

# 采用二维 Otsu 直方图斜分快速算法 实现方式的改进

钱卫星<sup>1,2</sup>, 黄丽亚<sup>2</sup>

(1. 杭州职业技术学院 信息工程学院, 浙江 杭州 310018;  
2. 南京邮电大学 电子科学与工程学院, 江苏 南京 210003)

**摘要:** 在二维 Otsu 自适应阈值快速算法的斜方窄带判决域特性基础上,依据斜分线穿越窄带区的同一特性,设立算法辅助轴,对原快速算法实现方式进行改进.简化算法实现过程,在原算法降维基础上,进一步降低算法复杂度.结果表明:改进后的二维 Otsu 自适应阈值快速算法更加适应图像分割的实际工况,在实验验证中取得良好的图像分割效果.

**关键词:** 图像分割; 二维 Otsu; 斜方窄带; 自适应阈值

**中图分类号:** TP 391.41      **文献标志码:** A

阈值分割<sup>[1]</sup>是图像处理算法的典型方法.日本学者大津于 1979 年提出了经典 Otsu 算法<sup>[2]</sup>,并发展了二维 Otsu 自适应阈值分割方法,充分应用了图像的关联信息量<sup>[3,4]</sup>,使分割效果得到改善.文献[5-7]对算法进行改进,改判决域直分为斜分,提出不同阈值分割准则,降低算法运算量,提高算法处理速度,在实时处理中得到较好的应用.文献[8]提出多段折线斜分阈值分割法.文献[9-11]提出二维 Otsu 快速斜分算法的最小误差方式改进方法,但在斜分算法实现时,对斜方窄带域的划分较复杂,同一特性域内也因边界域值的变化要进行再细分才能实现,需要进行 20 多次细分,大大增加了算法实现时的程序复杂度.文献[12-14]等从不同的侧面提出了快速算法方案.本文依据斜分窄带域区域同一特性原则,加入算法辅助轴,改进简化了算法实现过程.

## 1 算法原理

在经典 Otsu 算法一维直方图的基础上,关联图像中像素点与其邻域像素点的信息,形成图像像素信息联合直方图.建立 Otsu 算法二维直方图,并引入自适应阈值判别方法.原始图像,如图 1 所示.图像  $f(x,y)$  的灰度等级为  $0,1,2,\cdots,L-1$ ,共  $L$  级,其邻域平滑图像  $g(x,y)$  (以  $k\times k$  的邻域图像像素均值作为该灰度值)的灰度级为  $L$  级,其联立的二维直方图,如图 2 所示.



图 1 原始图像

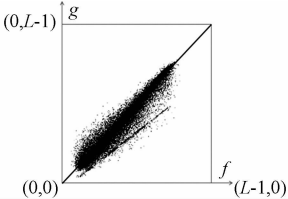


图 2 二维直方图

由图 2 可知:图像像素的灰度等级主要排布在二维直方图斜对角线附近,图像前景、背景各自像素灰度值和其邻域灰度均值较接近,但在两者分界领接处,像素自身和邻域灰度均值差距较大.在图 2 上设定一阈值数据  $(s,t)$ ,水平垂直分割直方图为  $A\sim D$  等 4 个区域,如图 3 所示.由图 3 可知:区域  $A,D$  大部分远离对角线,像素灰度与领域均值灰度差距较大,为图像边缘或图像噪声等;区域  $B,C$  分别对应于图像前景和背景.

在二维直方图的阈值直分法中, 由于区域 A 和区域 D 在靠近对角线附近实际为图像主体的像素数据被忽略了, 使图像分割达不到理想效果, 且在计算最佳阈值点数据  $s, t$  时, 直分法图像每帧均需要作双重循环  $s \times t + (L-s) \times (L-t)$  次累加运算, 对图像实时连续处理有一定影响. 二维直方图的阈值斜分快速算法分割方式, 如图 4 所示. 建立截距为  $N$  的与对角线平行的斜方窄带域, 截距斜线表达式为

$$f_{\text{上}} = g + N, \quad f_{\text{下}} = g - N. \quad (1)$$

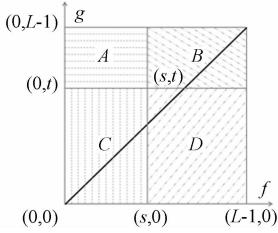


图 3 二维直方图直分法

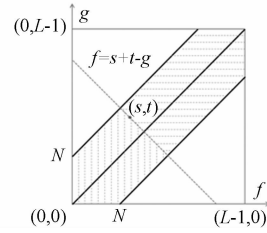


图 4 二维直方图斜分法

Fig. 3 2D Otsu histogram vertical segmentation method Fig. 4 2D Otsu histogram oblique segmentation method

