光电技术及应用

DOI：10.16818/j.issn1001－5868.2023010303

基于改进跳蛛算法的二维OTSU限重标志识别算法

代少升，方武鳞*

（重庆邮电大学 通信与信息工程学院，重庆 400065）

摘　要： 针对交通标志识别中限重标志和限速标志相似程度大、容易造成误检等问题，结合抗噪能力强的二维OTSU自动阈值分割法与寻优能力强的跳蛛优化算法，提出了一种基于跳蛛优化的二维OTSU限重标志识别算法。该算法通过统计交通标志牌感兴趣区域的二维灰度直方图，采用改进跳蛛算法求解二维OTSU分割阈值，快速地实现交通标志图像的二值化，再利用DBSCAN对二值化图像边缘点聚类，最后根据聚类结果的相对位置正确识别限重标志。实验结果表明：在不降低识别率的条件下，运算时间降低了34.16%，并能够正确区别限重标志和限速标志。

关键词： 限重标志；二维直方图；二维OTSU；跳蛛优化算法；DBSCAN

中图分类号： TN929.11　**文章编号：** 1001－5868(2023)02－0265－07

Two-dimensional OTSU Weight Limit Sign Recognition Algorithm Based on Improved Jumping Spider Algorithm

DAI Shaosheng，FANG Wulin

（School of Communication and Information Engin.，Chongqing University of Posts and Telecommun.，Chongqing 400065，CHN）

Abstract： In view of the large similarity between the weight limit sign and the speed limit sign，which is easy to cause false detection and other problems，combining the two-dimensional OTSU automatic threshold segmentation method with strong anti-noise ability and the Jumping Spider Optimization Algorithm (JSOA) with strong searching ability，a two-dimensional OTSU weight limit sign recognition algorithm based on improved JSOA is proposesd. The algorithm counted the two-dimensional gray histogram of region of interest of traffic signs，solved the threshold of segmentation of two-dimensional OTSU using the improved jumping spider algorithm，quickly realized the binarization of traffic sign images，used DBSCAN to cluster the edge points of the binarization image，and finally correctly identified the weight limit sign according to the relative position of the clustering results. The experimental results show that the operation time is reduced by 34.16% without reducing the recognition rate，and the weight limit sign and speed limit sign can be correctly distinguished.

Key words： weight limit sign；two-dimensional gray histogram；two-dimensional OTSU；JSOA；DBSCAN

0　引言

超速行驶[1]是引发严重交通事故的最主要原因。近年来，高级驾驶辅助系统(Advanced Driver Assistance Systems，ADAS)得到巨大发展[2]，其中交通标志识别功能可以识别道路限速信息并及时提醒驾驶员车辆速度限制，有助于行车安全，减少事故发生率。由于限重标志与限速标志外形十分相似，如果错把限重标志识别为限速标志，辅助驾驶系统的作用也会适得其反。为了能准确地筛选识别结果

收稿日期：2023－01－03.

* **通信作者：**方武鳞　**E-mail：** fangwulin96@163.com

中的限重标志，本文提出一种限重识别算法，识别到交通标志牌的数字后，利用改进跳蛛算法[3]求解二维 OTSU[4] 分割阈值，快速地完成对交通标志图像[5]的二值化，通过 DBSCAN[6] 聚类二值化图像边缘点，根据聚类结果的相对位置识别是否是限重标志，对行车安全和辅助驾驶都有着重要意义。

1 图像预处理

行驶车辆遇到交通标志[7]是一个由远及近的过程，不同位置限速牌在图像中大小不同，甚至可能存在形变。主要包括数字定位、ROI（Region of Interest）选取等。

1.1 数字定位

本文数字定位方法是使用神经网络对数字进行识别[8-9]，同时获取数字的位置信息[10]，根据数字位置信息统计相应感兴趣区域的二维灰度直方图并采用本文方法判断是否属于限重标志。

1.2 ROI 选取

在识别数字后选取如图 1 所示 ROI1，ROI2 作为后续二维直方图的计算区域。ROI1，ROI2 区域与数字区域同宽，高为数字区域的 0.4 倍，不包含角落边长为区域高度一半的三角形。ROI3（高为数字 0 的一半，宽为高的 1.5 倍）为限重判定区域，将 ROI3 缩放成 75×50 像素大小，为后续二值化及 DBSCAN 处理提供统一尺寸图片。



