2023年2月
第42卷 第2期

DOI:10.19652/j.cnki.femt.2204525

基于蛇优化算法的Otsu图像分割方法*

李圣涵　叶琳琳
(海南大学儋州 571700)

摘　要:Otsu算法是图像处理中运用广泛的图像分割方法,尽管有着计算简单、准确的特性,但因为需要进行穷举运算,所以计算效率不高。为提高图像分割的实时性,引入了蛇优化算法(SO)对Otsu进行了优化,创建了基于蛇优化算法的Otsu图像分割方法(SO-Otsu)。在该算法中,利用蛇优化算法来模拟蛇的特性进行最佳阈值的寻找,以降低迭代时间,提升计算速度。在仿真实验中,利用经典的Lena、Peppers、Goldhill、Cameraman图片进行测试,与基于果蝇优化算法的Otsu方法(FOA-Otsu)和基于麻雀搜索算法的Otsu方法(SSA-Otsu)进行对比。并通过计算峰值信噪比(PSNR)、结构相似性(SSIM)、特征相似性(FSIM)和计算时间作为评价指标结果进行评估。结果表明,与其他算法相比,算法计算效率高、分割细节效果好且综合分割性能优异,为提高图像分割的计算效率提供了一种理想的工具。

关键词:Otsu;蛇优化算法;图像分割
中图分类号:TP2　　**文献标识码**:A　　**国家标准学科分类代码**:510.1050

Otsu image segmentation method based on snake optimization algorithm

Li Shenghan　Ye Linlin
(Hainan University, Danzhou 571700, China)

Abstract: Otsu algorithm is a widely used image segmentation method in image processing, although it has the characteristics of simple calculation and accuracy, but because of the need for exhaustive operation, the calculation efficiency is not high. In order to improve the real-time performance of image segmentation, snake optimizer (SO) is introduced to optimize Otsu, and an Otsu image segmentation method (SO-Otsu) based on snake optimization algorithm is created. In this algorithm, snake optimization algorithm is used to simulate the characteristics of snakes to find the optimal threshold, so as to reduce the iteration time and improve the calculation speed. In the simulation experiment, classic Lena, Peppers, Goldhill and Cameraman images were used for testing, Otsu method based on fruit fly optimization algorithm (FOA-Otsu) and Otsu method based on sparrow search algorithm (SSA-Otsu) were compared. PSNR, SSIM, FSIM and calculation time were calculated as evaluation indicators. The results show that compared with other algorithms, the proposed algorithm has high computational efficiency, good detail segmentation effect and excellent comprehensive segmentation performance, which provides an ideal tool for improving the computational efficiency of image segmentation.

Keywords: Otsu; snake optimizer; image segmentation

0　引　言

图像分割是计算机视觉技术和模式识别的关键任务之一,是许多图像处理任务的预处理步骤,也是视觉图像如目标监测、特征提取、目标识别等相关识别操作和理解的基础。

在图像分割中,灰度或彩色图像被划分为多个非重叠段,使得每个段是均匀的,而两个或多个任意相邻段是异

收稿日期:2022-11-22
* **基金项目**:海南省自然科学基金(120QN178)、海南大学科研基金(KYQD(ZR)-21082)项目资助

质的。图像分割已广泛应用于医学图像分析[1]、工件无损分析[2]、雷达图像分析[3]、地质勘探[4]等领域，分割的好坏直接影响后续对图像的理解。因此，图像分割一直是图像处理领域的研究热点之一[5]。

传统的图像分割方法主要有基于阈值的分割方法、基于区域的分割方法、基于边缘的分割方法和基于灰度直方图的阈值分割方法[6]。在众多的图像分割方法中，最简单的一种是在图像的目标和背景具有明显的灰度级对比度和颜色变化的情况下，利用前景和背景的灰度级变化通过对比、边缘和灰度级检测方法进行分割。但在现实处理过程中，由于光照等条件的影响，图像的对比度特征可能不是很明显，因此许多算法存在着局限性[7]。

而基于灰度直方图的阈值分割方法，将利用整个图像的灰度特征，从而大大提高图像分割的性能。其中日本学者 Otsu[8]提出的 Otsu 方法是一种计算简单、准确的分割方法，该方法利用图像的灰度值计算前景与背景之间的最大方差，选出合适的阈值完成前景与背景的分割，显著提升了图像的分割效果。但它也存在计算量大的缺点，在计算过程中需要遍历所有的灰度值来计算方差，假设计算每个方差的时间为 T，那么总的计算时间就为 256 T，这大大降低了图像分割的计算效率。

