

# 工业检测图像灰度波动变换自适应阈值分割算法

魏巍<sup>1,2</sup> 申铨京<sup>1,2</sup> 千庆姬<sup>3</sup>

**摘要** 工业检测图像经常受到不均光照的影响, 对该类图像局部自适应分割算法比全局算法能产生更好的分割效果. 但局部算法中基于分块的算法对分块方法缺乏指导, 而基于邻域的算法容易在背景或前景内部产生误分. 针对上述缺点, 本文提出了一种多方向灰度波动变换的自适应阈值分割算法. 该算法先从多个方向依照灰度波动对图像进行转换, 构造以多维向量为基础的灰度波动变换矩阵, 然后利用主成分分析法 (Principal component analysis, PCA) 将高维向量压缩至一维并生成变换图像, 最后运用 Otsu 算法分割变换图像. 该算法无需分块, 并且只需波动幅度阈值和布尔型背景色两个参数. 实验结果表明, 该算法能够有效减少不均光照对工业检测图像分割的影响, 与 Niblack 法、Sauvola 法等几种局部算法相比, 该法在分割效果上具有了明显的提升.

**关键词** 图像分割, 局部阈值分割, 灰度波动, Otsu 算法

**DOI** 10.3724/SP.J.1004.2011.00944

## An Adaptive Thresholding Algorithm Based on Grayscale Wave Transformation for Industrial Inspection Images

WEI Wei<sup>1,2</sup> SHEN Xuan-Jing<sup>1,2</sup> QIAN Qing-Ji<sup>3</sup>

**Abstract** The industrial inspection images are usually under non-uniform illumination, and local adaptive thresholding algorithms have better segmentation performance on them than the global ones. But the local algorithms based on image's sub-blocks are short of instructions for partitioning, and the local algorithms based on pixel's neighborhood will probably cause some misclassifications within the background or foreground. To resolve these problems, a novel adaptive thresholding algorithm based on multi-directional grayscale wave transformation is proposed in this paper. Firstly, it performs the transformation by grayscale waves in multi-directions to get a matrix of multi-dimensional vectors. Secondly, the vectors are compressed to one dimension using the principal component analysis (PCA) method, and then the Otsu global method is employed to find optimal wave threshold for segmentation on this matrix. This algorithm does not need partitioning the image any more and only takes the peak height threshold and the boolean background color as its two parameters. Experiments demonstrate that this method has a excellent capability of decreasing the influence of non-uniform illumination in industrial inspection images, and its segmentation performance is better than several other local thresholding algorithms, such as Niblack's method and Sauvola's method.

**Key words** Image segmentation, local thresholding segmentation, grayscale wave, Otsu method

图像分割技术是图像分析与理解、机器视觉等领域内重要的研究课题, 其主要目的是为了分离图像内的目标和场景, 从而进行目标识别跟踪或场景分析, 目前在工业非接触性产品质量检测以及字符识别等领域得到了广泛的应用. 众多分割算法中, 阈值分割算法以其简单有效的优点受到了人们的普遍

欢迎. 从分割的空间特性来看, 现阶段阈值分割算法大体可以分为局部自适应算法<sup>[1-10]</sup>和全局自适应算法<sup>[11-12]</sup>两大类. 其中局部算法一般可分为两类: 1) 邻域法, 根据像素某个邻域内的灰度分布来确定该像素的阈值<sup>[1-2, 4-5]</sup>; 2) 分块法, 将图像分为子块, 单独对每一子块求取阈值并对其进行分割<sup>[3, 6-9]</sup>. 相比于全局算法, 局部算法在处理光照分布不均图像方面具有明显的优势, 因而受到了广泛的重视. 但从目前的研究来看, 局部算法在字符识别领域应用较多, 而对于工业检测图像的分割仍是以全局算法居多<sup>[13]</sup>.

1986 年 Bernsen 就提出单个像素的阈值可由它的邻域最大值和最小值确定<sup>[1]</sup>. 而 Niblack 则认为像素阈值可以与邻域均值和方差联系起来<sup>[2]</sup>. 之后, Taxt 等提出利用 EM 算法获取局部阈值<sup>[3]</sup>. 2000 年, Sauvola 等提出可将图像子块归为不同类别并分别采用对应的分割准则, 此外还给出了一种新的邻域

收稿日期 2010-06-22 录用日期 2011-02-01  
Manuscript received June 22, 2010; accepted February 1, 2011  
国家自然科学基金 (60773098), 吉林省科技发展计划 (20080317) 资助

Supported by National Natural Science Foundation of China (60773098) and Development Program in Science and Technology of Jilin Province (20080317)

1. 吉林大学计算机科学与技术学院 长春 130012 2. 吉林大学符号计算与知识工程教育部重点实验室 长春 130012 3. 吉林大学物理学院 长春 130025