设定阈值点  $(s, t)$ , 斜分直线  $f$  通过阈值点, 与对角线相垂直, 将斜分窄带域分割成上下两部分. 位于斜分线下侧的窄带区域对应于图像前景, 上侧部分对应于图像背景, 该斜分线表达式为

$$f = s + t - g. \quad (2)$$

采用斜分分割, 在计算最佳阈值点时, 运算仅包括窄带内数据, 可以忽略窄带外部分, 缩小运算范围. 将  $s+t$  看成一个整体, 从而将阈值判断从二维  $(s, t)$  转换成一维  $s+t$  阈值判断. 在编程计算时, 从双重循环降为单重循环, 大幅降低了数据运算量. 由图 4 可知: 当截距  $N$  足够大时, 窄带域内可以包括所有概率不为零的像素点信息, 获得理想的阈值点数据, 但运算量增大.

## 2 算法实现及改进

### 2.1 原二维 Otsu 斜分快速算法实现方式

根据  $(s, t)$  的位置, 将其分为左侧区域 ( $f = N - g, s+t < N$ )、右侧区域 ( $f = 2L - 2 - N - g, 2L - 2 - N \leq s+t < 2L - 2$ )、中间区域 ( $f = N - g, f = 2L - 2 - N - g, N \leq s+t < 2L - 2 - N$ ), 如图 5 所示.

以区间的算法实现为例. 分割斜线左侧部分为前景区, 根据斜分直线与上截距直线交点坐标与  $N$  的关系, 原算法实现分为 2 种情况, 如图 6 所示.

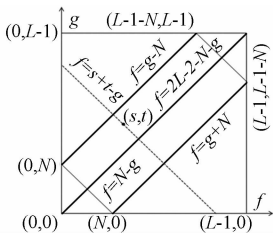


图 5 斜分法区域划分

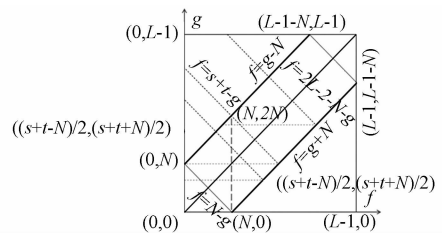
图 6  $N \leq s+t < 2L-2-N$  时的区域划分

Fig. 5 Region division for oblique segmentation method

Fig. 6  $N \leq s+t < 2L-2-N$  region division

1)  $(s+t+N)/2 < 2N$ , 即  $s+t < 3N$ . 此时, 图像前景  $C_0$  包括 3 个区间: 当  $g < (s+t-N)/2$  时,  $0 \leq f \leq g+N$ ; 当  $(s+t-N)/2 \leq g < N$  时,  $0 \leq f \leq s+t-g$ ; 当  $N \leq g < (s+t+N)/2$  时,  $g-N \leq f \leq s+t-g$ .

2)  $(s+t+N)/2 \geq 2N$ , 即  $s+t \geq 3N$ . 此时, 前景  $C_0$  包括 3 个区间. 当  $g < N$  时,  $0 \leq f \leq g+N$ ; 当  $N \leq g < (s+t-N)/2$  时,  $g-N \leq f \leq s+N$ ; 当  $(s+t-N)/2 \leq g < (s+t+N)/2$  时,  $g-N \leq f \leq s+t-g$ .

后景区  $C_1$  算法实现与此相似. 其他 2 个区间段的算法实现以此类推.