为了提高分割的计算速度，许多国内外学者都通过结合群体智能优化算法的方法，提出了 Otsu 方法的改进版本。Huang 等[9]提出了基于果蝇优化算法的 Otsu 图像分割方法，通过模拟果蝇通过嗅觉寻找食物的方法，来寻找最佳阈值。李鹏等[10]也提出了基于麻雀优化算法的 Otsu 分割方法，模拟麻雀的觅食和反捕食行为完成最佳阈值的搜索。Houssein 等[11]通过改良的金豺狼算法优化了 Otsu，并将其用于医学领域的皮肤癌图像分割中，取得了较好的成果。Horng 等[12]也将人工蜂群算法用于图像分割的阈值计算中。孙振营等[13]结合了缎蓝园丁鸟算法优化了多阈值 Otsu 方法，并将其应用于遥感图像的分割上，具有抗噪性好，计算速度快的优点。吴京城等[14]提出了反背景差分结合 Otsu 的细胞分割方法，能准确、客观自动分析细胞汇和度[14]。

本文提出了一种结合蛇优化(snake optimizer，SO)的 Otsu 算法。SO 算法是 Hashim 等[15]于 2022 年提出的一种新的元启发式优化算法，通过模拟蛇的行为模式完成寻优过程。

1 算法原理和方法

1.1 Otsu 方法

Otsu 方法是一种全局自适应二值化阈值图像分割算法，该算法以背景和目标图像之间的最大类间方差作为阈值选择规则，完全基于图像的直方图进行计算[16]。

令 $\{0,1,2,\cdots,L-1\}$ 表示一幅大小为 $M\times N$ 像素的数字图像中的 L 个不同的灰度级，n_i 表示灰度级为 i 的像素数。图像中的像素总数 MN 为 $MN=n_0+n_1+n_2+\cdots+n_{L-1}$。归一化的直方图具有分量 $p_i=n_i/MN$，由此有：

$$\sum_{i=0}^{L-1} p_i = 1, \quad p_i \geqslant 0 \tag{1}$$

假设选取阈值 $T(k)=k$，$0<k<L-1$，并使用它把输入图像阈值化处理为两类 C_1 和 C_2，其中，C_1 由图像中灰度值在范围 $[0,k]$ 内的所有像素组成，C_2 由灰度值在范围 $[k+1,L-1]$ 内的所有像素组成。用该阈值，像素被分到类 C_1 中和类 C_2 中的概率 $P_1(k)$ 和 $P_2(k)$ 分别为：

$$P_1(k) = \sum_{i=0}^{k} p_i \tag{2}$$

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k) \tag{3}$$

分配到类 C_1 的像素的平均灰度值 $m_1(k)$ 和分配到类 C_2 的像素平均灰度值为 $m_2(k)$ 分别为：

$$m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} i p_i \tag{4}$$

$$m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} i p_i \tag{5}$$

而整个图像的平均灰度（即全局均值）为：

$$m_G = \sum_{i=0}^{L-1} i p_i \tag{6}$$

σ_B^2 为类间方差，定义为：

$$\sigma_B^2 = P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2 \tag{7}$$

再次引人 k，最终结果为：

$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]} \tag{8}$$

从而最佳阈值是 k^*，其最大化 $\sigma_B^2(k)$：

$$\sigma_B^2(k^*) = \max_{0\leqslant k\leqslant L-1} \sigma_B^2(k) \tag{9}$$

1.2 SO 算法

1)算法介绍

SO 算法是一种新的元启发式优化算法，模拟了蛇在不同温度、不同食物条件下的不同行为模式，进行最优值的寻找。

SO 算法的灵感来源于蛇的交配模式。只有在温度较低且食物可用的条件下，蛇才会发生交配行为；否则蛇只会寻找食物或者吃现有的食物。基于此，将考虑蛇优化算法的搜索过程分为两个阶段，即勘探和开发。

