统故障诊断和遥感等领域中广泛应用。求一幅图的最佳阈值, 可以看作求一个约束优化

问题, 由于求取阈值计算量大, 所以结合智能算法求取阈值更加准确。用智能算法求取图像的阈值有差分进化算法<sup>[1-2]</sup> (Differential Evolution, DE)、细菌觅食算法<sup>[3-4]</sup> (Bacterial Foraging Algorithm, BFA)、萤火虫算法<sup>[5-8]</sup> (Firefly Algorithm, FA)、人工蜂群算法<sup>[9-11]</sup> (Artificial Bee Colony Algorithm, ABC)、布谷鸟算法<sup>[12-15]</sup> (Cuckoo Search, CS) 等。电力系统检测主要使用红外图像于中, 研究红外图像主要为了实现变电站在线监测技术。红外图像具有噪声和分辨率低的特点, 传统算法分割效果差, 为了改进分割效

收稿日期: 2019-06-08; 修订日期: 2020-06-06.

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)。

果, 引用智能算法优化传统阈值分割算法进行分割, 能够更好地分割出每个部分, 为电力系统在线诊断提供可靠依据。

## 1 最大类间方差法 (Otsu)

最大类间方差法<sup>[16]</sup>是由日本学者 Otsu 于 1979 年提出的, 是一种高效的图像分割算法。Otsu 阈值分割法是一种非参数形式的分割方法, 通过求取中间的方差最大值, 将图像进行分类。

### 1.1 单阈值图像分割

假设一幅图像大小为  $H \times W$ , 则像素点数为  $N(H \times W)$ , 灰度级为  $L(L=256)$ , 灰度级的范围为  $0, 1, \dots, L-1$ , 灰度值为  $i$  的像素点个数为  $N_i$ ,  $N=N_0+N_1+\dots+N_{L-1}$ , 灰度值  $i$  出现的概率为:

$$p_i = \frac{N_i}{N}, \quad p_i > 0 \text{ 且 } \sum_{i=0}^{L-1} p_i = 1 \quad (1)$$

对于单阈值分割, 最优阈值  $t$  将图像分割成 C0 和 C1 两类, 这两类像素的概率和分别为:

$$w_0 = \sum_{i=0}^t p_i, \quad w_1 = \sum_{i=t+1}^{L-1} p_i = 1 - w_0 \quad (2)$$

图像的总均值为:

$$\mu_T = \sum_{i=1}^{L-1} ip_i \quad (3)$$

C0 和 C1 两类的均值分别为:

$$\mu_0 = \sum_{i=1}^t ip_i / w_0 = \mu(t) / w_0, \quad \mu(t) = \sum_{i=1}^t ip_i \quad (4)$$

$$\mu_1 = \sum_{i=t+1}^{L-1} ip_i / w_1 = (\mu_T - \mu(t)) / w_1 \quad (5)$$

由式(3)、(4)和(5)得:

$$\mu_1 = w_0 \mu_0 + w_1 \mu_1, \quad w_0 + w_1 = 1 \quad (6)$$

图像的两个类的类间方差为:

$$\begin{aligned} f(t) &= \sigma_0 + \sigma_1 \\ &= w_0 (\mu_0 - \mu_T)^2 + w_1 (\mu_1 - \mu_T)^2 \\ &= w_0 w_1 (\mu_0 - \mu_1)^2 \end{aligned} \quad (7)$$

最优阈值  $t$  是通过类间方差公式在整个图像中搜索得到, 最优阈值  $t$  为:

$$t^* = \arg \max_{0 \leq t \leq L-1} (f(t)) \quad (8)$$

### 1.2 多阈值图像分割

将单阈值方法拓展到多阈值上, 假设一幅图像被分成了  $M$  层,  $M$  层之间的类间方差为:

$$f(t) = \sum_{i=0}^{M-1} \sigma_i \quad (9)$$

图像的  $M$  层每层的均值为:

$$\begin{aligned} \mu_0 &= \sum_{i=0}^{t_1} ip_i / w_0, \\ \mu_1 &= \sum_{i=t_1+1}^{t_2} ip_i / w_1, \\ &\vdots \\ \mu_{M-1} &= \sum_{i=t_{M-1}+1}^{L-1} ip_i / w_{M-1} \end{aligned} \quad (10)$$

图像的  $M$  层每层的  $\sigma$  值为:

$$\begin{aligned} \sigma_0 &= w_0 (\mu_0 - \mu_T)^2 \\ \sigma_1 &= w_1 (\mu_1 - \mu_T)^2 \\ &\vdots \\ \sigma_{M-1} &= w_{M-1} (\mu_{M-1} - \mu_T)^2 \end{aligned} \quad (11)$$

图形的  $M$  类的类间方差公式为:

$$t^*(t_1, t_2, \dots, t_{M-1}) = \arg \max_{0 \leq t \leq L-1} (\sum_{i=0}^{M-1} \sigma_i) \quad (12)$$

## 2 飞蛾扑火算法

在 2015 年, Mirjalili 用数学方法模仿飞蛾的飞行行为, 提出的一种新的启发式智能算法, 称为飞蛾扑火优化算法<sup>[17]</sup>。该算法的主要灵感来自飞蛾的横向导航飞行机制, 飞蛾在夜间飞行时, 对月亮保持一个固定的角度, 这种行为使他们在长距离飞行时始终保持直线飞行。在日常生活中, 飞蛾把人造光误以为月光, 于是就有飞蛾绕着人造光作螺旋曲线运动。