图 1 ROI 选取示意图

2 限重判别算法

2.1 二维 OTSU

二维 OTSU 是在图像灰度直方图增加邻域均值信息而形成的二维直方图，相比于一维直方图[11]，二维直方图在含有噪声的情况下依然能很好地将前景和背景区分开。其原理如下：

设 $f(x,y)(0\leqslant x<W,0\leqslant y<H)$ 为宽为 W，高为 H，灰度级为 L 的灰度图上 (x,y) 位置的灰度值，$g(x,y)$ 为以 (x,y) 为中心 $n\times n$ 的邻域均值。记 $f(x,y)$ 和 $g(x,y)$ 所组成的二元组为 (i,j)，二元组 (i,j) 出现的频数为 f_{ij}，则：

$$\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}f_{ij}=HW \tag{1}$$

$$p_{ij}=\frac{f_{ij}}{HW},\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}p_{ij}=1 \tag{2}$$

式中，p_{ij} 为二元组 (i,j) 的概率密度。

假设 s 是像素灰度值分割阈值，t 是邻域灰度均值分割阈值，(s,t) 将概率密度图如图 2 分割为 C_0,C_1,N_0,N_1 四部分。C_0,C_1 代表前景和背景区域，N_0,N_1 代表部分边缘和噪声区域。

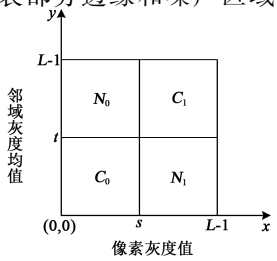


图 2 二维灰度直方图

区域 C_0,C_1 的概率和为：

$$\omega_0=\sum_{i=0}^{s}\sum_{j=0}^{t}p_{ij} \tag{3}$$

$$\omega_1=\sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}p_{ij} \tag{4}$$

C_0,C_1 对应的二维直方图均值为：

$$\boldsymbol{\mu}_0=(\mu_{0i},\mu_{0j})^{\mathrm{T}}=\left(\sum_{i=0}^{s}\sum_{j=0}^{t}\frac{ip_{ij}}{\omega_0},\sum_{i=0}^{s}\sum_{j=0}^{t}\frac{jp_{ij}}{\omega_0}\right)^{\mathrm{T}} \tag{5}$$

$$\boldsymbol{\mu}_1=(\mu_{1i},\mu_{1j})^{\mathrm{T}}=\left(\sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\frac{ip_{ij}}{\omega_1},\sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\frac{jp_{ij}}{\omega_1}\right)^{\mathrm{T}} \tag{6}$$

二维直方图的全局均值为：

$$\boldsymbol{\mu}_{\mathrm{T}}=(\mu_{\mathrm{T}i},\mu_{\mathrm{T}j})^{\mathrm{T}}=\left(\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}ip_{ij},\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}jp_{ij}\right)^{\mathrm{T}} \tag{7}$$

非零概率密度值一般分布在图 2 中 $y=x$ 区域附近，即对角线附近，而远离对角线区域概率和约等于零，则有：

$$\omega_0+\omega_1\approx 1 \tag{8}$$

$$\boldsymbol{\mu}_{\mathrm{T}}\approx\omega_0\boldsymbol{\mu}_0+\omega_1\boldsymbol{\mu}_1 \tag{9}$$

类间的离散度矩阵为：

$$\boldsymbol{S}_{\mathrm{B}}=\sum_{k=0}^{1}P_{\mathrm{r}}(C_k)[(\boldsymbol{\mu}_{\mathrm{T}}-\boldsymbol{\mu}_k)(\boldsymbol{\mu}_{\mathrm{T}}-\boldsymbol{\mu}_k)^{\mathrm{T}}] \tag{10}$$