1. College of Computer Science and Technology, Jilin University, Changchun 130012 2. Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012 3. College of Physics, Jilin University, Changchun 130025

法来分割文本内容<sup>[4]</sup>. Kim 在 Sauvola 的基础上将邻域大小设定为可根据图像目标动态调整<sup>[5]</sup>. Huang 等则提出了一种基于自适应窗口选择的分块法<sup>[6]</sup>. Tsai 又提出了一种退火算法与 Otsu 法相结合的四叉树分块法<sup>[7]</sup>. 近年来, Chou 等利用支持向量机判定子块类别并对子块采取不同的分割策略<sup>[8]</sup>. Moghaddam 等构建了基于自适应 Otsu 法和网格模型的多尺度系统来分割低质量文本图像<sup>[9]</sup>. Chen 等提出了一种基于边缘检测的双阈值分割法, 该法的效果较大程度上取决于边缘提取的准确度<sup>[10]</sup>.

总体来看, 目前局部自适应阈值分割算法应用于工业检测图像时存在的问题有: 1) 机械地将图像划分为等大的子块并在子块内分割, 容易造成子块分割结果之间的不连贯; 2) 图像的分块数量并无统一标准, 一般依据经验、实验效果或样本学习给出, 鲁棒性较差; 3) 像素邻域法由于受到邻域的限制, 容易造成背景或前景内部的大量误分. 为了解决上述问题, 本文提出一种基于灰度波动变换的自适应阈值分割算法. 该法无需对图像分块且参数较少, 既利用了灰度的局部波动信息, 又借助了全局算法的整体协调性, 可以有效削弱图像不同局部间的光照差异, 提升对不均光照下工业检测图像的分割效果.

## 1 多方向灰度波动变换

### 1.1 灰度波动变换的思想

由于受到拍摄环境的制约, 工业检测图像经常属于明暗相间的光照不均衡图像. 在某一局部区域归属于背景灰度范围内的灰度级, 在另一局部区域内很可能就和目标灰度相等. 图 1(a) 的 Rice 图像就是一幅光照不均图像, 其下方存在一块明显的阴暗区域. 该区域的整体灰度比上方低, 如果仅凭借单一的绝对阈值来分割该图像, 那么就无法同时兼顾上下两块明暗不同的区域. 也就是说, 单纯依靠灰度信号的“绝对性”对此类光照不均衡图像进行分割是不合适的. 但既然我们能够区分出下方阴暗区域中的目标与背景, 就说明在该区域内二者之间仍然是存在灰度差异的, 这种差异就是图像局部内灰度信号之间“相对性”的体现.

从直观上看, 图像可以看作像素灰度在二维空间的波动. 对于单个像素, 其灰度的局部特性与它在波动上的位置息息相关. 位于波峰的像素灰度相对较高, 而位于波谷的像素灰度相对较低. 因此可以考虑抛开像素的原始灰度, 直接以该像素灰度在波动中的位置来替代, 这样就摒除了局部基础性光照强度的影响, 突显出了该像素的邻域相对特性. 而根据像素灰度的波动相对高度, 就能很容易地区分出目

标与背景. 这种局部波动式的分割方法不必对图像分块, 而且保证了分割后图像的连贯和自然.

图 1(b) 显示了图 1(a) 的二维灰度波动. 可以看出, 虽然图像上下方的基础光照不同, 但是波峰区域仍明显对应于目标, 而波谷区域则对应于背景. 因此只要给出像素的波动位置, 就可以依照它与波峰或波谷的接近程度来动态地划定它的归属, 具体做法类似于为每个像素赋予一个隶属于波峰的隶属值, 当像素位于“峰顶”时为 1, 位于“谷底”时则为 0.

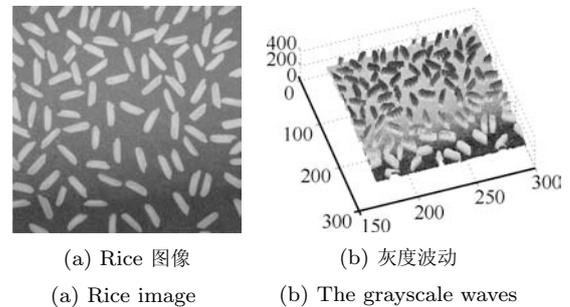


图 1 Rice 图像的灰度波动

Fig. 1 The grayscale waves of Rice image

为了获取二维波动上像素的波动位置, 本算法的思想是: 从给定的多个方向上分别将二维波动分解为相应的一维波动, 并对一维波动进行变换, 以获得一维的波动位置, 最后将各方向的变换结果加以整合以获取近似的二维波动位置. 具体做法就是从不同方向上分别对图像进行“一维灰度波动变换”, 并将每个像素各方向的一维波动位置信息合成为一个对应于该像素的高维波动向量, 该向量就近似代表了该像素二维空间的波动位置信息. 图像“一维灰度波动变换”与整体“灰度波动变换”定义如下:

**定义 1.** 给定方向  $d$ , 提取图像沿  $d$  方向的一条直线  $f_d$ , 设  $f_d$  对应的灰度曲线函数为  $g$ . 给定阈值  $\alpha$ ,  $\alpha \in [0, L - 1]$ ,  $L$  为图像灰度级数, 如果存在由  $g$  的若干极大值点坐标  $p_0, p_1, \dots$  与若干极小值点坐标  $t_0, t_1, \dots$ , 按坐标由小到大交错排列构成的序列, 不妨设为  $\{p_0, t_0, p_1, t_1, \dots, p_m, t_m\}$ ,  $p_0 < t_0 < \dots < p_m < t_m$ ,  $m$  为正整数, 且该序列满足  $g(p_i) - g(t_{i-1}) > \alpha$ ,  $g(p_i) - g(t_i) > \alpha$ ,  $i = 1, 2, \dots, m$ , 那么给定单调递增函数  $u$ , 点  $Q(k, g(k))$ ,  $k \in [p_i, t_i]$  的相对高度  $H_r(k)$  为

$$H_r(k) = \frac{g(k) - g(t_i)}{g(p_i) - g(t_i)} \quad (1)$$

则它的变换值  $w_Q$  表示为

$$w_Q = u(H_r(k)) \quad (2)$$

区间  $[t_{i-1}, p_i]$  与此同理. 将  $g$  上每个点的灰度值均以它的变换值  $w_Q$  替换, 得到曲线函数  $G$ . 转换  $G$

至图像坐标空间中,得到的函数  $F_d$  就是对直线  $f_d$  变换的结果,该过程称为对  $f_d$  的“一维灰度波动变换 (1D Grayscale wave transformation)”,用符号  $\psi$  表示. 将图像沿  $d$  方向的所有直线均进行一维灰度波动变换称为对图像的“一维灰度波动变换”.

**定义 2.** 如果从  $n$  个方向分别对图像  $f$  进行一维灰度波动变换,那么对图像任意一个像素  $Q$ ,它对应于方向  $d_1, d_2, \dots, d_n$  上的变换值  $w_1, w_2, \dots, w_n$  就可以构成一个  $n$  维向量  $W_Q = \{w_1, w_2, \dots, w_n\}$ ,  $W_Q$  称为像素  $Q$  的灰度波动向量 (Grayscale wave vector, GWV). 对图像的所有像素获取对应灰度波动向量的总过程称为图像的灰度波动变换 (Grayscale wave transformation, GWT),用符号  $\Psi$  表示. 将图像每个像素的灰度值用对应的灰度波动向量替换,由此构成的像素矩阵  $F$  称为图像的灰度波动变换矩阵 (Grayscale wave transformed matrix, GWTM).

即对  $F$  有:

$$F = \Psi(f) \tag{3}$$

设  $f_{d,i}$  为图像  $f$  在方向  $d$  上的第  $i$  条直线,  $d = d_1, d_2, \dots, d_n, i = 1, 2, \dots$ , 则对  $f_{d,i}$  有:

$$F_{d,i} = \psi(f_{d,i}) \tag{4}$$

其中,  $F_{d,i}$  表示对  $f_{d,i}$  一维灰度波动变换后的函数. 对于  $f_{d,i}$  上任意像素  $Q(x, y)$ ,  $(x, y)$  为图像坐标, 它的灰度波动向量  $W_Q$  对应于方向  $d$  的分量  $w_d$  为

$$w_d(x, y) = F_{d,i}(x, y), Q_{(x,y)} \in f_{d,i} \tag{5}$$

因此图像的整体灰度波动变换  $\Psi(f)$  是由各方向  $d_1, d_2, \dots, d_n$  的一维灰度波动变换  $\psi(f_d)$  组合而成:

$$\{\psi(f_{d_1}), \psi(f_{d_2}), \dots, \psi(f_{d_n})\} \rightarrow \Psi(f) \tag{6}$$

下面将分析灰度波动变换削弱不均光照的原因:

假设一幅图像为  $f(x, y)$ ,  $g(k) = f_{d,i}(x, y)$  为  $f$  在任意方向  $d$  上第  $i$  条截线的灰度曲线函数,  $k = 0, 1, \dots, K$ ,  $K$  为该截线的像素总数.  $g(k)$  中存在两个受到不同光照强度影响的波动  $\phi_1, \phi_2$ , 其峰顶分别为  $(p_1, g(p_1)), (p_2, g(p_2))$ , 谷底分别为  $(t_1, g(t_1)), (t_2, g(t_2))$ , 如图 2 所示, 则

$$H_r(t_1) = \frac{g(t_1) - g(t_1)}{g(p_1) - g(t_1)} = 0 \tag{7}$$

$$H_r(p_1) = \frac{g(p_1) - g(t_1)}{g(p_1) - g(t_1)} = 1 \tag{8}$$

$$H_r(t_2) = \frac{g(t_2) - g(t_2)}{g(p_2) - g(t_2)} = 0 \tag{9}$$

$$H_r(p_2) = \frac{g(p_2) - g(t_2)}{g(p_2) - g(t_2)} = 1 \tag{10}$$

所以

$$u(H_r(p_1)) = u(H_r(p_2)) \tag{11}$$

$$u(H_r(t_1)) = u(H_r(t_2)) \tag{12}$$

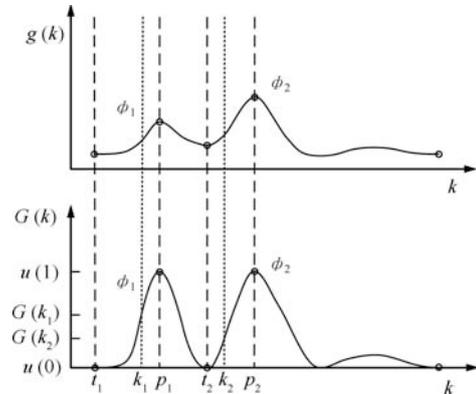


图 2 一维灰度波动变换示意图

Fig. 2 The illustration of 1D grayscale wave transformation

设点  $k_1, k_2$  分别属于波动  $\phi_1$  与  $\phi_2$ , 且  $H_r(k_1) > H_r(k_2)$ , 如图 2. 由于  $u$  单调递增, 故

$$u(H_r(k_1)) > u(H_r(k_2)) \tag{13}$$

式 (11)~(13) 说明通过一维灰度波动变换, 无论波峰  $p_1, p_2$  与波谷  $t_1, t_2$  原始灰度是多少,  $p_1$  与  $p_2$  的变换值始终相等,  $t_1$  与  $t_2$  的变换值始终相等. 而且  $\phi_1$  与  $\phi_2$  上所有像素的灰度均按照各自的相对波动高度归一化到  $[u(0), u(1)]$  区间, 如图 2. 这样就减弱了不均匀光照施加在  $\phi_1$  与  $\phi_2$  上的影响. 由式 (6) 可以将此一维变换的效果推广到二维.

为了选取最好的变换方向组合, 我们分别对 4 方向组合 ( $180^\circ/4$  间隔)、8 方向组合 ( $180^\circ/8$  间隔) 和 16 方向组合 ( $180^\circ/16$  间隔) 下的变换效果进行测试. 其中 8 方向组合设为  $d_1 = \angle 0^\circ, d_2 = \angle 26.6^\circ, d_3 = \angle 45^\circ, d_4 = \angle 63.4^\circ, d_5 = \angle 90^\circ, d_6 = \angle 116.6^\circ, d_7 = \angle 135^\circ, d_8 = \angle 153.4^\circ$  (以  $x$  轴正向为  $0^\circ$ ), 间隔近似为  $180^\circ/8$ . 这种方向组合均分了  $360^\circ$ , 涵盖了二维波动各方向的信息, 能够适应不同波动的各个形态, 鲁棒性较强, 同时也保证了各角度取正切或反正切均为整数, 简化了直线提取运算.

图 3 显示了 8 方向组合下的变换, 其他方向组合与其类似. 我们在上述三种方向组合下分别利用本文算法对一幅化纤长丝截面图像实施变换, 变换效果和处理耗时如图 4 所示.