在 MFO 算法中, 假设飞蛾为优化问题的候选解, 问题的变量是飞蛾在空间中的位置。通过改变它们的位置矢量飞蛾可以在任意空间中飞行。  $n$  个飞蛾在  $d$  维空间的位置为:

$$M = (m_{i1}, m_{i2}, \dots, m_{id}), \quad i = 1, 2, \dots, n$$

数组 **OM** 用于存储相应的适应度值:

$$OM = [OM_1, OM_2, \dots, OM_n]^T$$

该算法的另一个关键组成部分是火焰, 火焰矩阵:

$$F = (F_{i1}, F_{i2}, \dots, F_{id}), \quad i = 1, 2, \dots, n$$

数组 **OF** 用于存储相应的适应度值:

$$OF = [OF_1, OF_2, \dots, OF_n]^T$$

在 MFO 算法中, 飞蛾和火焰都被认为是解。它们之间的区别是我们在每次迭代中处理和更新它们的方式。飞蛾实际上是在搜索空间中移动的搜索主体, 飞蛾获得的最佳位置用火焰表示。火焰可以看作是飞蛾在搜索空间时落下的旗子, 每个飞蛾在旗子周围搜索, 并在找到更好的解决方案时更新其位置。有

了这种方法，飞蛾就不会失去它最好的解决方案。

MFO 算法是一个近似于优化问题的全局最优的三元组  $MFO=(I, P, T)$ ,  $I$  生成初始化解和计算相应的适应度值的函数，为  $I:\phi \rightarrow \{M, OM\}$ ，初始化后，函数  $P:M \rightarrow M$  成为主函数，循环运行，直到函数  $T:M \rightarrow \{true, false\}$  返回 true。在 MFO 中，对每只飞蛾的位置  $M_i$  进行火焰更新，选择对数螺旋  $S(M_i, F_j)$  作为飞蛾的主要更新机制，定义如下：

$$M_i = S(M_i, F_j) \quad (13)$$

$$S(M_i, F_j) = D_i \cdot \exp(bt) \cdot \cos(2\pi t) + F_j \quad (14)$$

$$D_i = |F_j - M_i| \quad (15)$$

式中： $M_i$  表示第  $i$  个飞蛾； $F_j$  表示第  $j$  个火焰； $S$  表示螺旋函数； $D_i$  表示第  $i$  个飞蛾到第  $j$  个火焰的距离； $b$  是一个定义对数螺旋线形状的常数； $t$  是  $[-1,1]$  之间的随机数。

为了防止 MFO 陷入局部最优状态，每个蛾只需要使用其中一个火焰来更新自己的位置，更新机制如式(14)。火焰位置被更新，并根据它们的适应度值进行排序，然后蛾子根据相应的火焰更新它们的位置。在搜索空间中，蛾子相对于  $n$  个不同位置可能会降低对最优解的寻找。为提高开发能力，提出了火焰数量的自适应机制，公式为：

$$\text{flame no} = \text{round} \left( N - l * \left( \frac{N-1}{T} \right) \right) \quad (16)$$

式中： $l$  为当前迭代次数； $N$  为最大火焰数； $T$  为最大迭代次数。

MFO 算法目前处于研究阶段，该算法的收敛速度和计算精度可以进一步提高。

### 3 改进飞蛾扑火算法

受文献[5]中智能算法的启发，文献中 Lévy 飞行和萤火虫算法 (FA) 相结合能够提高算法的收敛速度和计算精度，本文采用对飞蛾扑火算法采用自适应权重和 Lévy 飞行两种方法改进飞蛾扑火算法。

#### 3.1 自适应权重

由于 MFO 算法使用对数螺旋函数更新飞蛾的位置，该函数简单定义了飞蛾扑火的位置，使得飞蛾容易陷入局部最优，在全局优化中存在一定的不足。本文采用自适应权重法。当飞蛾接近火焰寻找最优解时，自适应权重减小，从而提高了飞蛾的局部最优能力。权重公式为：

$$w = 0.6 - 0.4 \frac{l}{T} \quad (17)$$

应用自适应权重的飞蛾更新公式为：

$$S(M_i, F_j) = D_i \cdot \exp(bt) \cdot \cos(2\pi t) + w \cdot F_j \quad (18)$$

#### 3.2 Lévy 飞行

Lévy 飞行在开发未知的高维的搜索空间比布朗运动更加有效，Lévy 飞行能够增加种群的多样性，使得算法有效地跳出了局部最优。换言之，这种方法有利于 MFO 算法在搜索和开发能力之间取得更好的平衡。因此，在飞蛾位置更新后，使用下式对飞蛾群执行一次 Lévy 飞行，公式为：

$$M_i^{t+1} = M_i^t + \alpha \oplus \text{Levy}(\beta) \quad (19)$$

式中： $\alpha$  表示一个常数，与具体问题有关； $M_i^t$  为第  $i$  只个体第  $t$  次的位置； $\oplus$  表示点乘； $\text{Levy}(\beta)$  为随机搜索。