勘探模式发生在食物不足的情况下，在该模式下蛇种群的每一个个体都会四处勘探寻找食物。

如果食物充足，蛇将处于开发模式。在温度较高的情况下，蛇会关注与吃可用的食物；而在温度下较低的情况下，就会发生交配过程。在交配过程中，蛇种群可能会发生战斗，雄性间会相互战斗以获得最佳的雌性，雌性亦然。待交配完成后，雌性就会产卵，并选择是否孵化来替代现种群中最差的雄性和雌性个体，如图 1 所示。

2)数学模型和算法

SO 算法的模型公式如下，其中参数定义如表 1 所示，

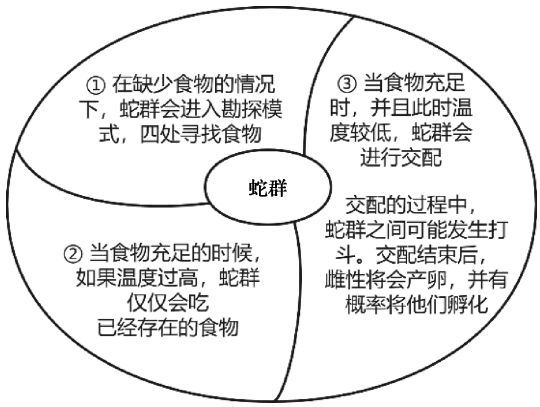

图 1 蛇的行为模式

算法流程如图 2 所示。

首先进行蛇种群的初始化：

$$X_i = X_{\min} + rand \times Limits \tag{10}$$

将蛇种群划分为雌性和雄性两个组，每组各占 50%：

$$N_m \approx N/2 \tag{11}$$

$$N_f = N - N_m \tag{12}$$

找到雄性、雌性和整个种群中最佳位置的个体 $X_{\text{best},m}$、$X_{\text{best},f}$ 和 X_{food}。

计算适应度 $f_{\text{best},m}$、$f_{\text{best},f}$ 和 f_{food}。

定义温度和食物数量：

$$Temp = \exp\left(\frac{-t}{T}\right) \tag{13}$$

表 1 SO 算法参数定义

符号	含义
$X_{i,m}$、$X_{i,f}$、X_i	分别表示雄性、雌性和整个种群中第 i 个个体的位置
$rand$	表示 [0,1] 范围内的随机数
$X_{\max}$ 和 $X_{\min}$	求解问题的上下边界
N、N_m 和 N_f	分别为蛇种群大小、雄性数目、雌性数目
$f_{\text{best},m}$、$f_{\text{best},f}$ 和 f_{food}	分别为雄性、雌性和整个群体中最好个体的适应度
$X_{\text{best},m}$、$X_{\text{best},f}$ 和 X_{food}	分别为雄性、雌性和整个群体中最好个体的位置
t、T	分别为当前迭代次数和最大迭代次数
c_1、c_2 和 c_3	都为常数，分别为 0.5、0.05 和 2
$X_{rand,m}$、$X_{rand,f}$	随机的雄性位置和雌性位置
$X_{\text{worst},m}$、$X_{\text{worst},f}$	分别表示最差的雄性和雌性个体的位置
$Limits$	求解问题上下边界的差值，即 $X_{\max} - X_{\min}$

$$Q = c_1 \cdot \exp\left(\frac{t-T}{T}\right) \tag{14}$$

如果 $Q<0.25$，蛇通过选择任何随机位置来搜索食物，并更新它们的位置。要模拟勘探阶段，如下所述：

$$X_{i,m} = X_{rand,m} \pm c_2 \times A_m \times (Limits \times rand + X_{\min}) \tag{15}$$

$$X_{i,f} = X_{rand,f} \pm c_2 \times A_f \times (Limits \times rand + X_{\min}) \tag{16}$$

A_m、A_f 为雄性寻找食物的能力，计算公式如下：

$$A_m = \exp\left(\frac{-f_{rand,m}}{f_{i,m}}\right) \tag{17}$$

$$A_f = \exp\left(\frac{-f_{rand,f}}{f_{i,f}}\right) \tag{18}$$

在 $Q>0.25$ 的条件下，如果 $temperature>0.6$，就说明温度处于热状态。蛇只会寻找食物，位置更新公式如下：

$$X_i = X_{food} \pm c_3 \times Temp \times rand \times (X_{food} - X_i) \tag{19}$$