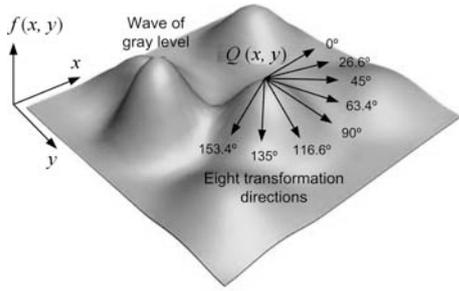


图 3 8 方向灰度波动变换示意图

Fig. 3 The illustration of grayscale wave transformation in 8 directions

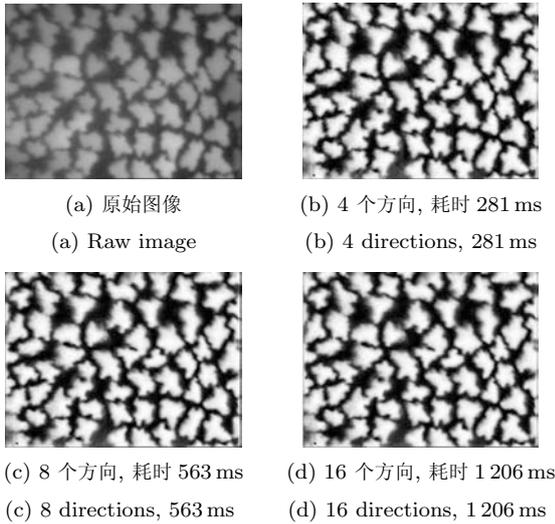


图 4 各种方向组合下灰度波动变换的效果和耗时

Fig. 4 The performance and time costs of grayscale wave transformation under various direction combinations

可以看出, 各方向组合下的运算耗时与各自所含的方向数是成正比的. 4 方向与 8 方向在目标块大体轮廓上比较接近, 但是 4 方向在细节上不如 8 方向细腻, 边缘不够圆滑, 明显残留着 4 个方向的变换痕迹. 16 方向由于方向间角度差很小, 存在一定的信息冗余, 因此与 8 方向相比仅在细节处有很小的差异, 总体效果非常接近. 所以综合考虑运算速度和变换效果, 我们选择 8 方向组合作为本算法的变换方向组合. 实际情况中, 如果分割对速度要求较高且对精度要求较低, 也可选用 4 方向组合.

因此灰度波动变换的整体过程为:

**步骤 1.** 从  $d_1$  至  $d_8$  中选定方向  $d$ , 将原始图像  $f$  在该方向上的所有直线  $f_{d,i}(x, y), i = 1, 2, \dots$ , 依次取出, 构成方向  $d$  的各条原始灰度曲线  $g_{d,i}(k)$ .

**步骤 2.** 对方向  $d$  上每一条  $g_{d,i}(k)$  做一维灰度波动变换  $G_{d,i}(k) = \psi(g_{d,i}(k))$ , 并将  $G_{d,i}(k)$  还原至原始图像空间, 得到方向  $d$  上的变换结果  $F_{d,i}(x, y)$ .

**步骤 3.** 对  $d_1$  至  $d_8$  各方向的变换全部完成后, 对每一像素的 8 个波动分量  $w_1, w_2, \dots, w_8$  进行综

合, 分别得到各个像素的八维灰度波动向量, 由此构成灰度波动变换矩阵  $F(x, y)$ .

### 1.2 一维灰度波动变换

#### 1) 一维灰度波动变换算法概述

假设  $g(k) = f_{d,i}, k = 0, 1, \dots, K$  为方向  $d$  上图像直线  $f_{d,i}(x, y)$  在一维灰度空间下的波动曲线,  $(x, y)$  为图像原始坐标,  $K$  为该图像直线的像素点总数, 曲线如图 5 所示, 其中局部极大值点  $k_{\text{peak}}$  为波峰, 局部极小值点  $k_{\text{trough}}$  为波谷. 为了方便计算, 本文引入整数  $L$  作为离散化的波动幅度等级数, 将  $u$  的值域设为  $[0, L - 1]$  整数区间. 则  $g(k)$  上  $k_{\text{peak}}$  与  $k_{\text{trough}}$  间任意一点  $(k', g(k'))$  的变换值  $G(k')$  为

$$G(k') = \left\lceil \frac{g(k') - g(k_{\text{trough}})}{g(k_{\text{peak}}) - g(k_{\text{trough}})} \times (L - 1) \right\rceil \quad (14)$$

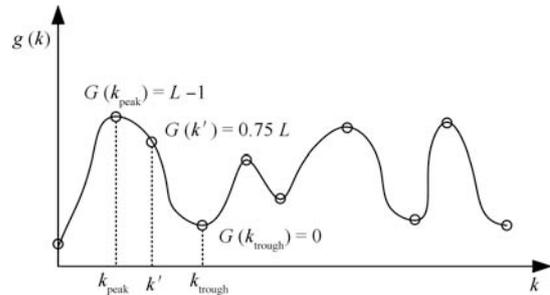


图 5 一维灰度波动变换值的计算

Fig. 5 The computation of 1D grayscale wave transformation value

一维灰度波动变换的关键在于求取一维灰度曲线上的波峰和波谷. 一般而言, 灰度波动曲线不可能如图 5 中的曲线那样平滑, 而是大量不同幅度波动的随机组合, 如图 6 所示, 一个大尺度的波动实际上由许多小尺度的波动组合而成, 大波动是众多小波动的宏观形式. 而需要定位的大波动的波峰点与波谷点正隐含于大量小波动的波峰点和波谷点之中. 因此为了控制变换对灰度波动幅度的敏感程度, 需要预先设定波动幅度阈值  $\alpha$ . 只有幅度  $h$  大于  $\alpha$  的波动才进行变换, 小于  $\alpha$  的波动则被忽略.