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：

$$\text{Levy}(\beta) \sim u = t^{-\beta}, \quad (1 < \beta < 3) \quad (20)$$

式(21)计算 Lévy 随机数：

$$\text{Levy}(\beta) = \frac{\phi \times u}{|v|^{1/\beta}} \quad (21)$$

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

$$\phi = \left[ \frac{\Gamma(1+\beta) \times \sin(\pi \times \beta / 2)}{\Gamma[(1+\beta)/2] \times \beta \times 2^{(\beta-1)/2}} \right] \quad (22)$$

### 4 仿真测试

#### 4.1 实验平台和参数设置

本节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 Matlab2016b 编程实现。最大类间方差分割法 (Otsu) 是通过对图像进行灰度化，计算不同的灰度值的类间方差值，找到最大类间方差值进而找到最佳分割阈值。文中把最大类间方差公式作为算法的适应度函数，使用智能算法寻找到最优函数值即得到图像分割的最佳阈值。

为了验证本文提出的改进飞蛾扑火算法 (Improve Moth-Flame Optimization Algorithm, IMFO) 对红外图像阈值分割的能力，本文算法与粒子群算法 (Particle Swarm Optimization, PSO)、生物地理算法 (Biogeography-Based Optimization, BBO)、飞蛾扑火算法 (MFO) 进行对比分析，分别对电力设备中负荷开关、架空线路夹、35 kV 隔离开关刀口和 220 kV 电流互感器头部 4 幅图像进行阈值为 2, 3, 4, 5,

6 的阈值分割实验。实验中算法参数设置为: 种群规模为 20 个个体, 最大迭代次数为 100 次。

#### 4.2 实验结果

图 1 为 4 幅图像的原图像及其灰度直方图, 图 2 是 4 幅图像基于 Otsu 的粒子群算法 (PSO) 的 2 阈值分割实验结果, 图 3 是 4 幅图像基于 Otsu 的生物地理算法 (BBO) 的 2 阈值分割实验结果, 图 4 是 4 幅图像基于 Otsu 的飞蛾扑火算法 (MFO) 的 2 阈值分割实验结果, 图 5 是 4 幅图像基于 Otsu 的改进飞蛾扑火算法 (IMFO) 的 2 阈值分割实验结果。

通过对比图 2~图 5, PSO、BBO、MFO 和 IMFO 四种算法对 4 幅红外图像的 2 阈值分割结果, 可以发

现每种算法的分割效果不同, 并不能比较出算法分割效果的优劣。通过图 6 四幅图形的算法收敛曲线, 能够得到改进的飞蛾扑火算法能有更好的收敛速度。

对比表 1 是 4 幅图像的 4 种算法运行时间可以发现 PSO 比 BBO、MFO 和 IMFO 算法运行时间很短, 但是算法复杂度很低, 从而能够在很短时间内收敛。表 2 中为 4 种算法对 4 幅图像进行多阈值分割结果, 通过比较数值并不能得出算法的优劣。通过对比表 3 中 PSO、BBO、MFO 和 IMFO 算法的适应度函数值和表 4 图像分割评价指标 PSNR 和 SSIM 值的结果, 能够证明表 2 中 IMFO 算法取得图像阈值更加准确。