S_{B} 的迹作为类间离散度测度，有：

$$\begin{aligned}\mathrm{tr}(\boldsymbol{S}_{\mathrm{B}})=&\omega_0[(\mu_{\mathrm{T}i}-\mu_{0i})^2+(\mu_{\mathrm{T}j}-\mu_{0j})^2]+\\&\omega_1[(\mu_{\mathrm{T}i}-\mu_{1i})^2+(\mu_{\mathrm{T}j}-\mu_{1j})^2]\end{aligned} \tag{11}$$

则最佳阈值(s',t')为：

$$\mathrm{tr}[\boldsymbol{S}_{\mathrm{B}}(s',t')]=\max_{0\leqslant s,t<L}\{\mathrm{tr}[\boldsymbol{S}_{\mathrm{B}}(s,t)]\} \tag{12}$$

2.2　跳蛛优化算法

跳蛛优化算法是由 Hernán Peraza-Vázquez 等提出的一种新型元启发式算法。该算法是模拟跳蛛的捕猎行为而设计的，具有求解速度快、不易陷入局部最优等优点。算法模拟了跳蛛捕猎时的迫害、跳向猎物、搜索行为，并通过信息素来优化位置不佳的跳蛛。

2.2.1　迫害

当跳蛛跳跃距离不够捕捉猎物时，它会通过一些隐蔽的移动靠近猎物。最终迫害的移动策略如式(13)所示。

$$\vec{x}_i(g+1)=\frac{1}{2}(\vec{x}_i(g)-\vec{x}_r(g)) \tag{13}$$

式中，$\vec{x}_i(g+1)$为第 i 只跳蛛(搜索代理)在迭代第 $g+1$ 次的位置，$\vec{x}_i(g)$表示第 i 只跳蛛在迭代 g 次的位置，$\vec{x}_r(g)$表示第 r 只跳蛛在第 g 次迭代的位置。其中 r 是在从 1 到最大搜索个体数之间的随机整数，并且 $i\neq r$。

2.2.2　跳上猎物

跳蛛扑向猎物的行为可以表示为初速度为 V_0，与水平夹角为 ϕ 的抛物线运动。该抛物线运动水平方向速度与竖直方向速度为：

$$\begin{aligned}\vec{x}_i&=V_0\cos(\alpha)t\vec{i}\\\vec{y}_i&=\left(V_0\sin(\alpha)t-\frac{1}{2}gt^2\right)\vec{j}\end{aligned} \tag{14}$$

每次迭代的时间为 1，可以得到：

$$y=x\tan(\alpha)-\frac{gx^2}{2V_0^2\cos^2(\alpha)} \tag{15}$$

最后，跳向猎物可以用以下表达式表示：

$$\begin{aligned}\vec{x}_i(g+1)&=\vec{x}_i(g)\tan(\alpha)-\frac{g\vec{x}_i^2(g)}{2V_0^2\cos^2(\alpha)}\\\alpha&=\frac{\phi\pi}{180}\end{aligned} \tag{16}$$

式中，$\vec{x}_i(g+1)$为第 i 只跳蛛在迭代第 $g+1$ 次的位置，$\vec{x}_i(g)$表示第 i 只跳蛛在迭代 g 次的位置，$V_0=100$ mm/seg，$g=9.806\,65$ m/s^2，α 是由在(0，1)区间产生的随机值 ϕ 计算而来的。

2.2.3　搜索猎物

跳蛛在周围环境进行随机搜索以找到猎物，算法提供了局部搜索和全局搜索两个表达式，局部搜索的表达式如下：

$$\vec{x}_i(g+1)=\vec{x}_{\mathrm{best}}(g)+WALK\left(\frac{1}{2}-\varepsilon\right) \tag{17}$$

式中，$\vec{x}_i(g+1)$为第 i 只跳蛛在迭代第 $g+1$ 次的位置，$\vec{x}_{\mathrm{best}}(g)$是历史以来适应度最小的位置，$WALK$ 是(−2,2)范围内的随机数，ε 是(0,1)范围内的随机数。

全局搜索的表达式如下：

$$\vec{x}_i(g+1)=\vec{x}_{\mathrm{best}}(g)+(\vec{x}_{\mathrm{best}}(g)-\vec{x}_{\mathrm{worst}}(g))\lambda \tag{18}$$