在 $Q>0.25$ 的条件下，如果 $temperature<0.6$，就说明温度处于冷状态，蛇将处于战斗模式或交配模式。

战斗模式的公式如下：

$$X_{i,m} = X_{i,m} + c_3 \times FM \times rand \times (Q \times X_{\text{best},f} - X_{i,m}) \tag{20}$$

$$X_{i,f} = X_{i,f} + c_3 \times FF \times rand \times (Q \times X_{\text{best},m} - X_{i,f}) \tag{21}$$

FM 和 FF 表示雌性战斗能力，公式如下：

$$FM = \exp\left(\frac{-f_{\text{best},f}}{f_i}\right) \tag{22}$$

$$FF = \exp\left(\frac{-f_{\text{best},m}}{f_i}\right) \tag{23}$$

交配模式的公式如下：

$$X_{i,m} = X_{i,m} + c_3 \times M_m \times rand \times (Q \times X_{i,f} - X_{i,m}) \tag{24}$$

$$X_{i,f} = X_{i,f} + c_3 \times M_f \times rand \times (Q \times X_{i,m} - X_{i,f}) \tag{25}$$

M_m、M_f 分为雄性和雌性的交配能力，公式如下：

$$M_m = \exp\left(\frac{-f_{i,f}}{f_{i,m}}\right) \tag{26}$$

$$M_f = \exp\left(\frac{-f_{i,m}}{f_{i,f}}\right) \tag{27}$$

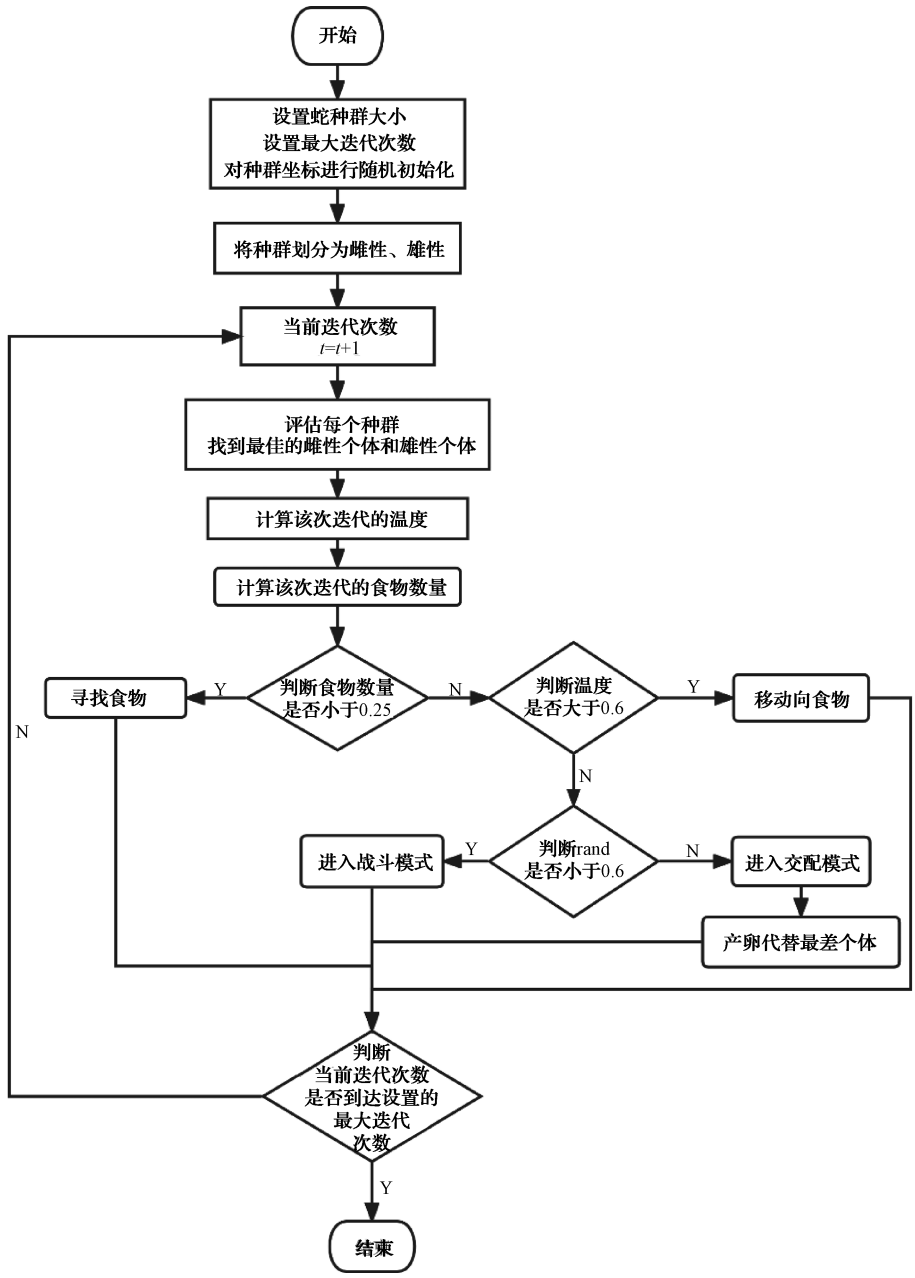

图 2　算法流程

交配完成后可以选择是否将卵孵化，如果选择孵化，就选择最差的雄性和雌性个体，并替换它们。

$$X_{\text{worst},m} = X_{\min} + rand \times (X_{\max} - X_{\min}) \tag{28}$$

$$X_{\text{worst},f} = X_{\min} + rand \times (X_{\max} - X_{\min}) \tag{29}$$

2　基于 SO 算法改进的 Otsu 图像分割方法

传统的 Otsu 在选取最佳阈值时采用的是穷举法，将所有灰度值对应的类间方差全部求出，再一一比较，选出类间方差最大值对应的灰度值。

为解决传统 Otsu 算法计算时间长的问题，本文将 SO 算法与 Otsu 分割方法结合，利用 SO 模拟蛇的行为模式来寻找最佳阈值，以降低算法的计算时间，形成了 SO-Otsu 算法。

