

DOI:10.19651/j.cnki.emt.2518233

# 基于IZOA结合最小交叉熵的图像分割算法<sup>\*</sup>

刘庭亭 何志琴

(贵州大学电气工程学院 贵阳 550025)

**摘要:** 针对图像多阈值分割过程中存在的分割精度低、效率低、随着阈值增加分割效果不稳定等问题,提出了一种基于改进斑马优化算法(IZOA)的多阈值图像分割算法。首先,利用混沌映射方法初始化种群;其次,引入邻域波动策略精细化搜索;然后,结合杂交与变异操作生成新的解,提高算法全局搜索能力;再采用精英保存策略保留最优解。使用图像分割前后得到的最小对称交叉熵作为适应度函数进行多阈值分割,表现出了更高的分割精度、分割效率以及分割的稳定性。实验结果表明,与ZOA、GWO、WOA等算法对比,基于IZOA分割图像的质量指标FSIM、SSIM和PSNR方面具有显著优势,最优截断均值占比分别达到91.7%、88.9%、100%。

**关键词:** 图像分割;斑马优化算法;最小对称交叉熵;多策略改进

**中图分类号:** TN919.82; TP391.41 **文献标识码:** A **国家标准学科分类代码:** 520.6040

## Based on IZOA combined with minimum cross-entropy image segmentation algorithm

Liu Tingting He Zhiqin

(Electrical Engineering College, Guizhou University, Guiyang 550025, China)

**Abstract:** To address the issues of low segmentation accuracy, low efficiency, and unstable segmentation results with increasing thresholds in color image multi-threshold segmentation, an improved multi-threshold image segmentation algorithm based on the improved zebra optimization algorithm (IZOA) is proposed. Firstly, a chaotic mapping method is used to initialize the population; secondly, a neighborhood fluctuation strategy is introduced for fine searching; then, hybridization and mutation operations are combined to generate new solutions, enhancing the global search capability of the algorithm; finally, an elite retention strategy is employed to preserve the optimal solution. The minimum symmetric cross-entropy obtained before and after image segmentation is utilized as the fitness function for multi-threshold segmentation, demonstrating higher segmentation accuracy, efficiency, and stability. Experimental results show that compared with ZOA, GWO, WOA, and other algorithms, the image quality indices FSIM, SSIM, and PSNR achieved by the IZOA-based segmentation exhibit significant advantages, with the optimal truncation mean proportions reaching 91.7%, 88.9% and 100%, respectively.

**Keywords:** color image segmentation; zebra optimization algorithm; minimum symmetric cross-entropy; multi strategy improvement

## 0 引言

随着图像处理技术的发展,图像分割逐渐成为研究的重点对象<sup>[1]</sup>。图像分割是图像进一步处理的必要前提,被广泛应用于农业作物监测、医学影像分割分析、交通运输控制、火灾实时监测等多个领域<sup>[2]</sup>。分割方法主要包括阈值分割<sup>[3]</sup>、聚类分割<sup>[4]</sup>、区域分割<sup>[5]</sup>还有深度学习人工神经网络等<sup>[6-7]</sup>。其中,拥有简单、稳定、高效特点的阈值分割成为

众多学者的热门研究方向。常见阈值分割方法包括:最大类间方差法、最小交叉熵法、最大熵法<sup>[8]</sup>,还有最小对称交叉熵法等<sup>[9]</sup>。

传统的阈值分割方法在面对复杂的多阈值图像分割问题时,存在分割效率低、计算时间长以及精度不足等问题。为了解决这些问题,智能优化算法<sup>[10]</sup>被广泛应用于图像分割领域<sup>[11]</sup>。虽然智能算法的引入取得不错的分割效果,但仍然存在改进空间。例如,Erdmann等<sup>[12]</sup>使用萤火虫算法

收稿日期:2025-03-03

<sup>\*</sup> 基金项目:贵州省科技支撑计划资助项目(黔科合支撑[2023]一般096、黔科合支撑[2024]一般007)、贵州省科技支撑计划“高效/碳效建筑节能柔性调控关键技术研究与应用研发”(黔科合支撑[2023]一般409)资助

在多阈值分割中取得了接近手动分割的效果,但是对于图像的渐变部分分割表现不佳,Wang等<sup>[13]</sup>提出的基于莱维飞行改进樽海鞘算法(slap swarm algorithm, SSA)在多阈值分割中获得优异的实验结果,但是对于像9个、10个等多阈值分割时容易陷入局部最优。吕鑫等<sup>[14]</sup>利用基于改进的麻雀搜索算法(improved sparrow search algorithm, ISSA)的多阈值分割方法,虽然展示了分割稳定性,但是并未对比分割后精度问题。Yao等<sup>[15]</sup>将佳点集<sup>[16]</sup>应用到灰狼优化(grey wolf optimization, GWO)算法中提升了分割质量,但是对比的算法较少,不足以展示论文算法的优异。

为了改进上述不足之处,本文提出了一种基于改进斑马优化算法(improved zebra optimization algorithm, IZOA)结合最小交叉熵的多阈值图像分割方法。斑马优化算法(zebra optimization algorithm, ZOA)<sup>[17]</sup>是模仿斑马在觅食和防御捕食者时的行为,通过觅食行为和防御策略实现全局搜索和局部开发的平衡。然而,标准ZOA在处理复杂的多阈值图像分割问题时,仍存在分割精度不高,容易陷入局部最优解、优化收敛速度慢等问题。本文对ZOA进行以下改进:首先引入了Logistic混沌映射<sup>[18]</sup>来初始化种群,增强了种群的遍历性和多样性;其次利用邻域波动策略<sup>[19]</sup>、杂交和变异策略<sup>[20]</sup>帮助算法跳出局部最优解。此外,本文还引入了精英保留策略<sup>[21]</sup>,确保每一代中都能保留表现最好的部分个体,从而加速算法的收敛并提高最终解的质量。通过这些改进,IZOA在处理复杂的多阈值图像分割问题时表现出了更高的分割效率和精度。

本文采用Berkeley图像库中的9张图像进行图像多阈值分割实验,并对比了ZOA、GWO、鲸鱼优化算法(whale optimization algorithm, WOA)、金豺优化算法(golden jackal optimization, GJO)、白鲸优化算法(beluga whale optimization, BWO)5种智能算法。结果表明,改进的斑马优化算法(IZOA)在图像分割后得到的特征相似度指数(FSIM)、结构相似性指数(SSIM)和峰值信噪比(PSNR)方面均优于其他智能算法,同时能够稳定快速地完成多阈值分割问题。

## 1 斑马优化算法

ZOA是一种模仿斑马觅食和防御行为的自然启发式算法。主要通过两个阶段实现优化:觅食阶段和防御阶段。该算法的保卫猎物阶段具体包括以下几个方面:

1)初始化ZOA参数,在所寻优的空间里面随机初始化种群,如式(1)所示。

$$x_{i,j} = lb_j + r \times (ub_j - lb_j) \quad (1)$$

其中, $ub_j$ 与 $lb_j$ 分别是寻优的上下边界, $r$ 为 $[0,1]$ 之间的随机数, $x_{i,j}$ 为第 $i$ 只斑马在第 $j$ 维的位置。

2)觅食行为模拟斑马在草原上寻找食物的过程。在这个阶段,斑马根据距离最优位置的远近来调整自己的位置。如式(2)所示。

$$x_{i,j}^{t+1} = x_{i,j}^t + r \times (PZ_j - I \times x_{i,j}^t) \quad (2)$$

$$\mathbf{X}_i^{t+1} = \begin{cases} \mathbf{X}_i^{t+1}, \mathbf{F}_i^{t+1} < \mathbf{F}_i \\ \mathbf{X}_i^t, \text{其他} \end{cases} \quad (3)$$

其中, $x_{i,j}^{t+1}$ 是第 $j$ 维第 $i$ 个斑马新的位置, $t$ 是迭代次数。 $PZ$ 是当前种群中的先锋斑马。 $r$ 是在 $[0,1]$ 的随机数,用于增加位置更新的随机性, $I$ 属于集合 $\{1,2\}$ 之间的随机值, $\mathbf{F}_i$ 为所有维度中第 $i$ 只斑马的适应度函数值, $\mathbf{F}_i^{t+1}$ 为适应度函数值, $\mathbf{X}_i^t$ 是当前代所有维度的第 $i$ 个斑马位置。

3)斑马的防御策略因捕食者的种类而异。当狮子对斑马发起攻击时,斑马的防御策略是随机的转弯动作以Z字形逃跑。而面对小型捕食者的攻击时斑马采取攻击策略。在ZOA中,两类捕食者出现的概率相同。

$$x_{i,j}^{t+1} = \begin{cases} S_1: & x_{i,j}^t + R \times (2r - 1) \times \\ & \left(1 - \frac{t}{T}\right) \times x_{i,j}^t, & P_s \leq 0.5 \\ S_2: & x_{i,j}^t + r \times (AZ_j - I \times x_{i,j}^t), & \text{其他} \end{cases} \quad (4)$$

$$\mathbf{X}_i^{t+1} = \begin{cases} \mathbf{X}_i^{t+1}, \mathbf{F}_i^{t+1} < \mathbf{F}_i \\ \mathbf{X}_i^t, \text{其他} \end{cases} \quad (5)$$

该策略数学模型如式(4), $P_s$ 是 $[0,1]$ 之间的随机数,当 $P_s$ 小于0.5时,斑马采用S1策略逃跑,否则采用S2策略聚集,当新位置的适应度函数值更优时,则更新当前位置。其中 $t$ 为迭代次数; $T$ 为最大迭代次数; $R$ 为0.01的常数; $AZ$ 为被攻击的斑马的 $j$ 维位置。