式中，$\vec{x}_i(g+1)$为第 i 只跳蛛在迭代第 $g+1$ 次的位置，$\vec{x}_{\mathrm{best}}(g)$是历史以来适应度最小的位置，$\vec{x}_{\mathrm{worst}}(g)$是当前最差个体位置，$\lambda$ 是 μ 为 0，θ 为 1 的柯西随机数。

2.2.4　信息素

第 i 只跳蛛信息素的计算公式为：

$$pheromone(i)=\frac{Fitness_{\max}-Fitness(i)}{Fitness_{\max}-Fitness_{\min}} \tag{19}$$

式中，$Fitness_{\max}$ 为最大适应度，$Fitness_{\min}$ 为最小适应度，$Fitness(i)$为个体 i 的适应度，在本文中，

$$Fitness(i)=-\mathrm{tr}[\boldsymbol{S}_{\mathrm{B}}(s_i,t_i)] \tag{20}$$

当信息素小于或等于 0.3 时，会使用如下的位置更新公式：

$$\vec{x}_i(g)=\vec{x}_{\mathrm{best}}(g)+\frac{1}{2}(\vec{x}_{r1}(g)-(-1)^{\sigma}\vec{x}_{r2}(g)) \tag{21}$$

式中，$\vec{x}_i(g)$是信息素低于阈值的个体位置，$\vec{x}_{\mathrm{best}}(g)$是历史以来适应度最小的位置，$r_1$，$r_2$ 是在从 1 到最大搜索个体数之间的随机整数，且 $r_1\neq r_2$，$\vec{x}_{r1}$，$\vec{x}_{r2}$ 分别是第 r_1，r_2 个被选中的个体。

2.3　改进跳蛛算法的二维 OTSU 算法

原始的跳蛛优化算法虽然寻优能力强，但在局部搜索和全局搜索未做到很好的平衡[12]，且在各维度范围均在 (0,255)的二维 otsu 解空间里会有大量的越界情况[13-14]。统计 100 次跳蛛数为 30，最大迭代次数为 100 的实验数据，如表 1 所示，其中 p_{ol} 为越下界的概率，p_{ou} 为越上界的概率。

对于不同的策略，其越上、下界的概率也各不相同。为降低算法的越界概率，增加一个偏置量对原

算法做出修改，偏置量$offset$的定义如下：

$$offset = \left\lfloor \frac{ub - lb}{2} \right\rfloor \tag{22}$$

式(13)修改为：

$$x_i(g+1) = offset + \frac{1}{2}(\vec{x}_i(g) - \vec{x}_r(g)) \tag{23}$$

式(16)修改为：

$$\vec{x}_i(g+1) = offset + [\vec{x}_i(g) - offset]\tan(\alpha) - \frac{g[\vec{x}_i(g) - offset]^2}{2V_0^2\cos^2(\alpha)} \tag{24}$$

式(21)修改为：

$$\vec{x}_i(g) = \vec{x}_{\text{best}}(g) + \frac{1}{2}([\vec{x}_{r1}(g) - offset] - (-1)^{\sigma}[\vec{x}_{r2}(g) - offset]) \tag{25}$$

修改后迫害、跳向猎物、更新低信息素跳蛛行为的越界概率大大下降，提高算法的搜索效率。其越界概率如表 2 所示，JSOA 和改进 JSOA 的收敛曲线如图 3 所示。

表 1 JSOA 各行为越界概率

更新方式	p_{ol}	p_{ou}
迫害/%	54.34	0.00
跳向猎物/%	100	0.00
局部搜索/%	0.00	0.17
全局搜索/%	2.14	2.99
更新低信息素跳蛛/%	0.02	27.92

表 2 改进 JSOA 各行为越界概率

更新方式	p_{ol}	p_{ou}
迫害/%	0.00	0.00
跳向猎物/%	0.00	0.00
局部搜索/%	0.00	0.04
全局搜索/%	1.31	1.59
更新低信息素跳蛛/%	0.02	2.12