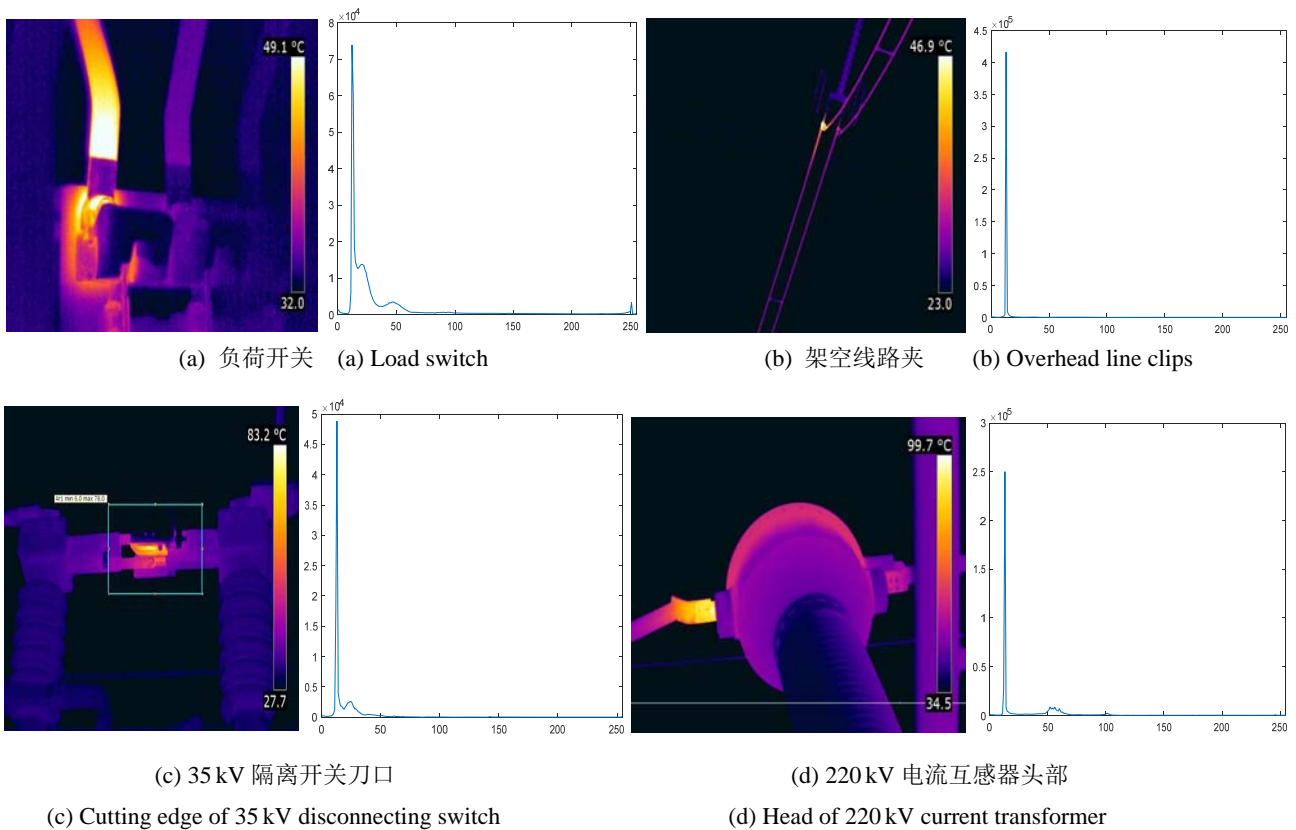
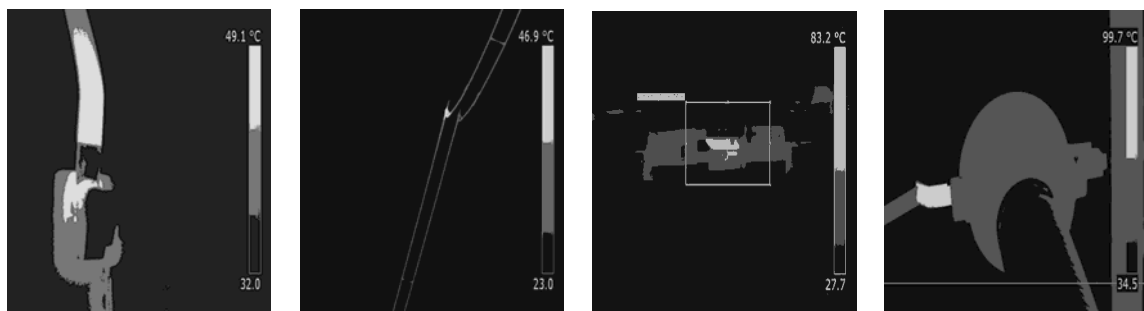


图 1 原图像及其灰度直方图

Fig.1 Original image and its gray histogram

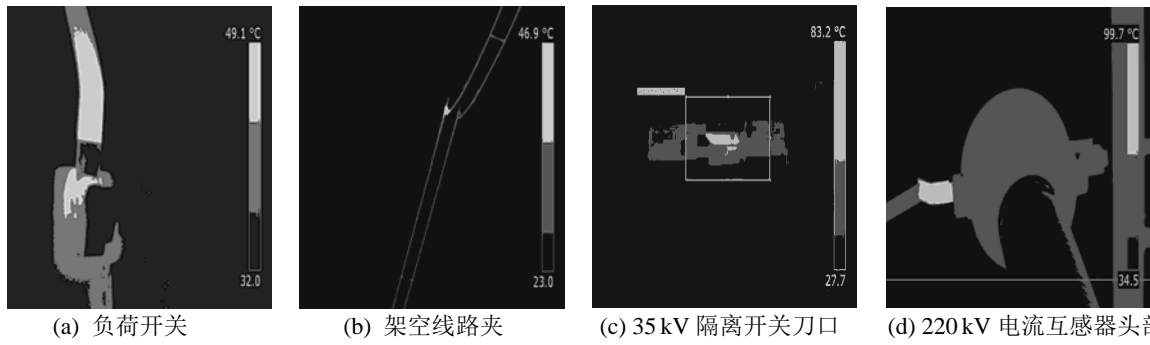


(a) 负荷开关 (b) 架空线路夹 (c) 35 kV 隔离开关刀口 (d) 220 kV 电流互感器头部

(a) Load switch (b) Overhead line clips (c) Cutting edge of 35 kV disconnecting (d) Head of 220kV current transformer switch