解决方案是将 Otsu 计算过程中的灰度值 K 视为 SO 算法中的蛇种群的坐标 X，并根据式(8)计算每一个蛇个体的适应度，把适应度取反，利用 SO 算法模拟蛇行为模式在迭代中比较适应度，更新坐标 X，最终找到最佳阈值，以代替式(9)穷举法的效果。

由于公式设定的原因，计算所得到坐标的结果为浮点数。而图像的灰度值为处于[0，255]的离散整数，所以在计算个体适应度时，会对坐标进行处理，将其限定为整数，供后续计算。

在迭代过程中，如果该次迭代得到的最佳阈值与前3次的结果相同，就视为该优化过程已经收敛，可以提前结束迭代，以减少算法耗时。

SO-Otsu伪代码如算法1所示。

算法1 基于蛇优化算法优化的Otsu算法

1. 初始化参数：维度 $Dim=1$、寻优上下界$\in[1,256]$、迭代次数 $T=30$、当前迭代次数 $t=0$，蛇种群数目 $N=10$
2. 在上下界内随机初始化蛇群的坐标 根据式(10)
3. 将蛇群分为两类 根据式(11)、(12)
4. **While($t<T$) do**
5. 计算计算每个个体的适应度 fitness 根据式(8)
6. 评估雌性群体和雄性群体中每个个体的适应度，找到两类群体的最佳位置 X_{food}
7. 定义温度公式 T 和食物公式 Q 根据式(13)、(14)
8. **If($Q<0.25$) then**
9. 进入勘探模式，寻找食物，更新坐标 根据式(15)、(16)
10. **else**
11. **if $T>0.6$ then**
12. 进入勘探模式，移动向食物，更新坐标 根据式(19)
13. **else**
14. **if *rand*>0.6 then**
15. 进入战斗模式，更新坐标 根据式(20)、(21)
16. **else**
17. 进入交配模式，更新坐标 根据式(24)、(25)
18. 选择是否孵化，孵化后替换最差的雄性和雌性，更新坐标 根据式(28)、(29)
19. **end if**
20. **end if**
21. **end if**
22. **If($X_{food}==X_{food}[t-3]$)**
23. 结束迭代
24. **End while**

3 评价指标

为了验证分割效果的有效性，选取峰值信噪比(peak signal to noise ratio，PSNR)、特征相似性(feature similarity index measure，FSIM)和结构相似性(structure similarity index measure，SSIM)作为客观评价标准。

3.1 PSNR

PSNR是一种质量验证标准，可以用于评估原始图像和分割的图像之间的差异[17]，PSNR如下：

$$PSNR=20\lg\frac{255}{RMSE} \tag{30}$$

均方根误差(RMSE)如下：

$$RMSE=\sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}((I(i,j)-Seg(i,j))^2)}{M\times N}} \tag{31}$$