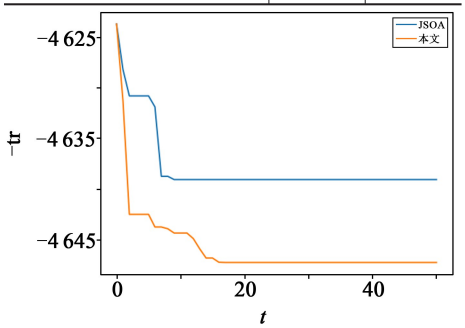


图 3 JSOA 与本文改进 JSOA 收敛曲线

2.4 限重判别

通过计算限重判别区内圆环区域的相对距离，与判别阈值比较，小于阈值为限速标志，大于阈值为限重标志。

2.4.1 相对距离

利用上文方法计算出图像分割阈值后，对限重判别区进行二值化，筛选出水平方向梯度减小最多的点，即左边是 255 右边是 0 的边缘点。通过 DBSCAN 对边缘点进行聚类，定义类内横坐标均值为圆环相对距离。聚类效果如图 4 所示。

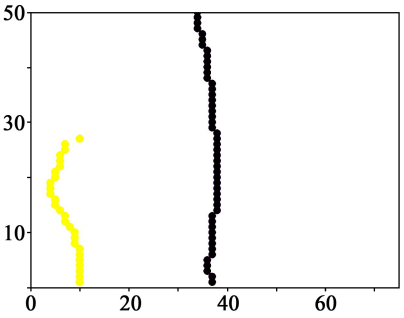
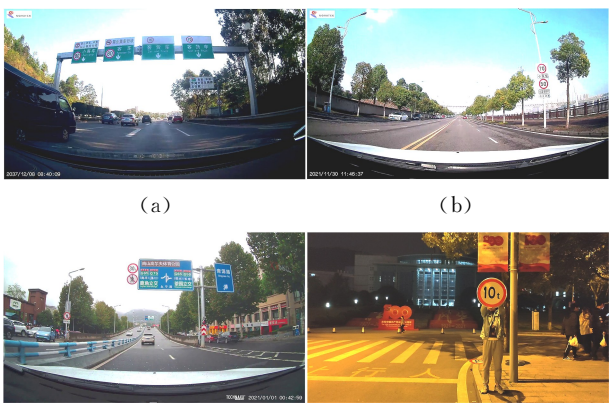


图 4 DBSCAN 聚类图

2.4.2 阈值选取

阈值选取的好坏直接影响判别的准确率[15]，本文基于自建数据集选取最佳相对距离判别阈值。数据集由 260 张限速标志图片和 247 张限重标志组成，主要由安装摄像头的车辆上路采集的图片和模拟场景拍摄图片(83 张)，包含不同距离的阴天、晴天、城市路段、高速等不同场景下的图片。数据集部分图片如图 5 所示。



(a) (b)

(c) (d)

图 5 部分数据集展示图

选取不同阈值，会有不同的限速、限重、整体错判概率，如图 6 所示。

限速判错概率随着阈值的增大逐渐减小至 0，限重判错概率随着阈值的增大逐渐增大至 1.0，整体判错概率在阈值为 26，27，28 时为零，故最终阈值选为 27。

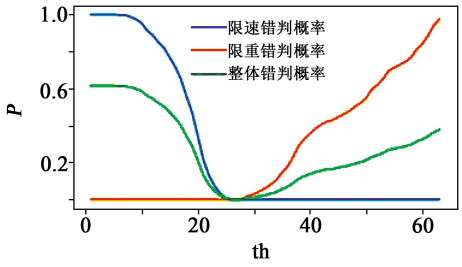


图 6　判错概率图

2.5　算法流程

算法整体流程图如图 7 所示，由 ROI1，ROI2 组成区域的二维直方图后，通过改进跳蛛算法搜索阈值，先随机初始化位置并计算其适应度，每个跳蛛随机采取迫害/跳向猎物/局部搜索/全局搜索中的一种策略，每次搜索完毕后更新低信息素个体、最优个体、个体适应度，直到满足迭代标准。最后通过分割限重判别图像并对边缘点聚类算出相对距离，判别交通标志类别。