## 2 改进斑马优化算法

针对ZOA在面对图像多阈值分割时容易陷入局部最优,收敛速度慢等问题,本文从以下几个方面进行改进:1)引入了混沌映射初始化方法进行种群的初始化。2)引入邻域波动策略,能够在局部进行更加精细的搜索。3)引入杂交与变异生成新的解,以提高算法的多样性和全局搜索能力。4)采用精英保存策略保留优异个体直接进入下一代。因而得到改进斑马优化算法(IZOA)。

### 2.1 混沌映射初始化种群

传统的斑马优化算法(ZOA)通常采用随机分布方式进行种群初始化,这种方法无法充分体现ZOA种群的多样性。为了解决这一问题,本文提出采用Logistic混沌映射来初始化ZOA的种群位置,文献<sup>[18]</sup>指出,混沌映射是一种均匀且有效的选点方法。与随机选点相比,混沌映射能够更均匀地分布在搜索空间中,从而加快收敛速度。如图1随机初始化与混沌映射初始化在3维空间内的种群分布情况,控制参数 $r$ 为3.995,种群数为1500,搜索范围为 $255 \times 255 \times 255$ 的三维空间。可以看出,混沌映射初始化在搜索范围内的点分布更加均匀,空间利用率更高。通过引入混沌序列初始化ZOA种群,提升了ZOA种群的多样

性,有助于算法在早期阶段进行更全面的全局搜索,从而提高收敛速度。

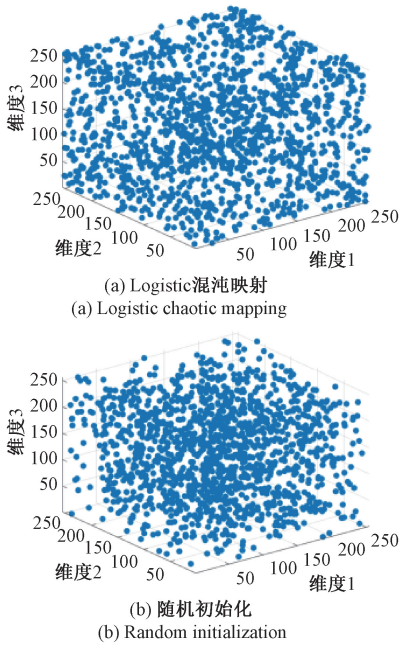


图 1 Logistic混沌映射与随机初始化对比图

Fig. 1 Comparison of logistic chaotic mapping and random initialization

## 2.2 邻域波动策略

ZOA 算法在寻优过程中为了避免陷入局部最优解,本文引入了邻域波动策略,邻域波动策略帮助算法在全局范围内进行更有效的搜索,增加了找到全局最优解的可能性。它通过不断引入随机扰动来调整搜索方向,从而增强了探索能力,如式(6)所示,这种策略还帮助算法在探索新区域和开发已有区域之间保持平衡,确保搜索过程不会过于集中于某个狭小的区域,使得 ZOA 在处理复杂优化问题时,具有更强的鲁棒性和适应性。

$$\mathbf{X}_i^{t+1} = \mathbf{X}_i^t + \mathbf{Q} \times (u - 0.5) \times \mathbf{X}_i^t \quad (6)$$

其中,  $\mathbf{Q}$  是一个调节参数,控制扰动的幅度,  $u$  是一个与当前解的维度相同的随机向量,每个分量服从均匀分布  $U(0,1)$ 。

## 2.3 杂交与变异策略

为在寻优过程中提高算法多样性和全局搜索能力,IZOA 引入了杂交变异策略,通过结合杂交和变异策略,可以提高斑马优化算法的多样性和全局搜索能力,从而更好地解决复杂的图像多阈值图像分割问题,具体公式如式(7)所示。

$$C_o = \begin{cases} \mathbf{P}_1 \times \partial + \mathbf{P}_2 \times (1 - \partial), & r_1 < P_c \\ \mathbf{P}_1, & \text{其他} \end{cases} \quad (7)$$

其中,  $C_o$  为后代,  $\mathbf{P}_1$  代表父代 1,  $\mathbf{P}_2$  代表父代 2,  $r_1$  是  $[0,1]$  间的随机数,  $P_c$  是杂交的概率。

$$x_{i,j}^{t+1} = \begin{cases} l_b + rand \times (u_b - l_b), & r_2 < p_m \\ C_{o,i,j}^{t+1}, & r_2 \geq p_m \end{cases} \quad (8)$$

经过式(7)与(8)最终得到后代,其中,  $x$  为最终变异后代。  $l_b$  为搜索的下界,  $u_b$  是寻优的上界。  $rand$  为  $[0,1]$  间的随机值,  $P_m$  为变异概率,  $r_2$  为  $[0,1]$  间的随机值,用于随机选择变异与保留。

## 2.4 精英保留策略

为了获取更好的分割效果,IZOA 算法在增加的策略改动中都增加了动态精英保留策略,具体而言,本文动态调整精英保留比例,在进化初期保留较多的精英个体(20%),以加速搜索,在进化过程中逐步减少精英保留比例(5%),以维持种群的多样性。另外,结合 2.2 的邻域波动策略以及 2.3 的杂交与变异策略能够进一步增加种群的多样性。通过上述策略有效平衡了精英保留与种群多样性之间的关系,避免了早熟收敛问题。

$$\mathbf{X}^{t+1} = \mathbf{E}^t \cup \mathbf{G}^{t+1} \quad (9)$$

其中,  $\mathbf{X}^{t+1}$  表示下一代的种群,由两部分组成,其中  $\mathbf{E}^t$  是表示当前代  $t$  中被保留下来的精英个体集合,  $\mathbf{G}^{t+1}$  表示通过常规的交叉、变异等操作生成的新一代个体集合。这样可以保证这些最优个体的基因信息不会丢失,从而提高算法的性能和最终解的质量。

## 3 多阈值图像分割

### 3.1 最小对称交叉熵阈值分割

对称交叉熵是一种衡量两个概率分布相似程度的度量方法。在图像分割中,对称交叉熵可以用来衡量图像中各个区域灰度分布的差异,通过最小化这种差异,可以获得最优的分割阈值。

$$SCE(P, Q) = D_{KL}(P \parallel Q) + D_{KL}(Q \parallel P) \quad (10)$$

通过式(10)可知交叉熵的定义可以表示成两个散度的和,即表示分布  $P$  相对于分布  $Q$  的差异之和。

将图像的多个区域进行散度求和得到对称交叉熵的公式如式(11)所示。

$$H = \sum p(i') \left[ i' \log\left(\frac{i'}{u}\right) + u \log\left(\frac{u}{i'}\right) \right] \quad (11)$$

其中,  $p(i')$  是概率密度函数,  $i'$  是灰度级别,其范围是  $[0, L]$ ,  $u$  是当前区域的灰度值均值。

因此,选择多个阈值将图像划分为若干个区域。假设选择  $k$  个阈值,大小为  $\mathbf{T} = \{t_1, t_1, \dots, t_k\}$  那么图像被划分为  $k+1$  个区域。假设图像的灰度级别范围  $0 \sim 255$ , 其中一个区域的直方图表示为:

$$\mathbf{F} = \{f_0, f_1, \dots, f_{L-1}\} \quad (12)$$

其中,  $\mathbf{F}$  表示灰度级别为  $[0, L]$  的像素数量。对于每个区域计算图像中各灰度级别的概率密度函数值用式(13)表示,计算方式用式(14)进行计算。

$$\mathbf{P} = \{p_0, p_1, \dots, p_{L-1}\} \quad (13)$$

$$p_{i'} = \frac{h_{i'}}{N} \quad (14)$$

其中,  $N$  是图像总像素,对于第一个区域,其当前区域

的像素灰度值均值如式(15)。

$$u_0 = \frac{\sum_{i=1}^{thresh} i \times p(i)}{\sum_{i=1}^{thresh} p(i)} \quad (15)$$

所以该区域的对称交叉熵计算如式(16)。

$$D_c(t_1) = \sum_{i=1}^{t_1} p(i) \left[ i \log\left(\frac{i}{u_0}\right) + u_0 \log\left(\frac{u_0}{i}\right) \right] \quad (16)$$

优化过程中需要选择一组最好的阈值  $T$  使得总的对称交叉熵最小,因此,式(17)就可以作为寻优算法的适应度函数。

$$T' = \arg \min_T \sum_{i=1}^k D_c(t_i) \quad (17)$$

### 3.2 基于IZOA最小对称交叉熵图像阈值分割算法流程

分割算法执行步骤描述如下:

- 1) 读取图像,设置阈值个数,使用式(11)作为适应度函数。设置参数,包括种群数量,最大迭代次数等;
- 2) 使用混沌映射初始化种群;
- 3) 在觅食阶段根据式(2)、(3)更新种群位置,在防御阶段根据式(4)、(5)更新种群位置;采用式(6)~(9)进行位置保留与优化,避免陷入局部最优;
- 4) 计算适应度值并获取全局最优位置;
- 5) 判断是否满足终止寻优条件,不满足则跳转到步骤3,否则结束寻优循环;
- 6) 将获取的阈值对读取的图像进行分割;
- 7) 程序结束,输出分割后的图像与分割的阈值。

## 4 图像分割实验与分析

### 4.1 实验背景

实验选择取 Berkeley 图像库中的 9 幅涉及人物、动物、

$$F_{FSIM} = \frac{\sum_{x \in \Omega} \left[ \frac{2P_{PC,1}(x)P_{PC,2}(x) + T_1}{P_{PC,1}^2(x) + P_{PC,2}^2(x) + T_1} \right]^\alpha \left[ \frac{2C_1(x)C_2(x) + T}{C_1^2(x) + C_2^2(x) + T_2} \right]^\beta P_{PC,m}(x)}{\sum_{x \in \Omega} P_{PC,m}(x)} \quad (22)$$