式中：I 是原始图像；而 Seg 是分割后大小为 $M\times N$ 的图像。

3.2 FSIM

FSIM是一种用于衡量原始图像与分割图像之间特征相似性的度量[18]。FSIM取决于两个主要标准，即相位一致性(PC)和梯度幅度(GM)。PC是用来简化功能检测过程，而GM用来计算图像梯度。

首先找到图像的 S_{PC}：

$$S_{PC}=\frac{2PC_1PC_2+T_1}{PC_1^2+PC_2^2+T_1} \tag{32}$$

式中：PC_1 和 PC_2 是原始图像和分割图像的相位一致性；T_1 用来提升 S_{PC} 的稳定性。

然后计算 S_G：

$$S_G=\frac{2G_1G_2+T_2}{G_1^2+G_2^2+T_2} \tag{33}$$

式中：G_1 表示原始图像的梯度而 G_2 表示第2幅图像的梯度；T_2 为一个正的常数。

图像的相似性 $S_L(x)$ 为：

$$S_L(x)=[S_{PC}(x)]^{\alpha}[S_G(x)]^{\beta} \tag{34}$$

式中：α 和 β 被用于调整 S_{PC} 和 S_G 的相似性。

3.3 SSIM

SSIM是一种用于评价原始图像和分割图像之间相似性的指标[19]，如下：

$$SSIM(I,Seg)=\frac{(2\mu_I\mu_{Seg}+c_1)(2\sigma_{I,Seg}+c_2)}{(\mu_I^2+\mu_{Seg}^2+c_1)(\sigma_I^1+\sigma_{Seg}^2+c_2)} \tag{35}$$

式中：μ_I 和 μ_{Seg} 分别为原始图像和分割图像的平均强度；σ_I 和 σ_{Seg} 是原始图像 I 和分割图像 Seg 的标准差；$\sigma_{I,Seg}$ 是两者的协方差；c_1 和 c_2 为常数。

4 仿真测试

4.1 主观评测

为了直观的对比本文算法在分割上的优势，对4幅标准图片分别利用Otsu、FOA-Otsu、SSA-Otsu和SO-Otsu算法进行分割。

4幅图像的分割结果如图3所示。从图3可以看出，SO-Otsu算法有较好的分割效果，在Lena中SO-Otsu算法相较于FOA-Otsu可以分割出帽子阴影的细节；在Cameraman中，3种算法的分割都较好；在Goldhill中SO-Otsu算法相较于FOA-Otsu更好的分割出了地面的细节，相较于SSA-Otsu更好的分割出了建筑和后面背景的细节；在Peppers中分割出来的细节相差不大。由此可