#### 2) 一维灰度波动变换步骤

**步骤 1.** 搜索灰度曲线首个波峰  $P_0$  或波谷  $T_0$

首先需要找出第一个波峰或波谷, 以此作为初始条件来启动迭代搜索过程. 假设图 5 中灰度曲线函数  $g(k)$  上的所有待求取的波峰点为  $P_1, P_2, \dots, P_R, R$  为波峰总数, 待求取的波谷点为  $T_1, T_2, \dots, T_S, S$  为波谷总数. 则第一个波峰或波谷可按如下方法求得: 对于  $i = 0, 1, 2, \dots$ , 设

$$\begin{cases} g(k_{\text{peak}}) = \max\{g(k_0), g(k_1), \dots, g(k_i)\} \\ g(k_{\text{trough}}) = \min\{g(k_0), g(k_1), \dots, g(k_i)\} \end{cases}$$

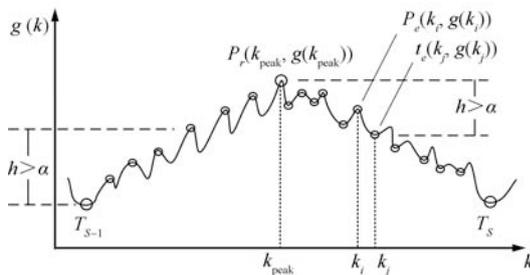


图 6 搜索波峰的过程

Fig. 6 The searching for peak

如果遍历到  $i = i_0$ , 使得:

$$g(k_{\text{peak}}) - g(k_{\text{trough}}) > \alpha \quad (15)$$

那么若  $k_{\text{peak}} < k_{\text{trough}}$ , 则  $P_0 = P(k_{\text{peak}}, g(k_{\text{peak}}))$  为第一个波峰点, 否则  $T_0 = T(k_{\text{trough}}, g(k_{\text{trough}}))$  为第一个波谷点.

**步骤 2.** 迭代搜索曲线的所有波峰和波谷

不妨以搜索图 6 中的波峰为例来说明搜索波峰的过程. 设上一轮的搜索已经成功定位了波谷点  $T_{s-1}$ , 那么在  $T_{s-1}$  与下一个未知的波谷点  $T_s$  之间必然存在许多局部极大值点和局部极小值点组成的极值点对  $(p_e, t_e)$ ,  $e = 1, 2, \dots, E$ ,  $E$  为  $T_{s-1}$  与  $T_s$  间的极值点对的总数. 所以该轮搜索需要定位的波峰点  $P_r$  就必然满足  $P_r \in \{p_1, p_2, \dots, p_E\}$ . 因此  $P_r$  可按如下方法获取: 如果存在  $k_i, k_j$  使得极值点对  $(p_e(k_i, g(k_i)), t_e(k_j, g(k_j)))$  同时满足:

$$\begin{cases} g(k_{\text{peak}}) = \max\{g(k_0), g(k_1), \dots, g(k_i)\} \\ h = g(k_{\text{peak}}) - g(k_j) > \alpha \end{cases} \quad (16)$$

则点  $P_r(k_{\text{peak}}, g(k_{\text{peak}}))$  为波峰点. 式 (16) 表示, 只要搜索出任何波动幅度大于  $\alpha$  的下降沿, 就将当前的波峰认为是符合条件的波峰, 而波峰的高度并不规定上限. 这种搜索策略不仅能够保证获取最大高度的波峰, 而且确保了下一个待求波谷的深度大于  $\alpha$ . 所以搜索波峰或波谷的过程可以看作是搜索大于  $\alpha$  的最高上升沿或最深下降沿的过程.

**步骤 3.** 对灰度曲线进行变换

灰度曲线函数  $g(k)$  上所有波谷  $\{T_1, T_2, \dots, T_S\}$  和波峰  $\{P_1, P_2, \dots, P_R\}$  全部搜索完毕后, 就可以对  $g(k)$  上所有像素点作灰度波动变换. 设  $T_s(k_{\text{trough}}, g(k_{\text{trough}}))$  和  $P_r(k_{\text{peak}}, g(k_{\text{peak}}))$  相邻且  $k_{\text{peak}} > k_{\text{trough}}$ , 那么  $[k_{\text{trough}}, k_{\text{peak}}]$  为上升沿区间. 则对  $k \in [k_{\text{trough}}, k_{\text{peak}}]$ , 点  $(k, g(k))$  就可按式 (14) 计算出它相对于波峰的变换值  $G(k)$ . 对于下降沿区

间的处理与上升沿一致. 首个波峰或波谷之前的曲线起始部分以及最后波峰或波谷之后的曲线末尾部分, 将与相邻的上升沿或下降沿合并处理. 至此对  $g(k)$  的一维灰度波动变换就全部完成. 最后将变换结果  $G(k)$  还原为图像原始坐标空间下的  $F_d(x, y)$ .