其中,  $P_{PC,m}(x) = \max(P_{PC,1}(x), P_{PC,2}(x))$ ;  $\alpha = 1$ ;  $\beta = 1$ 。

PSNR 是一种常用的图像质量评价指标,主要用于量化图像压缩或其他处理后的图像与原始图像之间的差异。它反映的是图像的峰值信号与噪声之间的比值。其计算公式如式(23)。

$$PSNR = 10 \log_{10} \left( \frac{MAX^2}{MSE} \right) \quad (23)$$

其中, MAX 表示图像的最大灰度值,定义如下:

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N [I(i,j) - K(i,j)]^2 \quad (24)$$

### 4.2 实验设计

本文实验在 4.70 GHz CPU 和 16 GB 内存的 PC 机、Windows11 的操作系统以及 MATLAB R2022b 环境中进行。选择 9 张伯克利测试图片(如图 3 中的原图所示),图 2 为图 3 中的 9 幅测试图像的灰度级分布图,可以看出

建筑、花草等复杂背景的实验图像作为分割实验对象,并将 ZOA、GWO、WOA、GJO、BWO 与 IZOA 进行分割实验对比,通过最优适应度收敛曲线检验算法效率,使用多次实验结果的标准差检验算法稳定性,以及原图像与分割图像的峰值信噪比(peak signal to noise ratio, PSNR)、结构相似性指数(structural similarity index, SSIM)以及特征相似度(feature similarity index measure, FSIM)的截断均值评估分割效果。

SSIM 是一种基于感知模型的图像质量评估指标,其主要目的是衡量两幅图像在结构上的相似度。SSIM 假设人类视觉系统对结构信息更加敏感,因此它通过对比亮度、对比度和结构 3 个方面来评估图像质量。

$$SSIM(x,y) = [l(x,y)]^\alpha \times [c(x,y)]^\beta \times [s(x,y)]^\gamma \quad (18)$$

SSIM 计算公式如式(18)所示,其中  $l(x,y)$ 、 $c(x,y)$ 、 $s(x,y)$  分别表示亮度、对比度和结构的比较函数:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (19)$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (20)$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \quad (21)$$

$\mu_x$  和  $\mu_y$  是图像  $x$  和  $y$  的均值,  $\sigma_x$  和  $\sigma_y$  是图像  $x$  和  $y$  的标准差,  $\sigma_{xy}$  是图像  $x$  和  $y$  的协方差,  $C_1$ 、 $C_2$ 、 $C_3$  是稳定常数,通常取较小的值以避免分母为零。

FSIM 是一种基于图像特征的质量评估指标,其核心思想是通过主要通过比较图像的低位特征(如相位一致性和梯度幅值)来评估图像质量,计算公式如式(22)所示。

不同类型的测试图像具有不同的灰度级,其包含了绝大部分图像的灰度级分布情况。实验中优化算法的种群数量均为 30,迭代次数为 100,所选算法均独立运行 30 次实验中,各算法参数取值如表 1 所示。为了更直观地感受 IZOA 分割的优异性,当阈值个数  $m$  为 4 时进行分割并呈现于图 3,并将部分测试图像的局部分割情况进行放大并呈现于图 4。同时为了展示 IZOA 结合最小对称交叉熵的多阈值分割优异性,本文还进行了阈值  $m$  为 7、8、9 和 10 时的分割,将实验结果去掉最好与最差并求取平均值得到截断均值,并将实验结果以表格的形式进行呈现。

### 4.3 ZOA 在图像分割中的改进空间分析

虽然从图 4 来看 ZOA 的分割效果表现一般。例如,在第 1 张图里对于草地的分割效果比 GJO 分割效果差很多;第 2 张图里对于天空的渐变色以及建筑轮廓分割表现不佳;第 3、4 张图里 ZOA 的表现也较为平凡。但是从

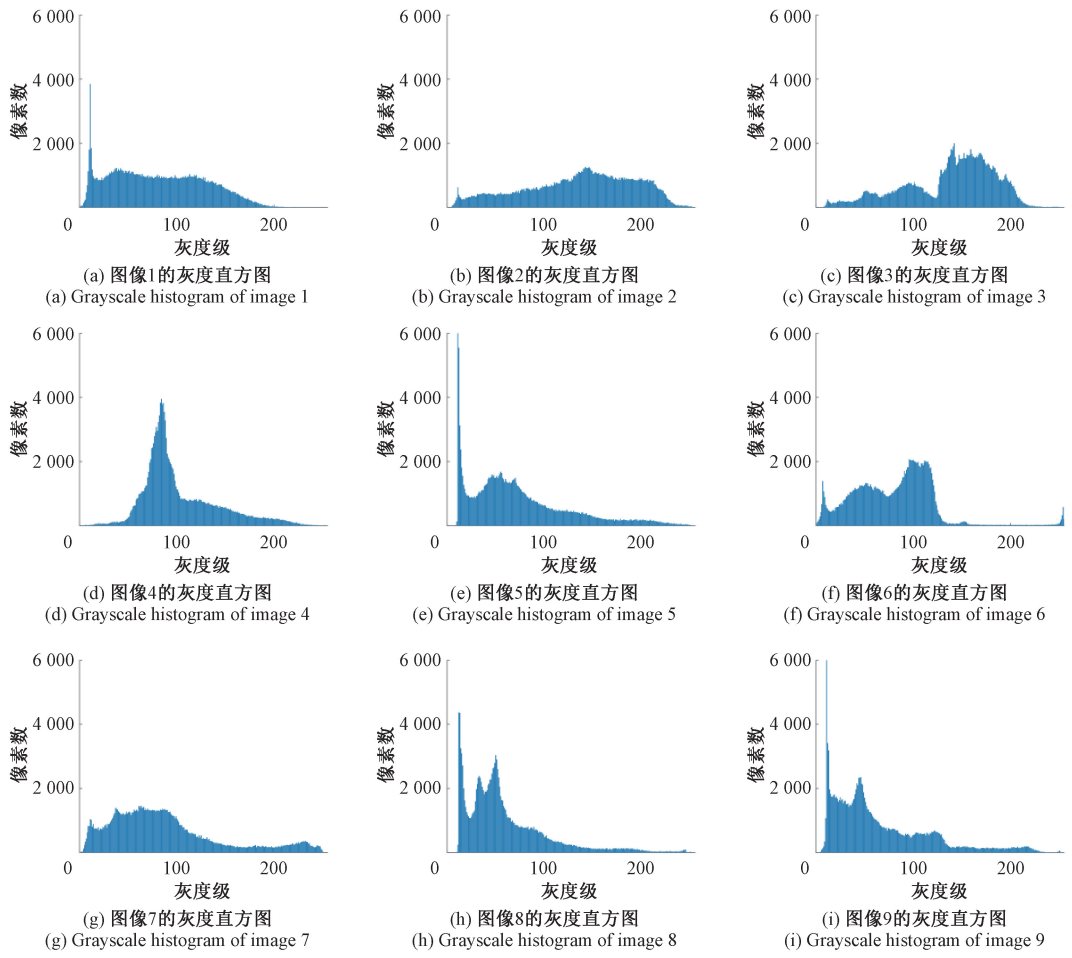
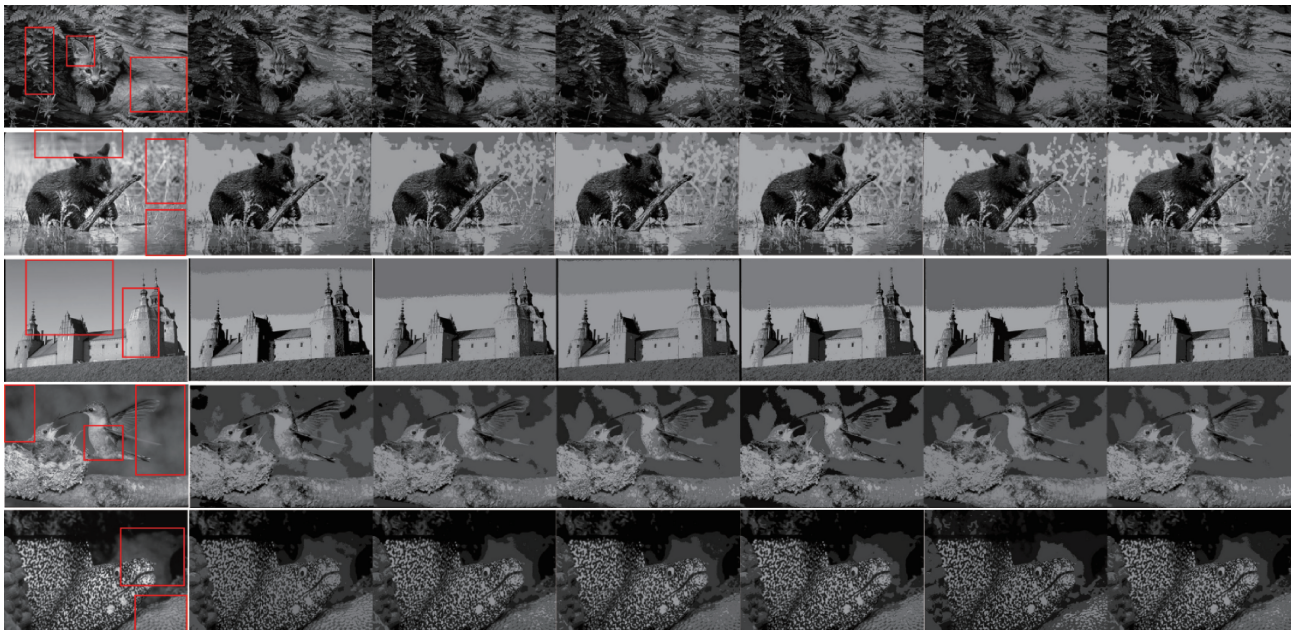