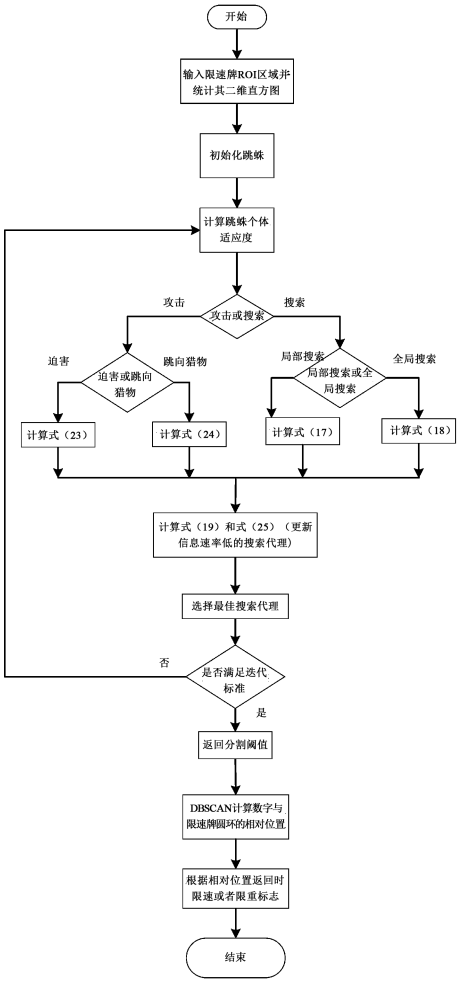


图 7　限重检测算法流程图

3　实验结果及分析

实验使用 Intel-Corei7-4710MQCPU@2.50 GHz 处理器、16 GB 内存、Windows10(64 位)操作系统，并在 PyCharm2021.3＋Python3.5 集成开发环境下进行编程。改进 JSOA 搜索个体数从 2～10 不等，最大迭代次数设为 30，DBSCAN 中 ε 设为$\sqrt{10}$，*MinPts* 设为 2。实验图像采用安装有摄像头的测试车辆实际上路采集的视频，图像大小为 1 920×1 080 像素，下面给出两个典型的交通标志在跳蛛数为 10，迭代次数为 10 的判别过程。