如果某一条截线未含有任何幅度大于  $\alpha$  的波峰, 则说明该条截线灰度变化比较平缓, 在工业检测图像中一般属于背景, 因此将该截线所有像素的变换值全部设为与背景色一致. 背景色参数为布尔变量, 需预先设定. 如果背景比前景灰度低, 则将背景色设为 0, 反之设为 255. 对同一类工业检测图像来说, 背景色一般是固定不变的, 因此该参数的设置并未给实际操作带来较大难度.

另外, 本文将  $L$  设为与原始图像的灰度级数一致, 即  $L = 256$ , 以尽可能多地保留灰度波动细节.

3) 阈值  $\alpha$  的选取依据

对于阈值  $\alpha$  的设定, 可以根据整幅图像中目标和背景间的灰度差异分布情况进行适当调整, 但是不能大于目标和背景间的最小灰度差, 否则将丢失目标信息. 而  $\alpha$  也不宜取值过小, 否则可能会将噪声保留下来. 因此合适的  $\alpha$  值需要根据不同实际条件下的分割需要来灵活选取, 如果需要将较弱的灰度波动呈现出来, 可以减小  $\alpha$ ; 如果希望保留较强波动, 滤除弱波动, 可以增大  $\alpha$ . 这其实也提高了该方法的适应性. 而且由  $\alpha$  的定义可知, 除了目标和背景十分接近的苛刻情况, 只要  $\alpha$  的取值在目标类与背景类的最小灰度差之内, 均可取得相近的分割效果, 因此本算法对  $\alpha$  的敏感性并不高.

## 2 灰度波动变换图像的 Otsu 阈值分割法

对  $d_1$  至  $d_8$  各方向的变换全部完成后, 可生成由八维向量构成的灰度波动变换矩阵  $F$ . 灰度波动变换矩阵缩小了原始图像不同局部之间的基础光照差异, 将像素统一为仅与邻域波动位置相关的相对性向量. 尽管改变了灰度的表现形式, 但是像素并没有与其原始灰度完全脱离, 因此可以考虑利用 Otsu 全局自适应算法<sup>[11]</sup> 在变换矩阵  $F$  上选取一个最佳的分割阈值, 并以此对  $F$  中的像素进行分类.

### 2.1 利用主成分分析法获取灰度波动变换图像

如果直接对像素的八维波动向量进行高维 Otsu 阈值分割, 计算复杂度将高达  $O(L^8)$ , 无法实际使用. 为了简化计算, 同时又在最大程度上保证分割效果, 本文采用主成分分析 (Principal component analysis, PCA) 方法<sup>[13]</sup> 将八维向量  $\{w_1, w_2, \dots, w_8\}$  压缩至一维  $\{w'\}$ . PCA 方法是在最小均方意义下寻找最代表原始数据的投影, 从而达到对特征空间的降维. 首先将  $F$  的每个元素作为样本, 构成八维向

量的样本矩阵  $X = \{x_1, x_2, \dots, x_N\}$ ,  $N$  为像素总数. 然后计算  $X$  的均值向量  $\mu$ , 并用  $\mu$  中心化  $X$ :

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i, \hat{X} = X - \mu I_{1 \times N} \quad (17)$$

再计算  $\hat{X}$  的协方差矩阵  $C$ , 并对  $C$  进行 SVD 变换得到特征向量矩阵  $U\{\xi_0, \xi_1, \dots, \xi_7\}$  以及特征值  $\{\lambda_0, \lambda_1, \dots, \lambda_7\}$ , 其中  $\lambda_0 > \lambda_1 > \dots > \lambda_7$ . 则

$$X' = \hat{X}^T \xi_0 \quad (18)$$

$X'$  即为压缩后的一维向量的样本矩阵, 将它转换到原始图像空间就可得到一维向量构成的灰度波动矩阵  $F'$ . 该矩阵也可以看作是图像, 称之为“灰度波动变换图像 (Grayscale transformed image, GTI)”. 从八维到一维的压缩过程虽然丢失了一定的信息, 但是降低了数据维数, 将计算复杂度减至  $O(L)$ , 以较小的损失尽可能地保留了灰度波动的原始信息, 折衷了效果和效率, 提高了算法的实用性. 图 7(a)~7(d) 给出了对图 1(a) 中 Rice 图像的  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ ,  $135^\circ$  一维变换结果图像, 其他方向与此类似, 图 7(e) 是总体灰度波动变换图像.