图 2 测试图集灰度级分布图

Fig. 2 Grayscale distribution of the test dataset



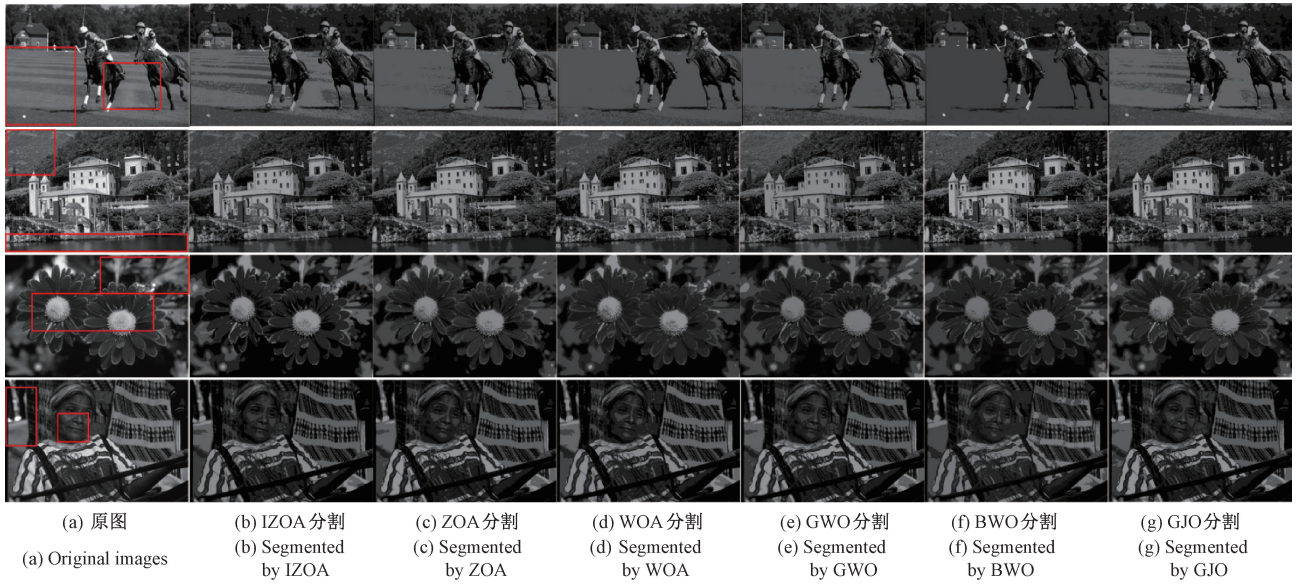
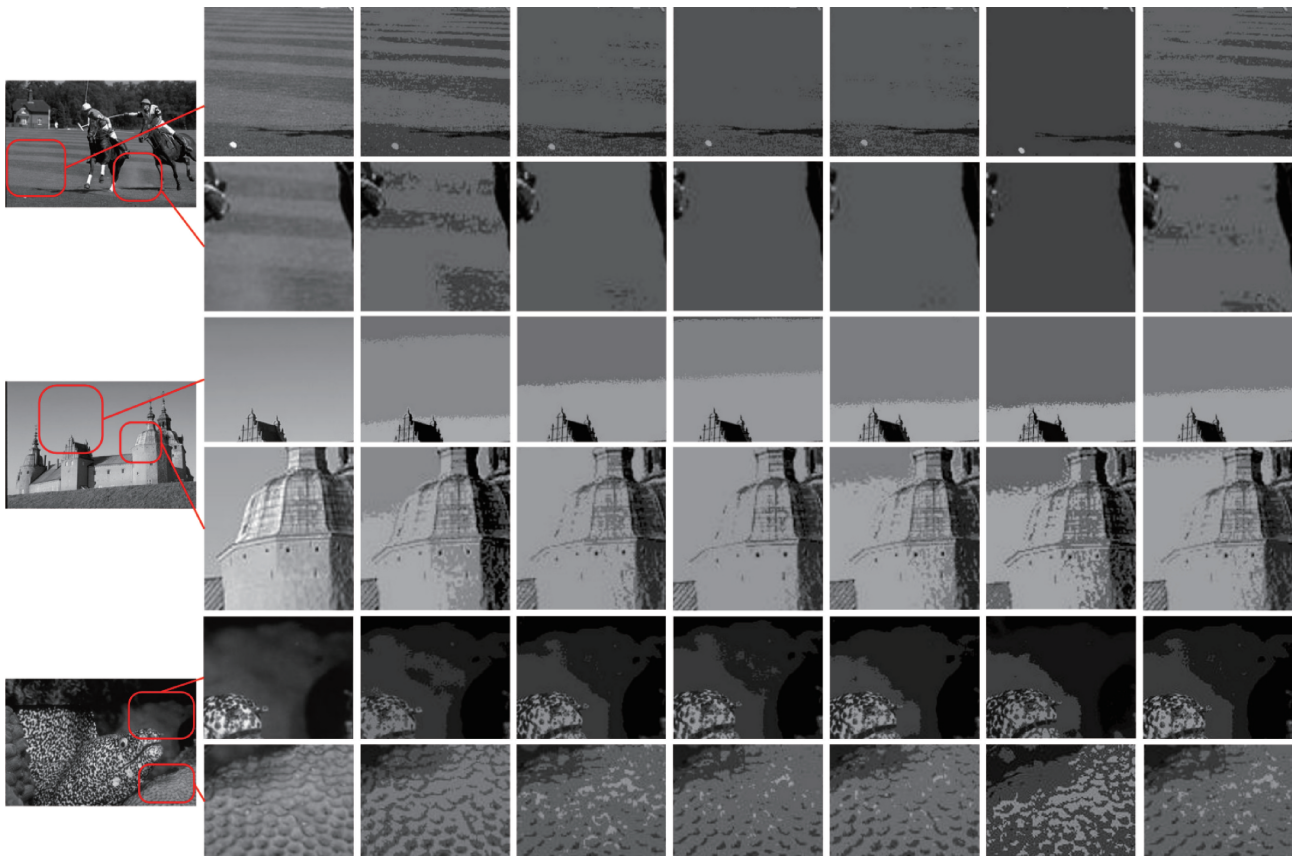


图3 测试图集基于各种算法的多阈值分割

Fig. 3 Multi-Threshold segmentation of the test image set using various algorithms



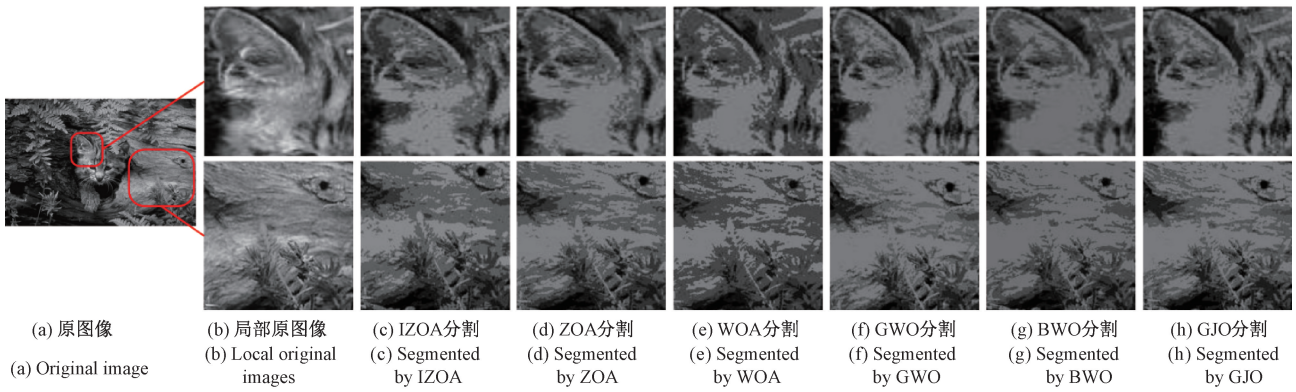


图 4 部分测试图的局部放大分割对比图

Fig. 4 Locally magnified segmentation comparison of selected test images

表 1 实验对比算法常用参数设定

Table 1 Default parameter settings of comparative algorithms in experiments

| 算法   | 参数设置   |
|------|--|
| WOA  | $b=1$  |
| GWO  | $a \in [0, 2]$ ;<br>$r1 \in [0, 1]; r2 \in [0, 1]$ |
| GJO  | $C1=1.5$   |
| BWO  | $\beta=1.5$  |
| IZOA | $R=0.05$   |
| ZOA  | $R=0.1$  |

图 5、6 来看,ZOA 的收敛速率以及分割的稳定性表现相较于其他对比算法并不是最差的。因此,ZOA 在图像分割中具有改进的意义与改进的空间。

#### 4.4 分割效果

由于篇幅原因,不同算法的详细分割效果以数据形式进行展示,同时为了能更直观地看出 IZOA 在图像分割中的优异性,图 3 中(左边从上到下,依次对应测试原图 1~9)列出测试图集在各个对比算法中 I 阈值个数为 4 时的分割效果,其中 IZOA 的分割情况表现优异,例如在原图各图像里圈出的各部分中 IZOA 在整体上要优于其他的对比算法。进一步可以从图 4 中更能直观地看到 IZOA 分割的优异性,例如如图 4 中首张图里对于草坪的分割,明显 IZOA 对