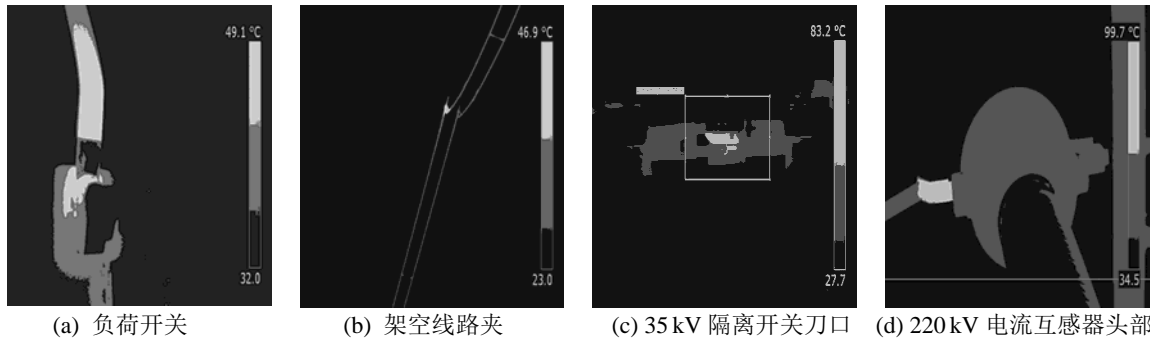
图 2 基于 Otsu 的粒子群算法 (PSO) 实验结果

Fig.2 Experimental results of Particle Swarm Optimization (PSO) based on Otsu



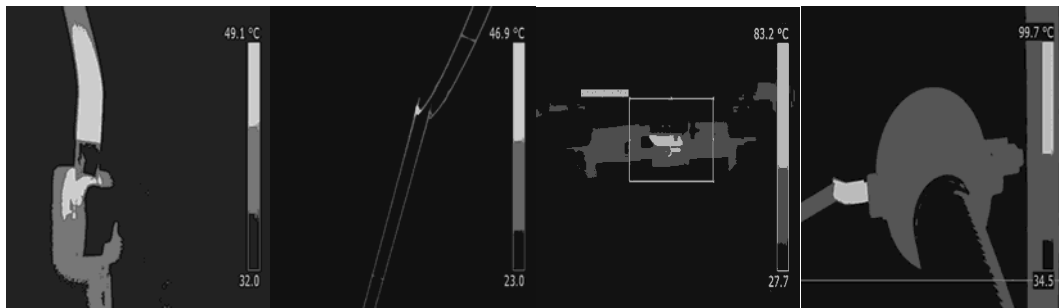
(a) 负荷开关 (b) 架空线路夹 (c) 35 kV 隔离开关刀口 (d) 220 kV 电流互感器头部  
(a) Load switch (b) Overhead line clips (c) Cutting edge of 35 kV disconnecting (d) Head of 220 kV current transformer switch  
图 3 基于 Otsu 的生物地理算法 (BBO) 实验结果

Fig.3 Experimental results of Biogeography-based Optimization (BBO) based on Otsu



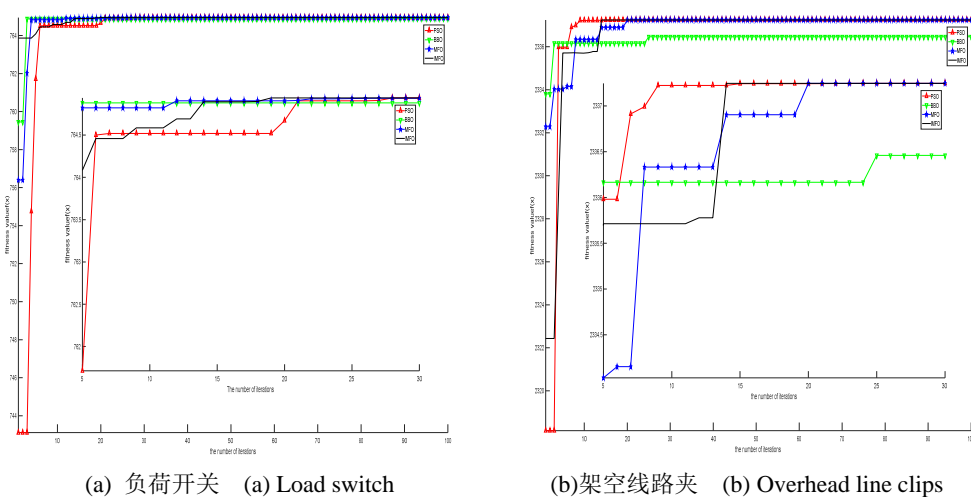
(a) 负荷开关 (b) 架空线路夹 (c) 35 kV 隔离开关刀口 (d) 220 kV 电流互感器头部  
(a) Load switch (b) Overhead line clips (c) Cutting edge of 35 kV disconnecting (d) Head of 220 kV current transformer switch  
图 4 基于 Otsu 的飞蛾扑火算法 (MFO) 实验结果

Fig.4 Experimental results of Moth-Flame Optimization Algorithm (MFO) based on Otsu



(a) 负荷开关 (b) 架空线路夹 (c) 35 kV 隔离开关刀口 (d) 220 kV 电流互感器头部  
(a) Load switch (b) Overhead line clips (c) Cutting edge of 35 kV disconnecting (d) Head of 220 kV current transformer switch  
图 5 基于 Otsu 的改进飞蛾扑火算法 (IMFO) 实验结果

Fig.5 Experimental results of Improved Moth-Flame Optimization Algorithm (IMFO) based on Otsu