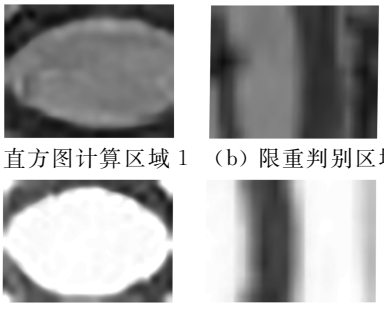
通过由神经网络组成数字识别模块识别图片中的数字信息并获取其位置信息。



(a) 结果 1　　(b) 结果 2

图 8　数字识别结果

根据得到的数字信息按照图 1 所示 ROI1，ROI2 将图像裁剪拼接得到直方图计算区域，并将 ROI3 裁剪缩放如图 9 所示。

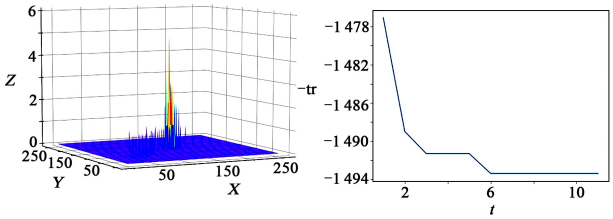


(a) 直方图计算区域 1　(b) 限重判别区域 1

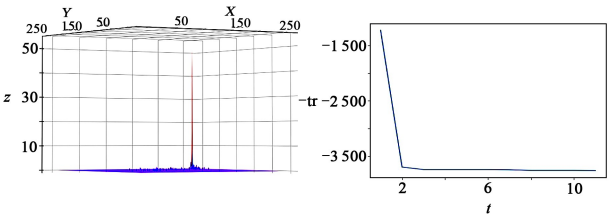
(c) 直方图计算区域 2　(d) 限重判别区域 2

图 9　直方图计算区域与限重判别区域

统计直方图计算区域中的二维直方图信息，并通过改进 JSOA 搜索最佳适应度值，得到相应二维直方图分割阈值。



(a) 二维直方图 1　　(b) 适应度收敛曲线 1



(c) 二维直方图 2　　(d) 适应度收敛曲线 2

图 10　二维直方图与适应度收敛曲线

将限重判别区域按照计算出的二维直方图阈值进行分割，效果如图 11 所示。



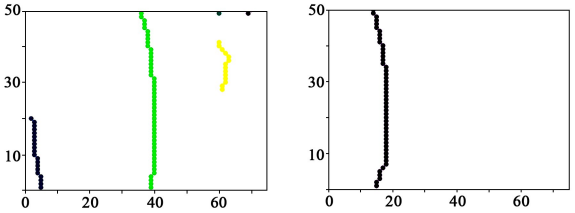
(a) 直方图计算区域 1 分割图　(b) 限重判别区域 1 分割图



(c) 直方图计算区域 2 分割图　(d) 限重判别区域 2 分割图

图 11　直方图计算区域与限重判别区域分割图

用 DBSCAN 对边缘点聚类，聚类结果如图 12 所示。



(a) 限重判别区域 1 聚类图　(b) 限重判别区域 2 聚类图

图 12　DBSCAN 聚类图

将改进 JSOA 计算的适应度、分割阈值与遍历法计算出的最佳适应度、最佳阈值进行比较，计算出圆环相对距离并给出最终判定结果。

表 3　本文方法结果图

参数	限重判别区 1	限重判别区 2
最佳适应度	−1 493.79	−3 777.01
本文适应度	−1 493.39	−3 761.21
最佳阈值	(98,117)	(181,212)
本文搜索阈值	(97,121)	(179,198)
圆弧相对距离	39.14	17.06
是否是限重	是	否

可以看到在跳蛛数及最大迭代次数均为 10 的改进 JSOA 搜索的适应度不一定是最优解，分割阈值与最佳阈值不同，但对二值化、DBSCAN、最后限重的判别的影响较小。为了观察不同跳蛛数量、迭代次数对限重识别正确率的影响，做了相应的对比实验，结果如图 13 所示。

实验表明在一定范围内随着迭代次数和跳蛛数的增加，限重识别准确率也增加，在跳蛛数为 10 迭代次数为 2 时，数据集的识别准确率达到 100%，其搜索过的区域不足传统二维 OTSU 搜索空间的千份之一，算法整体运行时间减少 32.16%。实际应用中可能有更复杂场景，为提高实际应用中算法的可靠性，可将跳蛛数量和迭代次数增大。

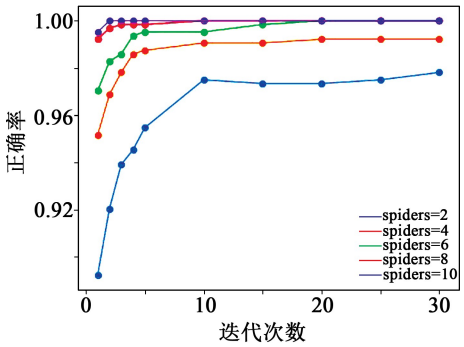


图 13　跳蛛数量与迭代次数对判别正确率的影响图

4　结论

对于限速标志牌和限重标志牌外形相似，容易识别错误。本文提出一种基于改进跳蛛算法的二维 OTSU 限重标志识别算法。该算法以摄像头获取的彩色图像作为输入，通过神经网络识别交通标志和其数字位置信息后，统计感兴趣区域二维灰度直方图，并用改进跳蛛优化算法求解二维 OTSU 分割阈值，利用阈值将限重判别区进行二值化，最后通过 DBSCAN 聚类对边缘点进行聚类，根据聚类结果相对位置，判定是否为限重标志。与传统二维 OTSU 计算相比，需要计算的适应度数量在准确率不变的前提下，减少至不足整个搜索空间的千分之一，整体运算时间减少 32.16%。该算法能有效区分限速标志与限重标志，对行车安全和辅助驾驶都有着重要意义。

参考文献：

[1] 钟　宁. 超速行驶的道路交通安全风险管控研究[D]. 南京：东南大学，2018.