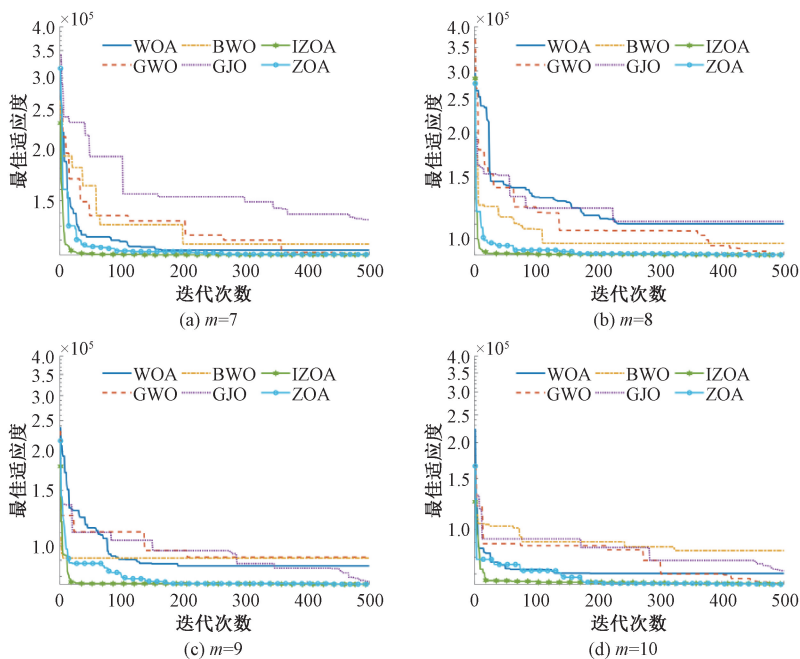
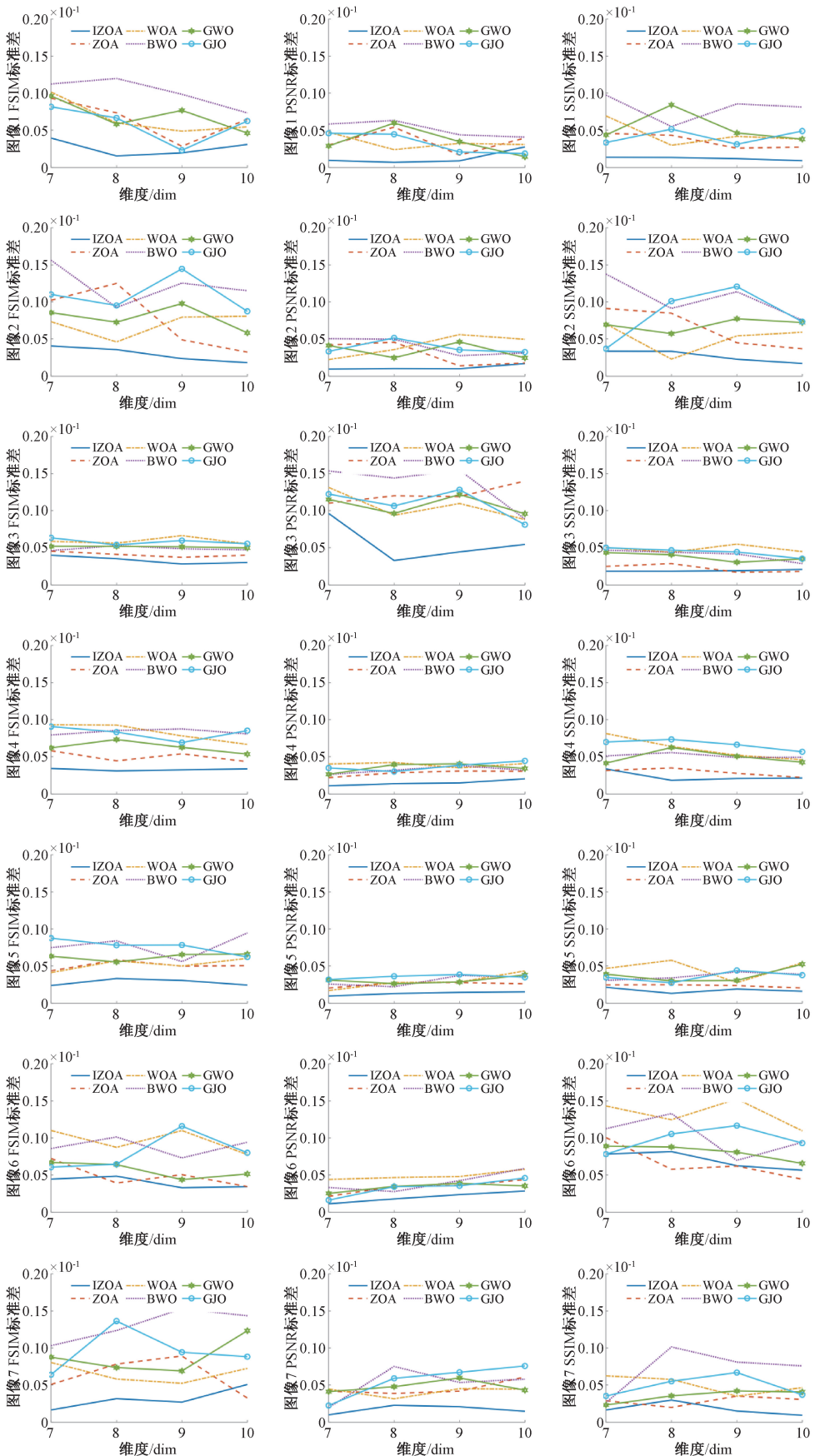


图 5 各个算法在图像 1 的不同阈值个数(m)下进行分割的收敛曲线

Fig. 5 Convergence curves of various algorithms for segmentation at different threshold numbers (m) of image 1



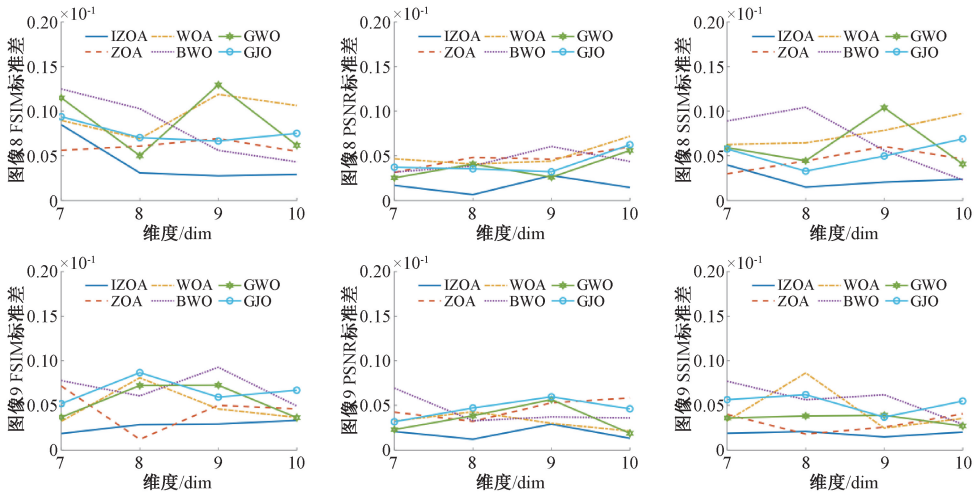


图 6 不同算法分割的图像 FSIM、PSNR、SSIM 标准差对比

Fig. 6 Standard deviation comparison of FSIM, PSNR, and SSIM metrics across segmentation algorithms

于原图的草坪还原更加优秀,又如图 4 中第 2 张图里的天空的渐变与建筑轮廓、阴影,明显 IZOA 分割的解析力更好,再如第 3 张图中猫的脸颊、岩石和植物,以及第 4 张图中对于

背景阴影和石头纹理的分割细节的表现来看,IZOA 相较于其他对比算法表现优异。另外在表 2、3、4 中也可以看出 3 个指标均能证明 IZOA 在图像分割中的优异表现。

表 2 不同算法不同阈值个数的 FSIM 截断均值对比

Table 2 Comparison of FSIM truncated means for different numbers of thresholds in various algorithms