### 2.2 二维 Otsu 斜分快速算法实现改进方式

原二维 Otsu 斜分快速算法虽然实现了降维运算, 但在具体算法实现时分段复杂, 在程序实现时不易累计  $\omega_0(s, t), \omega_1(s, t), \mu_{0,i}(s, t), \mu_{0,j}(s, t)$ . 在原算法斜方窄带划分的基础上, 增加辅助轴  $s+t$ , 并以斜分线与窄带区的交点坐标变化特点为依据, 在算法实现时, 对前景图像和后景图像再划分, 如图 7 所示.

图 7 中: 斜分线  $f = s_0 + t_0 - g$ , 在辅助轴  $s+t$  上的轴截距为  $s_0 + t_0$ . 由图 7 可知: 斜分线与斜方窄带

区的交点坐标变换存在 3 个不同区段.

- 1) 当  $s+t < N$  时, 边界交点坐标分别为  $(0, s+t), (s+t, 0)$ ;
- 2) 当  $N \leq s+t < 2L-2-N$  时, 边界交点的坐标分别为  $((s+t-N)/2, (s+t+N)/2), ((s+t+N)/2, (s+t-N)/2)$ ;
- 3) 当  $2L-2-N \leq s+t < 2L-2$  时, 边界交点的坐标分别为  $(s+t-L+1, L-1)$  和  $(t-L, s+t-L+1)$  划分图.

无论在哪一区段, 上交点坐标沿斜分线向下交点坐标过渡时,  $f, g$  的步进增量分别为  $+1, -1$ . 因此, 在算法实现时, 以辅助轴  $s+t$  数值从 0 到  $2L-2$  递进变化为主循环量, 以交点坐标变化为副循环量, 统一处理  $\omega_0(s, t), \omega_1(s, t), \mu_{0,i}(s, t), \mu_{0,j}(s, t)$  数据增减量. 改进后算法的实现程序框架如下

```
for (nThreshold_st=0;nThreshold_st<2L-2;nThreshold_st++)
if ((nThreshold_st>2L-2-N) & (nThreshold_st<=2L-2))
{
    if (nThreshold_st<=N)
    {
        g1=L-1;g2=nThreshold_st-L+1;
        f1=nThreshold_st-L+1;
        i=f1;
        for (j=g1;j>g2;j--)
        {
             $\omega_0+=dHistogram\_pij[i][j]$ ;
             $\mu_{0,i}+=i * dHistogram\_pij[i][j]$ ;
             $\mu_{0,j}+=j * dHistogram\_pij[i][j]$ ;
            i++;
        }
    }
    else
    {
         $s_b=((\omega_0 * \mu_{z,i} * \mu_{0,i})^2+(\omega_0 * \mu_{z,j} * \mu_{0,j})^2)/(\omega_0 * (1-\omega_0))$ ;
        if (s_b>temp_Data)
        {
            temp_Data=s_b;
            nThreshold_Data=nThreshold_st;
        }
    }
}
if ((nThreshold_st>N) & (nThreshold_st<=2L-2-N))
{
     $g_1=(nThreshold\_st+N)/2;g_2=(nThreshold\_st-N)/2$ 
     $f_1=(nThreshold\_st-N)/2;i=f_1$ ;
    :
}
```

最佳阈值获取耗时对比, 如表 1 所示. 表 1 中:  $t_1, t_2$  分别为算法改进前和改进后的耗时;  $t_{ave}$  为平均耗时. 由表 1 可知: 改进后比改进前的速度快了约 3.6 倍; 图像越大, 处理数据越多时, 改进后的实现方式优势越明显. 算法实现方式改进后的图像分割效果, 如图 8 所示. 由图 8 可知: 算法实现方式改进后, 图像分割效果依旧很理想.

表 1 最佳阈值获取耗时对比

Tab. 1 Time consuming comparison of optimal threshold

s

项目	第一次	第二次	第三次	第四次	第五次	$t_{ave}$
$t_1$	0.538 99	0.537 22	0.542 52	0.532 62	0.559 87	0.542 24
$t_2$	0.146 86	0.147 93	0.155 01	0.146 51	0.155 36	0.150 33

3 结束语

充分利用二维 Otsu 直方图的斜方窄带域各分区段的同一特性, 借助算法辅助轴, 统一判断依据, 斜分窄带判决域的划分由原先的 20 多个细分区域降为 3 个划分区域, 使算法实现时得到了极大简化. 在图像分割效果保持理想的情况下, 阈值计算的运算速度提高了近 4 倍. 因此, 改进的二维 Otsu 直