(a) 负荷开关 (a) Load switch

(b) 架空线路夹 (b) Overhead line clips

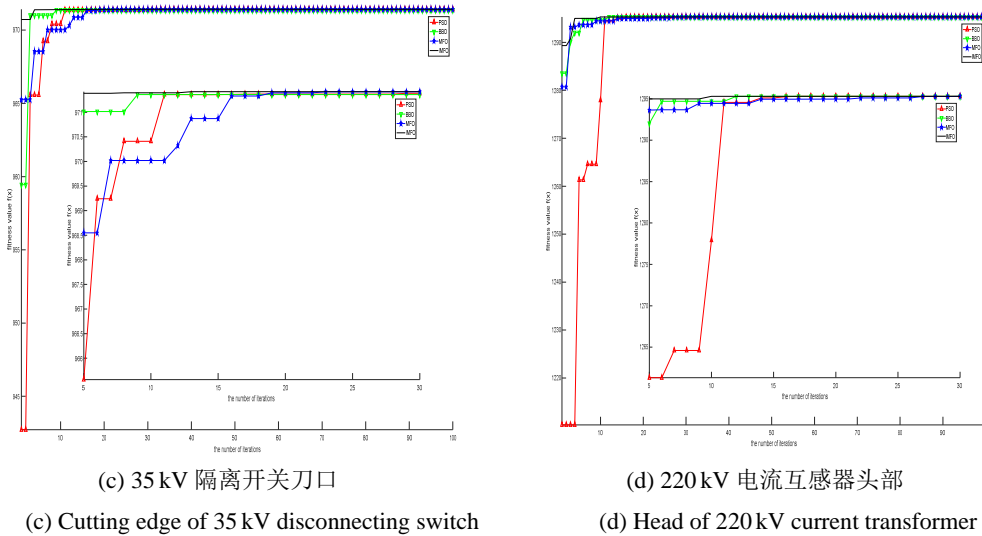


图6 二阈值分割时4种算法的适应度曲线

Fig.6 Fitness curves of the four algorithms in two-threshold segmentation

表1 基于 Otsu 的 PSO、BBO、MFO 和 IMFO 算法的运行时间

Table 1 Running time of PSO, BBO, MFO and IMFO algorithms based on Otsu

Infrared Image	the number of threshold	PSO	BBO	MFO	IMFO
		Time/s			
Load switch	2	1.2120	5.4511	5.2382	4.3017
	3	1.5600	5.6076	5.3620	4.4461
	4	1.6789	5.6290	5.5688	4.5889
	5	1.7322	5.7166	5.60157	4.6293
	6	1.8269	5.8577	5.7413	5.4121
Overhead line clips	2	1.2990	5.1919	5.1908	4.5029
	3	1.5538	5.3684	5.3387	4.5294
	4	1.1638	5.5027	5.4798	4.5635
	5	1.7872	5.6241	5.5907	4.7476
	6	1.8738	5.8356	5.7013	5.2253
Cutting edge of 35kV disconnecting switch	2	1.0130	5.0619	4.9941	4.2499
	3	1.1724	5.2295	5.0565	4.4117
	4	1.0267	5.5091	5.3143	4.4320
	5	1.5190	5.5777	5.4018	4.5334
	6	1.4943	5.6648	5.5432	4.8351
Head of 220kV current transformer	2	1.0514	5.5023	5.2689	4.4267
	3	1.1502	5.7031	5.4253	4.9980
	4	1.5730	5.8555	5.6967	5.0861
	5	1.6740	5.9458	5.7750	5.1250
	6	1.8763	6.2302	6.0332	5.2900

表 2 基于 Otsu 的 PSO、BBO、MFO 和 IMFO 算法的最佳阈值

Table 2 Optimal thresholds of PSO, BBO, MFO and IMFO algorithms based on Otsu

Infrared Image	the number of threshold	PSO	BBO	MFO	IMFO
		threshold			
Load switch	2	63,161	65,161	64,161	63,160
	3	34,88,175	34,88,180	35,89,174	34,88,174
	4	32,75,131,198	30,76,141,205	32,73,129,198	32,72,129,197
	5	31,66,111,159,211	35,66,112,157,212	32,67,110,157,210	31,67,111,159,211
	6	19,38,70,114,161,212	21,38,68,112,160,211	20,38,70,113,160,213	19,37,69,111,158,210
Overhead line clips	2	49,146	49,148	49,147	48,146
	3	39,106,188	42,113,192	41,108,189	39,106,188
	4	36,89,150,213	35,85,135,200	37,81,143,204	36,86,143,204
	5	24,53,97,151,208	26,58,106,160,212	26,54,98,151,208	25,54,98,151,208
	6	25,53,87,129,173,218	27,52,92,135,175,217	24,52,93,135,177,224	24,51,89,130,174,219
Cutting edge of 35kV disconnecting switch	2	36,119	36,119	37,119	36,118
	3	32,89,170	30,87,172	32,89,170	31,88,170
	4	20,45,100,176	21,46,93,173	21,45,101,177	21,48,105,180
	5	19,38,73,126,188	22,45,83,134,191	21,48,106,181,255	19,37,72,125,189
	6	19,36,66,112,161,210	19,37,62,105,147,200	20,37,69,115,164,213	19,34,56,91,139,196
Head of 220kV current transformer	2	37,131	39,131	38,131	37,130
	3	34,78,156	33,78,158	35,77,156	34,77,155
	4	33,74,130,198	33,75,132,201	26,48,80,156	33,74,129,196
	5	24,46,76,130,197	24,47,70,132,197	25,46,77,130,198	24,46,76,130,197
	6	24,46,63,85,134,200	22,45,64,85,128,191	25,47,76,120,171,216	24,46,75,121,169,213