| 图像 | 阈值数量   | FSIM            |                 |                 |          |          |          |
|----|--------|-----------------|-----------------|-----------------|----------|----------|----------|
|    |        | IZOA            | ZOA             | WOA             | BWO      | GWO      | GJO      |
| 1  | $m=7$  | <b>0.956 64</b> | 0.955 71        | 0.954 9         | 0.945 14 | 0.955 41 | 0.952 73 |
|    | $m=8$  | <b>0.965 16</b> | 0.964 22        | 0.963 53        | 0.953 35 | 0.964 75 | 0.960 83 |
|    | $m=9$  | <b>0.971 38</b> | 0.969 94        | 0.968 8         | 0.947 69 | 0.971 23 | 0.968 3  |
|    | $m=10$ | <b>0.976 00</b> | 0.974 44        | 0.972 22        | 0.952 63 | 0.974 72 | 0.971 15 |
| 2  | $m=7$  | <b>0.922 21</b> | 0.921 97        | 0.919 97        | 0.907 58 | 0.920 4  | 0.915 36 |
|    | $m=8$  | <b>0.937 61</b> | 0.936 89        | 0.935 61        | 0.923 39 | 0.933 47 | 0.929 93 |
|    | $m=9$  | <b>0.948 14</b> | 0.946 52        | 0.945 83        | 0.916 7  | 0.946 25 | 0.940 41 |
|    | $m=10$ | <b>0.953 91</b> | 0.953 13        | 0.951 75        | 0.923 62 | 0.953 44 | 0.949 08 |
| 3  | $m=7$  | <b>0.955 65</b> | 0.955 19        | 0.953 36        | 0.945 56 | 0.955 26 | 0.952 09 |
|    | $m=8$  | <b>0.962 64</b> | 0.962 45        | 0.960 54        | 0.954 78 | 0.961 74 | 0.959 56 |
|    | $m=9$  | <b>0.970 11</b> | 0.968 46        | 0.967 64        | 0.949 84 | 0.968 22 | 0.965    |
|    | $m=10$ | <b>0.973 75</b> | 0.972 17        | 0.971 06        | 0.954 65 | 0.972 67 | 0.968 64 |
| 4  | $m=7$  | 0.930 34        | <b>0.931 86</b> | 0.930 06        | 0.921    | 0.930 01 | 0.927 79 |
|    | $m=8$  | <b>0.945 05</b> | 0.944 19        | 0.940 34        | 0.934 53 | 0.943 61 | 0.934 73 |
|    | $m=9$  | <b>0.953 16</b> | 0.952 04        | 0.951 11        | 0.932 32 | 0.952 57 | 0.947 71 |
|    | $m=10$ | <b>0.960 08</b> | 0.959 27        | 0.957 82        | 0.936 2  | 0.958 52 | 0.954 06 |
| 5  | $m=7$  | <b>0.949 18</b> | 0.948 83        | 0.947 87        | 0.937 92 | 0.948 04 | 0.946 19 |
|    | $m=8$  | 0.956 92        | 0.956 63        | <b>0.956 93</b> | 0.947 62 | 0.956 53 | 0.955 61 |
|    | $m=9$  | <b>0.965 70</b> | 0.964 46        | 0.964 01        | 0.943 1  | 0.964 88 | 0.963 3  |
|    | $m=10$ | <b>0.970 56</b> | 0.969 09        | 0.968 78        | 0.948 45 | 0.970 01 | 0.969 17 |

表2(续)

Table 2 (continued)

| 图像 | 阈值数量   | FSIM            |                 |          |          |          |          |
|----|--------|-----------------|-----------------|----------|----------|----------|----------|
|    |        | IZOA            | ZOA             | WOA      | BWO      | GWO      | GJO      |
| 6  | $m=7$  | <b>0.945 00</b> | 0.943 89        | 0.939 91 | 0.929 38 | 0.943 88 | 0.937 4  |
|    | $m=8$  | 0.949 13        | <b>0.949 84</b> | 0.947 91 | 0.941 51 | 0.949 26 | 0.945 27 |
|    | $m=9$  | <b>0.959 83</b> | 0.956 52        | 0.953 08 | 0.937 37 | 0.956 36 | 0.953 14 |
|    | $m=10$ | <b>0.966 35</b> | 0.964 94        | 0.959 35 | 0.943 94 | 0.965 32 | 0.959 66 |
| 7  | $m=7$  | <b>0.936 27</b> | 0.935 73        | 0.935 7  | 0.924 98 | 0.935 64 | 0.935 98 |
|    | $m=8$  | <b>0.947 09</b> | 0.945 68        | 0.945 23 | 0.935 84 | 0.945 65 | 0.942 43 |
|    | $m=9$  | <b>0.954 92</b> | 0.953 02        | 0.951 83 | 0.927 74 | 0.953 17 | 0.950 49 |
|    | $m=10$ | <b>0.961 04</b> | 0.958 1         | 0.956 62 | 0.929 74 | 0.958 41 | 0.956 75 |
| 8  | $m=7$  | <b>0.920 27</b> | 0.917 67        | 0.915 27 | 0.903 33 | 0.914 76 | 0.908 37 |
|    | $m=8$  | <b>0.927 67</b> | 0.926 14        | 0.922 62 | 0.918 49 | 0.923 92 | 0.923 1  |
|    | $m=9$  | <b>0.940 18</b> | 0.937 51        | 0.933 5  | 0.916 82 | 0.938 59 | 0.932 35 |
|    | $m=10$ | <b>0.949 5</b>  | 0.945 32        | 0.944 93 | 0.919 87 | 0.945 9  | 0.941 44 |
| 9  | $m=7$  | <b>0.933 01</b> | 0.930 98        | 0.929 69 | 0.924 58 | 0.931 62 | 0.930 64 |
|    | $m=8$  | <b>0.944 93</b> | 0.941 22        | 0.941 49 | 0.935 22 | 0.943 47 | 0.939 9  |
|    | $m=9$  | <b>0.952 44</b> | 0.949 58        | 0.947 41 | 0.928 54 | 0.950 71 | 0.947 48 |
|    | $m=10$ | <b>0.957 16</b> | 0.955 26        | 0.954 77 | 0.935 58 | 0.954 54 | 0.953 88 |

表3 不同算法不同阈值个数的SSIM截断均值对比

Table 3 Comparison of SSIM truncated means for different numbers of thresholds in various algorithms

| 图像 | 阈值数量   | SSIM            |                 |          |          |                 |          |
|----|--------|-----------------|-----------------|----------|----------|-----------------|----------|
|    |        | IZOA            | ZOA             | WOA      | BWO      | GWO             | GJO      |
| 1  | $m=7$  | <b>0.945 24</b> | 0.944 63        | 0.943 95 | 0.937 01 | 0.944 28        | 0.941 87 |
|    | $m=8$  | <b>0.955 17</b> | 0.954 86        | 0.954 27 | 0.946 27 | 0.955 03        | 0.951 06 |
|    | $m=9$  | 0.962 49        | 0.961 77        | 0.960 33 | 0.943 22 | <b>0.962 56</b> | 0.959 39 |
|    | $m=10$ | <b>0.968 22</b> | 0.967 13        | 0.965 23 | 0.948 66 | 0.967 69        | 0.963 54 |
| 2  | $m=7$  | 0.912 39        | <b>0.912 63</b> | 0.911 01 | 0.901 78 | 0.911 66        | 0.906 2  |
|    | $m=8$  | <b>0.926 62</b> | 0.926 41        | 0.925 24 | 0.917 88 | 0.924 38        | 0.920 92 |
|    | $m=9$  | <b>0.938 3</b>  | 0.936 86        | 0.936 71 | 0.913 32 | 0.936 79        | 0.930 85 |
|    | $m=10$ | <b>0.943 61</b> | 0.943 27        | 0.942 05 | 0.922 48 | 0.943 21        | 0.938 45 |
| 3  | $m=7$  | <b>0.931 43</b> | 0.930 28        | 0.928 45 | 0.927 45 | 0.929 52        | 0.926 51 |
|    | $m=8$  | <b>0.941 49</b> | 0.940 67        | 0.938 28 | 0.933 67 | 0.939 38        | 0.936 61 |
|    | $m=9$  | <b>0.947 85</b> | 0.946 77        | 0.945 17 | 0.934 13 | 0.945 75        | 0.943 65 |
|    | $m=10$ | <b>0.952 94</b> | 0.951 29        | 0.949 89 | 0.938 95 | 0.951 7         | 0.948 86 |
| 4  | $m=7$  | 0.939 14        | <b>0.940 62</b> | 0.938 73 | 0.933 85 | 0.938 4         | 0.936 2  |
|    | $m=8$  | <b>0.949 62</b> | 0.949 3         | 0.945 08 | 0.943 18 | 0.947 89        | 0.941 92 |
|    | $m=9$  | <b>0.956 58</b> | 0.956 36        | 0.954 19 | 0.943 29 | 0.955 94        | 0.951 63 |
|    | $m=10$ | <b>0.962 6</b>  | 0.962 28        | 0.959 65 | 0.946 89 | 0.960 1         | 0.956 49 |
| 5  | $m=7$  | <b>0.935 06</b> | 0.934 3         | 0.933 93 | 0.929 99 | 0.934 19        | 0.932 21 |
|    | $m=8$  | <b>0.943 16</b> | 0.943 11        | 0.942 81 | 0.938 36 | 0.942 47        | 0.941 74 |
|    | $m=9$  | <b>0.949 27</b> | 0.948 74        | 0.949 19 | 0.939 19 | 0.948 22        | 0.949 22 |
|    | $m=10$ | <b>0.955 11</b> | 0.954 63        | 0.954 62 | 0.944 1  | 0.954 12        | 0.953 45 |

表 3(续)

Table 3 (continued)

| 图像 | 阈值数量   | SSIM            |          |          |          |                 |          |
|----|--------|-----------------|----------|----------|----------|-----------------|----------|
|    |        | IZOA            | ZOA      | WOA      | BWO      | GWO             | GJO      |
| 6  | $m=7$  | <b>0.929 94</b> | 0.928 51 | 0.923 43 | 0.911 29 | 0.928 3         | 0.918 99 |
|    | $m=8$  | 0.930 99        | 0.933 85 | 0.930 27 | 0.927 12 | <b>0.934 96</b> | 0.927 57 |
|    | $m=9$  | <b>0.943 94</b> | 0.939 45 | 0.934 14 | 0.924 1  | 0.938 45        | 0.936 75 |
|    | $m=10$ | <b>0.952 54</b> | 0.951 3  | 0.943 15 | 0.931 46 | 0.951 08        | 0.944 44 |
| 7  | $m=7$  | <b>0.941 3</b>  | 0.942 08 | 0.943 14 | 0.938 77 | 0.942 14        | 0.941 94 |
|    | $m=8$  | <b>0.951 95</b> | 0.951 76 | 0.951 03 | 0.946 95 | 0.953 3         | 0.949 5  |
|    | $m=9$  | <b>0.957 62</b> | 0.957 74 | 0.958 38 | 0.946 25 | 0.957 74        | 0.957 04 |
|    | $m=10$ | <b>0.962 75</b> | 0.961 58 | 0.962 98 | 0.949 81 | 0.963 55        | 0.961 99 |
| 8  | $m=7$  | <b>0.919 16</b> | 0.917 85 | 0.916 51 | 0.907 68 | 0.916 08        | 0.912 23 |
|    | $m=8$  | <b>0.929 29</b> | 0.928 72 | 0.926 8  | 0.920 96 | 0.927 65        | 0.926 33 |
|    | $m=9$  | <b>0.940 71</b> | 0.938 83 | 0.935 68 | 0.920 41 | 0.939 12        | 0.934 37 |
|    | $m=10$ | <b>0.948 65</b> | 0.946 08 | 0.945 01 | 0.924 04 | 0.945 94        | 0.943 02 |
| 9  | $m=7$  | <b>0.937 61</b> | 0.936 65 | 0.934 22 | 0.930 12 | 0.936 66        | 0.934 8  |
|    | $m=8$  | <b>0.948 79</b> | 0.946 87 | 0.945 68 | 0.941 64 | 0.947           | 0.944 01 |
|    | $m=9$  | <b>0.955 41</b> | 0.954 03 | 0.951 84 | 0.936 81 | 0.953 85        | 0.951 06 |
|    | $m=10$ | <b>0.961 29</b> | 0.960 01 | 0.958 28 | 0.944 84 | 0.958 61        | 0.957 27 |