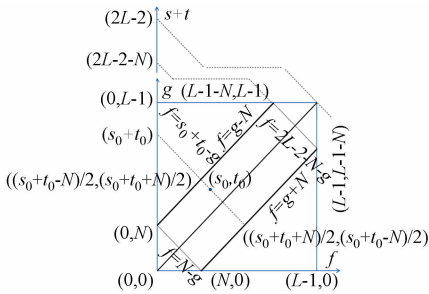


图 7 斜分新法阈值区域

Fig. 7 Region division for new oblique segmentation method



(a)  $N=20$  (b)  $N=40$

图 8 图像分割效果

Fig. 8 Image segmentation effect

方图斜分快速算法实现方式更加适合需实时处理的图像分割工况场合。

参考文献：

[1] 章毓晋. 图象分割评价技术分类和比较[J]. 中国图象图形学报, 1996, 1(2): 151-157.

[2] OTSU N. A threshold selection method from gray-level histograms[J]. IEEE Transactions on System Man and Cybernetic, 1979, 9(1): 62-66.

[3] 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.

[4] 郝颖明, 朱枫. 2 维 Otsu 自适应阈值的快速算法[J]. 中国图象图形学报, 2005, 10(4): 484-488.

[5] 杨金龙, 张光南, 厉树忠, 等. 基于二维直方图的图像分割算法研究[J]. 激光与红外, 2008, 38(4): 400-403.

[6] 吴一全, 潘喆, 吴文怡, 等. 二维直方图区域斜分的最大熵阈值分割算法[J]. 模式识别与人工智能, 2009, 22(1): 162-168.

[7] 范九伦, 雷博. 灰度图像最小误差阈值分割法的二维推广[J]. 自动化学报, 2009, 35(4): 386-393.

[8] 梁义涛, 庞蕊, 朱远坤. 灰度图像二维 Otsu 折线阈值分割法[J]. 计算机工程与应用, 2012, 48(33): 178-182.

[9] 朱齐丹, 荆丽秋, 毕荣生, 等. 最小误差阈值分割法的改进算法[J]. 光电工程, 2010, 37(7): 107-113.

[10] 陈琪, 熊博莅, 陆军, 等. 改进的二维 Otsu 图像分割方法及其快速实现[J]. 电子与信息学报, 2010, 32(5): 1100-1104.

[11] 李森, 杨恢先. 改进二维直方图区域划分的阈值分割方法研究[J]. 光电子·激光, 2013, 24(7): 1426-1433.

[12] 张新明, 李振云, 孙印杰, 等. 快速二维直方图斜分最小误差的图像阈值分割[J]. 电光与控制, 2012, 19(6): 8-12.

[13] 何志勇, 孙立宁, 陈立国. Otsu 准则下分割阈值的快速计算[J]. 电子学报, 2013, 41(2): 267-272.

[14] 钱卫星, 黄丽亚. 基于二维 Otsu 的自适应阈值快速算法改进[J]. 华侨大学学报(自然科学版), 2015, 36(4): 427-431.

## Improved Implementations for 2D Otsu Histogram Fast Algorithm

QIAN Weixing<sup>1</sup>, HUANG Liya<sup>2</sup>

(1. Electronic Information College, Hangzhou Vocational and Technical College, Hangzhou 310018, China;

2. School of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract:** Based on the characteristics of the trapezius narrowband verdict scope of 2D Otsu adaptive thresholdfast algorithm, and the same characteristic of obliqueline traversing anarrow bandregion, we proposed the establishment of algorithm auxiliary shaft which improved and simplified the realization of original fast algorithm. Compared to the original algorithm of dimension reduction, this algorithm simplify the implementation process and algorithm complexity. Results show that the improved fast algorithm of 2D Otsu adaptive threshold was more adapted to the actual condition of image segmentation, which has achieved good effect of image segmentation in experiment.

**Keywords:** image segmentation; 2D Otsu; trapezius narrowband; adaptive threshold

(责任编辑：钱筠      英文审校：吴逢铁)