表 3 基于 Otsu 的 PSO、BBO、MFO 和 IMFO 算法的适应度函数

Table 3 Fitness functions of PSO, BBO, MFO and IMFO algorithms based on Otsu

Infrared Image	the number of threshold	PSO	BBO	MFO	IMFO
		The value of fitness functions			
Load switch	2	2337.247	2337.1281	2337.247	2337.247
	3	2460.9688	2460.1655	2460.9688	2460.9688
	4	2510.3134	2507.4056	2510.4771	2510.4783
	5	2529.1932	2527.3613	2529.1158	2529.1868
	6	2548.5638	2548.2691	2548.511	2548.6362
Overhead line clips	2	764.9406	764.9264	764.9406	764.9406
	3	789.4618	789.3366	789.4614	789.4615
	4	796.8766	796.9242	797.1133	797.1133
	5	802.4645	802.3079	802.4647	802.4647
	6	805.2729	805.1028	805.3028	805.3028
Cutting edge of 35kV disconnecting switch	2	971.4179	971.4162	971.4179	971.4179
	3	1021.1907	1021.1467	1021.1907	1021.1907
	4	1047.0512	1046.875	1047.0668	1047.0668
	5	1061.7901	1060.0937	1061.8183	1061.8183
	6	1069.5513	1068.7841	1069.4928	1068.7433
Head of 220kV current transformer	2	1295.3049	1295.278	1295.3049	1295.3049
	3	1371.1606	1370.9676	1371.1606	1371.1606
	4	1393.221	1392.9042	1384.106	1393.2386
	5	1405.7227	1404.4958	1405.7227	1405.7227
	6	1411.4757	1410.8997	1411.6236	1411.6318

表4 基于 Otsu 的 PSO、BBO、MFO 和 IMFO 算法的 PSNR 和 SSIM 值

Table 4 PSNR and SSIM values of Otsu based PSO, BBO, MFO and IMFO algorithms

Infrared Image	Multilevel thresholding	PSO		BBO		MFO		IMFO	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Load switch	2	22.8091	0.7372	22.6471	0.7350	22.7504	0.7365	22.7579	0.7367
	3	26.4642	0.8385	26.2668	0.8357	26.3375	0.8373	26.4157	0.8386
	4	28.7683	0.8498	27.9310	0.8371	28.8336	0.8517	28.8540	0.8590
	5	30.2381	0.8558	28.8660	0.8455	30.0857	0.8485	30.1018	0.8529
	6	30.9614	0.8428	30.8314	0.8422	30.8586	0.8429	30.9742	0.8504
Overhead line clips	2	25.4921	0.8143	25.4242	0.8140	25.9915	0.8313	25.9974	0.8315
	3	29.7361	0.9040	28.8657	0.8868	29.7168	0.9039	29.7189	0.9039
	4	32.2456	0.9362	31.9381	0.9356	32.2135	0.9363	32.2781	0.9369
	5	37.2357	0.9830	36.1721	0.9838	37.4673	0.9860	37.4952	0.9866
	6	37.6828	0.9860	38.3632	0.9838	38.4812	0.9835	38.4854	0.9837
Cutting edge of 35kV disconnecting switch	2	26.1407	0.8900	26.1381	0.8899	26.1694	0.8902	26.1786	0.8989
	3	28.9700	0.9064	28.7482	0.9022	28.8954	0.9055	28.8961	0.9075
	4	30.2540	0.9022	30.3042	0.9025	30.4158	0.9026	30.4163	0.9054
	5	32.2417	0.9231	31.3575	0.9014	32.2602	0.9225	32.2845	0.9233
	6	33.5157	0.9266	33.3768	0.9237	33.5439	0.9273	33.5353	0.9331
Head of 220kV current transformer	2	23.2873	0.8846	23.2191	0.8740	23.3831	0.8856	23.3835	0.8857
	3	28.9014	0.9284	28.6734	0.9260	28.9659	0.9284	28.9889	0.9384
	4	31.1164	0.9320	29.9380	0.9287	29.6538	0.9500	30.5853	0.9311
	5	32.5874	0.9530	31.4858	0.9409	32.6798	0.9528	32.6831	0.9531
	6	33.7797	0.9534	33.5816	0.9483	33.7674	0.9536	33.7678	0.9540

### 4.3 结果分析

通过图 6 可以看出 PSO、BBO、MFO 和 IMFO 算法 4 种算法优化 Otsu 对红外图像二阈值分割时, 对比图中算法的收敛速度判断算法的优劣, IMFO 算法能够更快地收敛进而减少算法的执行时间。表 3 和表 4 对比 4 种算法的目标函数值、PSNR (Peak Signal to Noise Ratio) 和 SSIM (Structural Similarity) 值可以发现, 当阈值数小于等于 3 时 IMFO 和 MFO 算法分割效果相似, 当阈值数大于 3 时, IMFO 算法的优势显示出来, 表明 IMFO 算法有较好的分割效果。