Zhong Ning. Study of traffic safety risk management and control based on speeding behaviour[D]. Nanjing: Southeast

University，2018.

[2] Lubkowski S D，Lewis B A，Gawron V J，et al. Driver trust in and training for advanced driver assistance systems in real-world driving[J]. Transportation Research Part F：Traffic Psychology and Behaviour，2021，81：540-556.

[3] Peraza-Vázquez H，Peña-Delgado A，Ranjan P，et al. A bio-inspired method for mathematical optimization inspired by arachnida salticidade[J]. Mathematics，2022，10(1)：1-32.

[4] 刘健庄，栗文青. 灰度图象的二维 Otsu 自动阈值分割法[J]. 自动化学报，1993，19(1)：101-105.

Liu Jianzhuang，Li Wenqing. The automatic thresholding of gray-level pictures via two-dimensional Otsu method[J]. Acta Automatica Sinica，1993，19(1)：101-105.

[5] Yang Z，Fan X，Yu Z，et al. Traffic lights recognition based on position feature[C]// Inter. Conf. on Green Intelligent Transportation System and Safety，2022：133-146.

[6] 郭艳婕. 改进的 DBSCAN 算法及其应用研究[D]. 太原：中北大学，2020.

Guo Yanjie. Research on improved DBSCAN algorithm and its application[D]. Taiyuan：North University of China，2020.

[7] 陈　哲. 自然场景下的小目标交通标志检测技术研究[D]. 南京：南京邮电大学，2022.

Chen Zhe. The research on small target traffic sign detection technology in natural scene[D]. Nanjing：Nanjing University of Posts and Telecommunications，2022.

[8] 肖雨晴，杨慧敏. 目标检测算法在交通场景中应用综述[J]. 计算机工程与应用，2021，57(6)：30-41.

Xiao Yuqing，Yang Huimin. Research on application of object detection algorithm in traffic scene[J]. Computer Engineering and Applications，2021，57(6)：30-41.

[9] 张兴国，刘晓磊，李　靖，等. BP 神经网络下的限速交通标志实时检测识别[J]. 西安电子科技大学学报，2018，45(5)：136-142.

Zhang Xingguo，Liu Xiaolei，Li Jing，et al. Real-time detection and identification of speed limit traffic signs under the BP neural network[J]. J. of Xidian University，2018，45(5)：136-142.

[10] Ahlawat S，Choudhary A，Nayyar A，et al. Improved handwritten digit recognition using convolutional neural networks (CNN)[J]. Sensors，2020，20(12)：3344.

[11] Otsu N. A threshold selection method from gray-level histograms[J]. IEEE Trans. on Systems，Man，and Cybernetics，1979，9(1)：62-66.

[12] Li T，Liu Y，Chen Z. Design of gas turbine cooling system based on improved jumping spider optimization algorithm[J]. Machines，2022，10(10)：909.

[13] 甘福宝，黄友锐，韩　涛，等. 基于狮群优化二维 Otsu 算法的输送带撕裂检测方法[J]. 工矿自动化，2019，45(10)：55-60.

Gan Fubao，Huang Yourui，Han Tao，et al. Conveyor belt tear detection method based on lion group optimization two-dimensional Otsu algorithm[J]. Industry and Mine Automation，2019，45(10)：55-60.

[14] 杨宇博，田慕琴. 基于麻雀优化的二维 Otsu 煤岩界面识别方法[J]. 现代电子技术，2022，45(19)：49-53.

Yang Yubo，Tian Muqin. Two-dimensional Otsu coal-rock interface recognition method optimized by sparrow search algorithm[J]. Modern Electronics Technique，2022，45(19)：49-53.

[15] Cheremkhin P A，Kurbatova E A，Evtikhiev N N，et al. Adaptive digital hologram binarization method based on local thresholding，block division and error diffusion[J]. J. of Imaging，2022，8(2)：15.

作者简介：

代少升(1974—)，男，河南省潢川县人，博士，教授，主要从事红外成像技术系统及 SOPC 嵌入式系统设计与开发；

方武鳞(1996—)，男，广西省平南县人，硕士研究生，主要研究方向为图像处理与模式识别。