表 4 不同算法不同阈值个数的 PSNR 截断均值对比

Table 4 Comparison of PSNR truncated means for different numbers of thresholds in various algorithms

| 图像 | 阈值数量   | PSNR            |          |          |          |          |          |
|----|--------|-----------------|----------|----------|----------|----------|----------|
|    |        | IZOA            | ZOA      | WOA      | BWO      | GWO      | GJO      |
| 1  | $m=7$  | <b>25.466 9</b> | 25.362 6 | 25.250 8 | 24.692 6 | 25.322 2 | 25.138 1 |
|    | $m=8$  | <b>26.349 7</b> | 26.239 1 | 26.043 3 | 25.47    | 26.251   | 25.915 9 |
|    | $m=9$  | <b>27.233 9</b> | 27.055   | 26.793 3 | 25.230 7 | 27.136 5 | 26.716 2 |
|    | $m=10$ | <b>28.009 5</b> | 27.798 2 | 27.511 1 | 25.801 2 | 27.805 9 | 27.268 2 |
| 2  | $m=7$  | <b>23.616 1</b> | 23.583 1 | 23.422 7 | 22.909 2 | 23.436 4 | 23.218 7 |
|    | $m=8$  | <b>24.613 7</b> | 24.595 1 | 24.427 4 | 23.741 5 | 24.341 4 | 24.106 1 |
|    | $m=9$  | <b>25.476 6</b> | 25.422   | 25.238 9 | 23.557   | 25.251 3 | 24.880 3 |
|    | $m=10$ | <b>26.269 7</b> | 26.167 3 | 25.923 2 | 24.020 6 | 26.029 9 | 25.479   |
| 3  | $m=7$  | <b>25.404 5</b> | 25.294 9 | 25.095 9 | 24.660 3 | 25.326 3 | 25.107 3 |
|    | $m=8$  | <b>26.440 3</b> | 26.393 7 | 26.064 4 | 25.318 6 | 26.179 7 | 26.018 3 |
|    | $m=9$  | <b>27.265 4</b> | 27.083 3 | 26.725 2 | 25.351 4 | 26.924 2 | 26.646 7 |
|    | $m=10$ | <b>28.062 3</b> | 27.784   | 27.563 7 | 25.766 7 | 27.831 4 | 27.342 3 |
| 4  | $m=7$  | <b>24.791 4</b> | 24.758 4 | 24.552 2 | 24.071 6 | 24.636 8 | 24.424 8 |
|    | $m=8$  | <b>25.773 1</b> | 25.663 5 | 25.435 9 | 25.052 8 | 25.565 3 | 25.164 1 |
|    | $m=9$  | <b>26.736 8</b> | 26.549 1 | 26.441 9 | 24.871 3 | 26.500 5 | 26.186 8 |
|    | $m=10$ | <b>27.537 5</b> | 27.439 7 | 27.070 7 | 25.444 1 | 27.179 7 | 26.752 2 |
| 5  | $m=7$  | <b>24.665</b>   | 24.575 9 | 24.473 1 | 23.921 8 | 24.537 8 | 24.355 4 |
|    | $m=8$  | <b>25.645 3</b> | 25.504 5 | 25.432 9 | 24.785 4 | 25.519 8 | 25.329 1 |
|    | $m=9$  | <b>26.491 3</b> | 26.326   | 26.196 5 | 24.551 3 | 26.308 3 | 26.053 7 |
|    | $m=10$ | <b>27.228 4</b> | 26.989 5 | 26.822 4 | 24.979 3 | 27.065 7 | 26.729 5 |

表4(续)  
Table 4 (continued)

| 图像 | 阈值数量   | PSNR            |          |          |          |          |          |
|----|--------|-----------------|----------|----------|----------|----------|----------|
|    |        | IZOA            | ZOA      | WOA      | BWO      | GWO      | GJO      |
| 6  | $m=7$  | <b>25.028 7</b> | 24.874 1 | 24.716 4 | 24.118 9 | 24.870 4 | 24.665 9 |
|    | $m=8$  | <b>25.882 8</b> | 25.646 3 | 25.483 9 | 24.860 6 | 25.625 7 | 25.409 2 |
|    | $m=9$  | <b>26.789 8</b> | 26.552 1 | 26.340 2 | 24.612 4 | 26.518 5 | 26.092 9 |
|    | $m=10$ | <b>27.523 2</b> | 27.318 2 | 27.038 3 | 25.341 4 | 27.277 3 | 26.862 3 |
| 7  | $m=7$  | <b>24.592 3</b> | 24.531 7 | 24.584   | 23.859 8 | 24.465 5 | 24.424 4 |
|    | $m=8$  | <b>25.813 7</b> | 25.645   | 25.441 6 | 24.914 1 | 25.652 1 | 25.010 9 |
|    | $m=9$  | <b>26.658 6</b> | 26.556 6 | 26.380 2 | 24.484 1 | 26.447 7 | 26.001 8 |
|    | $m=10$ | <b>27.518 9</b> | 27.338 1 | 27.187 1 | 24.928   | 27.286 5 | 26.730 4 |
| 8  | $m=7$  | <b>24.984 4</b> | 24.846 7 | 24.72    | 24.022 9 | 24.775   | 24.546 7 |
|    | $m=8$  | <b>25.844 8</b> | 25.699 7 | 25.564 9 | 24.888 1 | 25.657   | 25.546 4 |
|    | $m=9$  | <b>26.784 8</b> | 26.522 4 | 26.295 9 | 24.697 8 | 26.589 9 | 26.209 6 |
|    | $m=10$ | <b>27.564 7</b> | 27.210 2 | 27.120 4 | 25.130 9 | 27.269   | 26.977 7 |
| 9  | $m=7$  | <b>24.195 3</b> | 23.954   | 23.888 7 | 23.257 4 | 23.982 2 | 23.792   |
|    | $m=8$  | <b>25.308 9</b> | 25.099 9 | 24.941 3 | 24.144 9 | 25.142 5 | 24.769 5 |
|    | $m=9$  | <b>26.328 2</b> | 25.918 3 | 25.718 2 | 23.722 9 | 26.025   | 25.463 4 |
|    | $m=10$ | <b>27.126 5</b> | 26.839 1 | 26.732 6 | 24.341 8 | 26.782 2 | 26.443 5 |

#### 4.5 IZOA 分割效率分析

本文将各个算法在对图像 1 进行寻优过程中的最优适应度绘制成如图 5 所示的曲线对比图直观地展示了 IZOA 的优异寻优效率。

从图 5 可以看出,相较于其他算法 IZOA 在不同的分割阈值数量下都能拥有更快的收敛优势,说明 IZOA 分割效率的优越性。同时在收敛过程中 IZOA 也拥有多个拐点,说明在这个过程中 IZOA 多次跳出局部最优从而获取更好的适应度。

#### 4.6 IZOA 分割稳定性分析

为了减少偶然因素的影响,在消融实验与对比实验中,每个算法均独立运行 30 次。为直观展示算法的稳定性,本文绘制了不同算法在每张图像上每次分割所获得的 3 个指标的标准差,随着分割阈值数量的增加而变化的情况进行对比。如图 6 所示,可以看出,在不同的图像中,IZOA 绝大部分情况都是靠最底下的位置,这说明算法在分割阈值寻优过程中的稳定性表现优于其他的对比算法。

#### 4.7 IZOA 分割数据分析

表 2 是不同算法对图 3 中的 9 张图像分割之后的特征相似度 FSIM 指标对比,其中加粗的数据表示最优值。由表 2 可知 IZOA 的最优值占比达到 91.7%,阈值  $m$  分别为 7~10 时,最优值占比为:88.9%、77.8%、100%、100%,说明随着阈值的增大,IZOA 跟其他 5 种算法分割相比表现依旧稳定。说明 IZOA 进行优化的分割图像的相似性和质量均高于另外 5 种算法,且随着分割阈值个数的增加不会影响其优势。

由表 3 可知,IZOA 分割图像的结构相似性指数 SSIM 最优值占比为 88.9%,并且在阈值个数  $m$  为 7~10 时,最优值占比为:77.8%、88.9%、88.9%、100%,根据 SSIM 指标含义,说明使用 IZOA 算法优化的阈值分割得到的图像质量优于另外 5 种优化算法。图像的亮度、对比度和结构信息都得到了更好的保留,随着阈值数量的增加,IZOA 表现越好。