### 5 总结

针对电力设备红外图像有噪声和对比度低的缺点, 传统阈值分割效果差, 提出了一种基于 Lévy 飞行的改进飞蛾扑火算法优化红外图像分割算法。使用智能算法得到最佳阈值, 使用得到阈值对图像进行分割, 以 PSNR 和 SSIM 两个评价指标评价阈值分割质量。PSNR 和 SSIM 值越大, 图像分割效果越好<sup>[18]</sup>。通过对比 PSO、BBO、MFO 和 IMFO 四种算法的评

价指标值, 表明改进飞蛾扑火算法 (IMFO) 在电力设备红外图像分割上具有一定的优势。准确分割出目标区域对电力设备故障诊断具有很大意义, 本文提出的算法能进一步推动电力设备故障诊断技术的发展。

### 参考文献:

- [1] Sarkar S, Das S, Chaudhuri S S. A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution[J]. *Pattern Recognition Letters*, 2015, **54**: 27-35.
- [2] 杨兆龙, 刘秉瀚. 基于改进差分进化算法的多阈值图像分割[J]. *计算机系统应用*, 2016, **25**(12): 199-203.  
YANG Zhaolong, LIU Binghan. Multi-threshold image segmentation based on improved differential evolution algorithm[J]. *Computer Systems & Applications*, 2016, **25**(12):199-203.
- [3] Sanyal N, Chatterjee A, Munshi S. An adaptive bacterial foraging algorithm for fuzzy entropy based image segmentation[J]. *Expert Systems with Applications*, 2011, **38**(12): 15489-15498.
- [4] LIU Y, HU K, ZHU Y, et al. Color image segmentation using multilevel



- thresholding- cooperative bacterial foraging algorithm[C]//*IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems*, 2015: 181-185.
- [5] Pare S, Bhandari A K, Kumar A, et al. A new technique for multilevel color image thresholding based on modified fuzzy entropy and Lévy flight firefly algorithm[J]. *Computers & Electrical Engineering*, 2018, **70**: 476-495.
- [6] Horng M H, Jiang T W . Multilevel Image Thresholding Selection Based on the Firefly Algorithm[C]//*Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing, IEEE*, 2010: 175-180.
- [7] Mohd Noor M H, Ahmad A R, Hussain Z, et al. Multilevel thresholding of gel electrophoresis images using firefly algorithm[C]//*IEEE International Conference on Control System, Computing and Engineering*, 2011: 18-21.
- [8] HE L, HUANG S. Modified firefly algorithm based multilevel thresholding for color image segmentation[J]. *Neurocomputing*, 2017, **240**: 152-174.
- [9] ZHANG S, JIANG W, Satoh S. Multilevel Thresholding Color Image Segmentation Using a Modified Artificial Bee Colony Algorithm[J]. *IEICE Transactions on Information and Systems*, 2018, **E101.D(8)**: 2064-2071.
- [10] Horng M-H. Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation[J]. *Expert Systems with Applications*, 2011, **38(11)**: 13785-13791.
- [11] Karaboga D, Akay B. A comparative study of Artificial Bee Colony algorithm[J]. *Applied Mathematics and Computation*, 2009, **214(1)**: 108-132.
- [12] 高宏进, 王力, 龚维印, 等. 基于改进 CS 算法的二维 Ostu 快速图像分割[J]. *通信技术*, 2017, **50(12)**: 2698-2703.
- GAO Hongjin, WANG Li, GONG Weiyin, et al. 2d Ostu fast image segmentation based on improved CS algorithm[J]. *Communications Technology*, 2017, **50(12)**: 2698-2703.
- [13] 杨晓, 胡可杨, 汪烈军, 等. 基于布谷鸟优化的三维 OTSU 图像分割算法[J]. *新疆大学学报: 自然科学版*, 2017, **34(4)**: 452-458.
- YANG Xiao, HU Keyang, WANG Lijun, et al. 3d OTSU image segmentation algorithm based on cuckoo optimization[J]. *Journal of Xinjiang University: Natural Science Edition*, 2017, **34(4)**: 452-458.
- [14] 卫洪春. 基于混合 PSO-CS 算法的彩色图像多阈值分割[J]. *计算机与现代化*, 2017(8): 61-65.
- WEI Hongchun. Multi-threshold segmentation of color images based on hybrid pso-cs algorithm [J]. *Computer and Modernization*, 2017(8): 61-65.
- [15] 尹晓叶, 李俊吉. 基于增强布谷鸟搜索的图像分割算法[J]. *控制工程*, 2017, **24(10)**: 2118-2124.
- YIN Xiaoye, LI Junji. Image segmentation algorithm based on enhanced cuckoo search [J]. *Control Engineering of China*, 2017, **24(10)**: 2118-2124.
- [16] Otsu N . A Threshold Selection Method from Gray-Level Histograms[J]. *IEEE Transactions on Systems Man & Cybernetics*, 2007, **9(1)**: 62-66.
- [17] Mirjalili S. Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm[J]. *Knowledge Based Systems*, 2015, **89(11)**: 228-249.
- [18] HE L, HUANG S. Improved Glowworm Swarm Optimization Algorithm for Multilevel Color Image Thresholding Problem[J]. *Mathematical Problems in Engineering*, 2016, **2016**: 1-24.