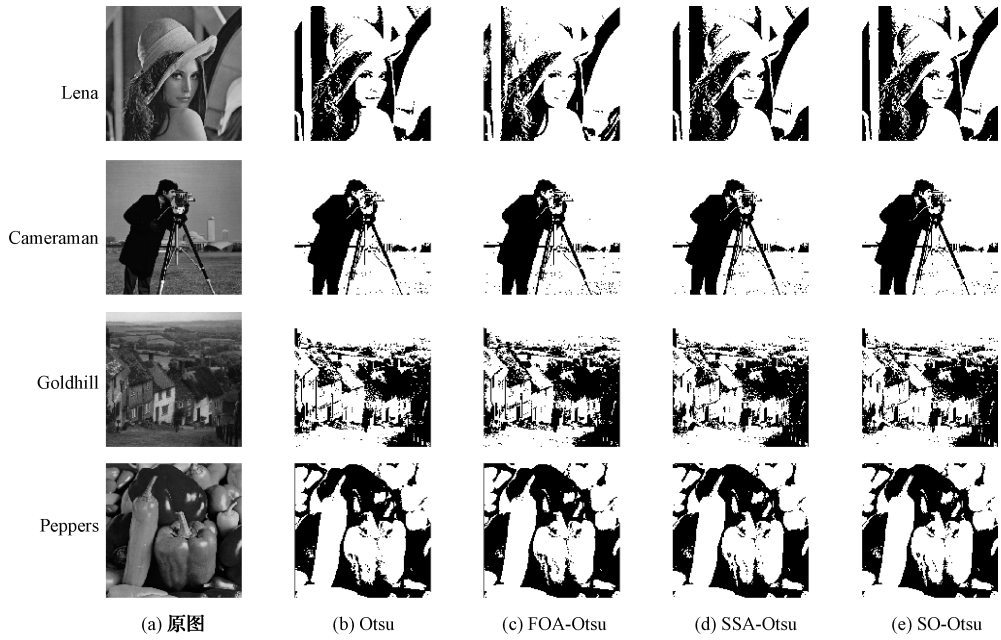

图 3　4 种算法分割效果对比

以说明，SO-Otsu 算法在分割图片细节方面较两种算法有优势。

4.2　客观评测

主观评测不能完全代表结果，本文通过计算 Otsu、FOA-Otsu、SSA-Otsu 和 SO-Otsu 的最佳阈值、迭代次数、PSNR、SSIM、FSIM 和运行时间作为客观评价指标，结果如表 2 所示。

表 2　4 种算法测试结果对比

图像	算法	最佳阈值	迭代次数	PSNR	SSIM	FSIM	时间/s
Lena	Otsu	940	2 048	5.702 5	0.995 6	0.513 9	0.151 4
	FOA-Otsu	800	5	5.708 2	0.988 9	0.495 0	0.055 9
	SSA-Otsu	936	3	5.702 8	0.995 3	0.513 3	0.086 4
	SO-Otsu	941	4	5.702 5	0.995 6	0.513 9	0.047 6
Cameraman	Otsu	707	2 048	5.636 9	0.994 8	0.555 4	0.144 8
	FOA-Otsu	736	6	5.632 6	0.998 4	0.488 4	0.061 2
	SSA-Otsu	713	3	5.636 9	0.994 8	0.555 4	0.081 6
	SO-Otsu	708	4	5.636 9	0.994 8	0.555 4	0.047 3
Goldhill	Otsu	1 045	2 048	6.395 5	0.999 2	0.450 7	0.167 2
	FOA-Otsu	1 057	3	6.395 0	0.999 0	0.450 7	0.056 2
	SSA-Otsu	1 043	4	6.396 0	0.999 4	0.450 8	0.084 5
	SO-Otsu	1 046	4	6.395 5	0.999 2	0.450 7	0.045 4
Peppers	Otsu	820	2 048	6.670 6	0.999 1	0.491 6	0.168 0
	FOA-Otsu	848	5	6.671 0	0.999 0	0.491 6	0.057 6
	SSA-Otsu	823	3	6.670 6	0.999 1	0.491 6	0.075 2
	SO-Otsu	824	4	6.670 6	0.999 1	0.491 6	0.047 1

算法的运行时间 t 与实时问题有关，是一个重要的客观标准。在相同条件下，相同算法的运行时间 t 不是固定的，而是波动的。因此，在相同的硬件配置下，相同的算法运行 100 次，并将其平均运行时间作为客观评价标准。

从表 2 可以看出，SO-Otsu 算法所得到的最佳阈值优于其他两种算法。3 种算法的迭代次数相差不大，基本在 5 次以内完成迭代。在 PSNR 与 SSIM 两个指标上面，3 种算法的差异保持在 1%以下；FSIM 指标中，使用 SO-Otsu 分割之后的 Lena 图像相较于 FOA-Otsu 算法提升了 3.8%，使用 SO-Otsu 分割之后的 Cameraman 图像相较于 FOA-Otsu 算法提升了 13.7%。从算法的运行时间

上分析，SO-Otsu算法相较于Otsu算法降低了70.3%的时间，相较于FOA-Otsu和SSA-Otsu算法降低了14.0%和42.3%的时间。

在利用FOA-Otsu算法计算阈值时，会出现计算出来的阈值相差过大的情况，如图4所示，分析原因可能是FOA-Otsu算法在计算时局限于局部最优解，而利用SO-Otsu算法寻找阈值时，基本不会出现这种状况，寻找的阈值较为稳定都在整体的最优解附近。

图4 根据局部最优解分割的效果

5 结 论

本文为了解决传统Otsu方法计算量大、效率低的缺点，将SO算法引入到Otsu中，提出了一种SO-Otsu图像分割方法。根据类之间的方差和类内的方差，结合SO最优搜索算法获得最优阈值，完成图像分割。该算法与传统的Otsu算法相比，极大减少了计算时间。引入PSNR、FSIM、SSIM和计算时间作为评价指标，与另外两种改进算法相比，SO-Otsu算法计算速度更快，综合分割性能更优异，主观上分割细节效果好，寻找的阈值较为稳定，不会局限于局部最优解，有较强的全局搜索能力。