由表 4 可知,由 IZOA 算法优化的阈值分割图像的峰值信噪比 PSNR 最优值占比为 100%,表明 IZOA 算法相比其他 5 种算法,在求解精度、分割效果上具有一定的优势。在阈值个数为 10 时,相比 ZOA、GJO、WOA、BWO、GWO 算法分别平均提高了 0.87%、1.70%、9.51%、1%、2.5%,说明相对于这 5 个算法,IZOA 拥有精度更高的分割效果,对原图信息的保留度有着优越的提升。当阈值  $m$  个数为 10 时,3 个指标中 IZOA 均占比 100%,说明在多阈值下的图像分割中 IZOA 非常有优势。

#### 4.8 IZOA 消融实验对比

为了验证本文的改进策略的有效性,将使用混沌映射的 IZOA1、增加邻域波动策略的 IZOA2、使用杂交与变异策略的 IZOA3 以及使用了精英保留策略的 IZOA4 与原本的 ZOA 以及最终的 IZOA 均使用最小交叉熵作为适应度函数对图 1 中的第一张图进行 10 阈值分割,每个算法分别独立运行 30 次,取截断平均值,获得如表 5 的数据。由表 5 可以看出,IZOA1 的值均高于 ZOA,说明使用混沌映射能够有效提高 ZOA 的图像分割能力。另外 IZOA2、IZOA3 以及 IZOA4 相对于 ZOA 在 3 个指标上至少两个

指标有所提升,说明这些改进策略对于图像的分割均发挥了作用。多策略改进的 IZOA 相对于 ZOA 在 3 个指标上提高最大。上述实验验证了各改进策略的有效性以及 IZOA 的优越性。

表 5 消融实验

Table 5 Experiment on ablation

| 算法    | FSIM      | SSIM      | PSNR     |
|-------|-----------|-----------|----------|
| ZOA   | 0.974 453 | 0.967 247 | 27.798 1 |
| IZOA1 | 0.974 476 | 0.967 782 | 27.799 3 |
| IZOA2 | 0.974 913 | 0.967 103 | 27.799 4 |
| IZOA3 | 0.975 458 | 0.968 115 | 28.008 3 |
| IZOA4 | 0.974 950 | 0.967 235 | 27.799 4 |
| IZOA  | 0.976 123 | 0.968 322 | 28.013 5 |

## 5 结 论

图像分割在现代图像处理中是一个非常重要的预处理步骤。传统的图像多阈值分割方法通常面临计算量大、复杂度高、分割精度低等问题。为了解决这些问题,本文提出了一种基于改进斑马优化算法(IZOA)结合最小对称交叉熵的多阈值图像分割算法。首先,在算法的初始化阶段引入了 Logistic 混沌映射,旨在增强种群的遍历性和多样性。混沌映射能够在搜索空间内生成具有良好分布特性的初始种群,从而提升全局搜索能力。其次,本文引入了邻域波动策略,这一策略帮助算法在全局范围内进行更有效的搜索,增加了找到全局最优解的可能性。此外本文引入了杂交和变异策略,从而提升整体搜索性能。另外,加入上述策略时进行精英保留,将最优值保存下来,防止在寻优过程中丢失最优解。为了评估分割效果,本文对改进算法在分割后的 FSIM、SSIM 和 PSNR 图像质量指标进行了详细的分析,并与其他分割算法进行了对比。结果表明,本文提出的基于 IZOA 的最小对称交叉熵的多阈值分割方法在 FSIM、SSIM、PSNR 等指标上均优于其他 5 种对比算法。并且随着分割阈值个数的增加,本文算法仍然表现出明显的优势,说明该算法在处理复杂多阈值分割问题时具备较高的稳定性和优越性。

## 参考文献

- [1] 朱良宽,刘亮,东旭,等. 基于改进三维 Otsu 法的森林冠层图像分割[J]. 计算机工程, 2019, 45(1): 253-258,263.  
ZHU L K, LIU L, DONG X, et al. Forest canopy image segmentation based on improved 3D Otsu method[J] Computer Engineering, 2019, 45(1):253-258,263.
- [2] 陈鹏翔,杨晟院,区域拟合的背景去除图像分割模型[J]. 中国图象图形学报,2016,21(6):683-690.

CHEN P X, YANG SH Y. Background removal image segmentation model based on region fitting[J]. Journal of Image and Graphics, 2016, 21(6): 683-690.

- [3] 李洁,于艺铭,陈茜,等. 基于分通道自适应理论的阈值分割算法及评价[J]. 南京理工大学学报,2022, 46(5): 606-614, 631.  
LI J, YU Y M, CHEN X, et al. Threshold segmentation algorithm and evaluation based on split-channel adaptive theory [J]. Journal of Nanjing University of Science and Technology, 2022, 46(5): 606-614, 631.
- [4] 祁颖,柴艳妹. 基于改进的 SLIC 和聚类算法结合的高分辨率遥感海冰图像分割[J]. 计算机科学, 2022, 49(S2): 591-596.  
QI Y, CHAI Y M. Segmentation of high-resolution remotely sensed sea ice images based on the combination of improved SLIC and clustering algorithms[J]. Computer Science, 2022,49(S2): 591-596.
- [5] 李忠成,解滨,张文祥. 基于多元分层感知的移动物联网区域分割算法[J]. 计算机工程, 2021, 47(9): 51-58.  
LI ZH CH, XIE B, ZHANG W X. Mobile IoT region segmentation algorithm based on multivariate hierarchical sensing[J]. Computer Engineering, 2021, 47(9): 51-58.
- [6] RAMESH K K D, KUMAR G K, SWAPNA K, et al. A review of medical image segmentation algorithms[J]. EAI Endorsed Transactions on Pervasive Health and Technology, 2021, 7(27):169-184.
- [7] 张相芬,刘艳,袁非牛. 基于倒金字塔深度神经网络的三维医学图像分割[J]. 计算机工程, 2022, 48(12): 304-311.  
ZHANG X F, LIU Y, YUAN F N. 3D medical image segmentation based on inverted pyramid deep learning network[J]. Computer Engineering, 2022, 48(12): 304-311.
- [8] URAJLIC E, BUZA E, AKAGIC A. Multilevel image thresholding based on Rao algorithms and Kapur's entropy [C]. International Conference on Information, Communication and Automation Technologies. Washington D. C., USA: IEEE Press, 2022: 1-7.
- [9] HAN B, WU Y Q. A novel active contour model based on modified symmetric cross entropy for remote sensing river image segmentation [J]. Pattern Recognition, 2017, 67: 396-409.

- [10] 李康乐,张云逸,韩劲松,等.基于WOA优化概率分布参考点的锂电池故障诊断[J].计算机应用研究,2023,40(12):3551-3558.  
LI K L,ZHANG Y Y,HAN J S,et al.Lithium battery fault diagnosis based on WOA-Optimized probability distribution reference points[J].Application Research of Computers,2023,40(12):3551-3558.
- [11] 潘峰,孙红霞.基于蝗虫算法的图像多阈值分割方法[J].电子测量与仪器学报,2019,33(1):149-155.  
PAN F, SUN H X. Image multi-threshold segmentation method based on grasshopper optimization algorithm[J].Journal of Electronic Measurement and Instrumentation,2019,33(1):149-155.
- [12] ERDMANN H, WACHS-LOPE G, GALLAO C, et al. A study of a firefly meta-heuristics for multi-threshold image segmentation[C]. Developments in Medical Image Processing and Computational Vision. Berlin: Springer, 2015: 279-295.
- [13] WANG S K, JIA H M, PENG X X. Modified salp swarm algorithm based multilevel thresholding for color image segmentation [J]. Mathematical Biosciences and Engineering,2020,17(1):700-724.
- [14] 吕鑫,慕晓冬,张钧,等.混沌麻雀搜索优化算法[J].北京航空航天大学学报,2021,47(8):1712-1720.  
LYU X, MU X D, ZHANG J, et al. Chaos sparrow search optimization algorithm[J]. Journal of Beijing University of Aeronautics and Astronautics, 2021, 47(8): 1712-1720.
- [15] YAO X, LI Z, LIU L, et al. Multi-threshold image segmentation based on improved grey wolf optimization algorithm [J]. IOP Conference Series: Earth and Environmental Science, 2019, 252(4): 042105.
- [16] ZHAO P J, LIU S Y. An improved symbiotic organisms search algorithm with good point set and memory mechanism [J]. The Journal of Supercomputing, 2023, 79(10): 11170-11197.
- [17] TROJOVSKA E, DEGHANI M, TROJOVSKY P. Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm [J]. IEEE Access, 2022, 10 (9): 49445-49473.
- [18] XU B R, YE X M, WANG G Y, et al. A fractional-order improved quantum logistic map: chaos, 0-1 testing, complexity, and control[J]. Axioms,2023,12(1):1-15.
- [19] 郭喜峰,黄裕海,单丹,等.改进全局ZOA优化MVMD-SCN的锂电池SOH估算[J].电子测量技术,2024,14(5):22-30.  
GUO X F, HUANG Y H, SHAN D, et al. Improved global ZOA optimization of MVMD-SCN for lithium battery SOH estimation[J]. Electronic Measurement Technology, 2024, 14(5): 22-30.
- [20] LIU M, YAO X, LI Y. Hybrid whale optimization algorithm enhanced with Lévy flight and differential evolution for job shop scheduling problems [J]. Applied Soft Computing, 2020, 87:105954.
- [21] WANG W Q, MA M, QIAN Y B, et al. Optimization of child protection system based on improved particle swarm optimization algorithm[J]. China Safety Science Journal, 2023, 33(1): 122-129.

### 作者简介

刘庭亭,硕士研究生,主要研究方向为图像处理。

E-mail: gs.tingtingliu23@gzu.edu.cn

何志琴(通信作者),教授,硕士生导师,主要研究方向为计算机控制。

E-mail: zqhe@gzu.edu.cn