根据本文的计算结果分析，预想将SO算法的计算优势应用在亟待解决时间优化问题的，其他阈值分割算法上，也将会有显著优化结果和广泛的应用前景。SO算法可以大大缩短计算时间的优势，尤其在未来升级改进的二维，三维运算的应用拓展上，其的优势将会更加显著。

参 考 文 献

[1] 邸拴虎，杨文瀚，廖苗，等. 基于RA-Unet的CT图像肝脏肿瘤分割[J]. 仪器仪表学报，2022，43(8)：65-72.

[2] 陈晓雨，张本松. 基于图像分割的机械工件表面缺陷无损识别方法[J]. 九江学院学报(自然科学版)，2021，36(1)：50-52，67.

[3] 邱佳月，赖际舟，李志敏，等. 面向复杂场景的激光雷达地面分割算法[J]. 仪器仪表学，2020，41(11)：244-251.

[4] 王鸣川，商晓飞，段太忠. 多点地质统计学建模中训练图像建立方法综述[J]. 高校地质学报，2022，28(1)：96-103.

[5] 吴毅强. 粒子群算法和活动轮廓波模型的图像分割研究[J]. 国外电子测量技术，2020，39(7)：47-51.

[6] 黄旭，张世义，李军. 图像分割技术研究综述[J]. 装备机械，2021(2)：6-9.

[7] 许新华，薛迪杰，齐国红. 基于区域差异性的激光热成像图像分割[J]. 激光杂志，2022，43(10)：83-86.

[8] OTSU N. A threshold selection method from gray-level histograms[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1979, 9(1)：62-663.

[9] HUANG C Y, LI X R, WEN Y L. An Otsu image segmentation based on fruit fly optimization algorithm[J]. Alexandria Engineering Journal, 2021, 60(1)：183-188.

[10] 李鹏，丁倩雯. 基于麻雀算法优化的OSTU分割算法[J]. 电子测量技术，2021，44(19)：148-154.

[11] HOUSSEIN E H, ABDELKAREEM D A, EMAM M M, et al. An efficient image segmentation method for skin cancer imaging using improved golden jackal optimization algorithm[J]. Computers in Biology and Medicine, 2022，149：106075-106075.

[12] HORNG M H. Multilevel thresholding selection based on the artificial bee colony algorithm for image1segmentation [J]. Expert Systems with Applications，2011, 38(11)：13785-13791.

[13] 孙振营，张文. 基于缎蓝园丁鸟算法优化的多阈值遥感图像分割[J]. 国外电子测量技术，2022，41(2)：33-39.

[14] 吴京城，洪欢欢，施露露，等. 反背景差分结合Otsu的细胞图像分割方法[J]. 电子测量与仪器学报，2021，35(4)：82-89.

[15] HASHIM F A, HUSSIEN A G. Snake optimizer：A novel meta-heuristic optimization algorithm [J]. Knowledge-Based Systems, 2022, DOI：10.1016/j.knosys.2022.108320.

[16] 胡道松，邱鹏程，周静，等. 基于改进Otsu算法的游标卡尺缝隙检测[J]. 机床与液压，2022，50(14)：8-13.

[17] 耿俊，李文海，吴子豪，等. 多分支图像去噪算法研究[J/OL]. 计算机工程与应用：1-14[2022-12-31]. http://kns.cnki.net/kcms/detail/11.2127.TP.

20220914.1726.010.html.

[18] 孙彦景,刘东林,谢新新, 等.基于人类视觉系统的特征相似性图像质量评价[J].华南理工大学学报(自然科学版),2017,45(3):11-19.

[19] 卜丽静,王涛.基于HVS的SSIM超分辨率重建图像质量评价方法[J].测绘与空间地理信息,2019,42(7):14-18,21.

作 者 简 介

李圣涵,本科,主要研究方向为深度学习和图像处理。
E-mail:2603204342@qq.com

叶琳琳(通信作者),博士,讲师,主要研究方向为生物医学光电成像及图像处理。
E-mail:994097@hainanu.edu.cn