## 2.2 利用 Otsu 算法选取最佳波动阈值

在灰度波动变换图像  $F'$  中记  $i = w'$ ,  $N$  为像素总数,  $f_i$  为  $i$  出现的频数, 则  $i$  出现的概率  $P_i = f_i/N$ , 且  $\sum_{i=0}^{L-1} P_i = 1$ . 设背景类  $C_0$  与目标类  $C_1$  具有不同的概率分布, 且选取  $t$  为划分两类的阈值, 则按照文献 [11] 中的结论,  $C_0$  与  $C_1$  的类间方差  $\sigma_B(t)$  为

$$\sigma_B^2(t) = \frac{[u(t) - u_T \omega(t)]^2}{\omega(t)[1 - \omega(t)]} \quad (19)$$

其中,  $u_T = \sum_{i=0}^{L-1} iP_i$ ,  $u(t) = \sum_{i=0}^t iP_i$ ,  $\omega(t) = \sum_{i=0}^t P_i$ . 而最佳阈值  $t^*$  为:

$$t^* = \arg \max_{t \in [0, L-1]} \{\sigma_B^2(t)\} \quad (20)$$

将  $F'$  按  $t^*$  进行像素类别划分, 就可以获得最终的分割图像, 如图 7(f) 所示.

## 2.3 算法的时间复杂度

由于每个方向上的一维灰度波动变换均需要遍历原始图像所有像素一次, 所以本算法的一维灰度波动变换部分的时间复杂度为  $O(\psi(f_d)) = O(N)$ ,  $N$  为像素总数. 则灰度波动变换的总时间复杂度为  $O(\Psi(f)) = O(N \times 8) = O(N)$ . PCA 方法的计算复杂度也为  $O(N)$ , 而 Otsu 阈值选取算法的时间复杂度为  $O(L)$ ,  $L$  为灰度级数,  $L \ll N$ , 因此分割算法的总时间复杂度为  $O(N + N + L) = O(N)$ .

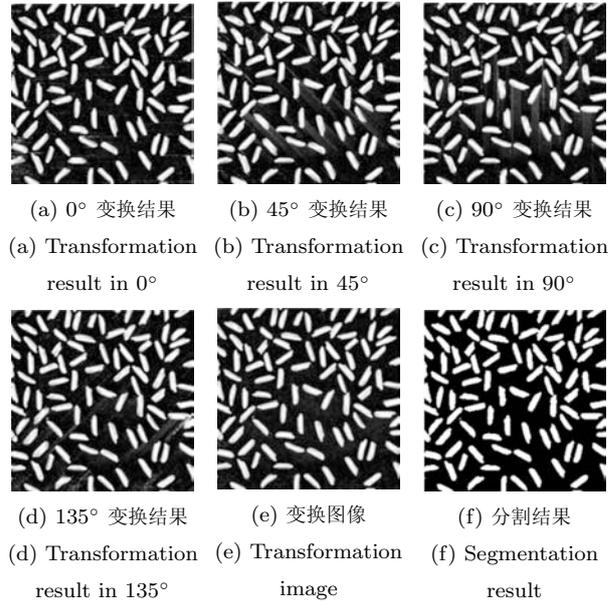


图 7 Rice 图像的  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  和  $135^\circ$  的一维变换结果图像、总体变换图像以及 Otsu 法的分割结果

Fig. 7 The 1D transformation result images of Rice image in  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ , and  $135^\circ$ , the transformed image, and its segmentation result by Otsu method

## 3 实验结果及分析

### 3.1 实验环境和测试图像库

本文采用如下仿真实验环境对算法进行测试: Intel Core2 E7200 CPU, 2G 内存, Microsoft VS2008 VC++ 编译环境, OPENCV 图像处理函数库, 编译模式为 Release 模式.

本文算法的主要处理对象是不均匀光照下的工业图像, 因此本文测试图像库的选图原则为:

- 1) 前景与背景的构成较为简单, 不受其他物体的干扰, 且二者内部的纹理质地均较为单一;
- 2) 受到不均匀光照的影响.

由于目前图像分割研究领域未能发现完全满足上述条件的公共测试库, 因此我们采用了自己构建的图像库. 由于工业检测图像的来源有限, 我们搜集了 30 幅工业检测图像, 并使用相机拍摄了 35 幅日常物体图像, 以及从网络与文献中搜集了 15 幅不均光照图像. 此外, 为了测试本文算法对文本图像的分割能力, 我们还额外选取了 10 幅不均光照文本图像. 因此图像库的总量为 90 幅图像, 其中一部分图像见图 8 第 1 列.

### 3.2 分割效果对比实验

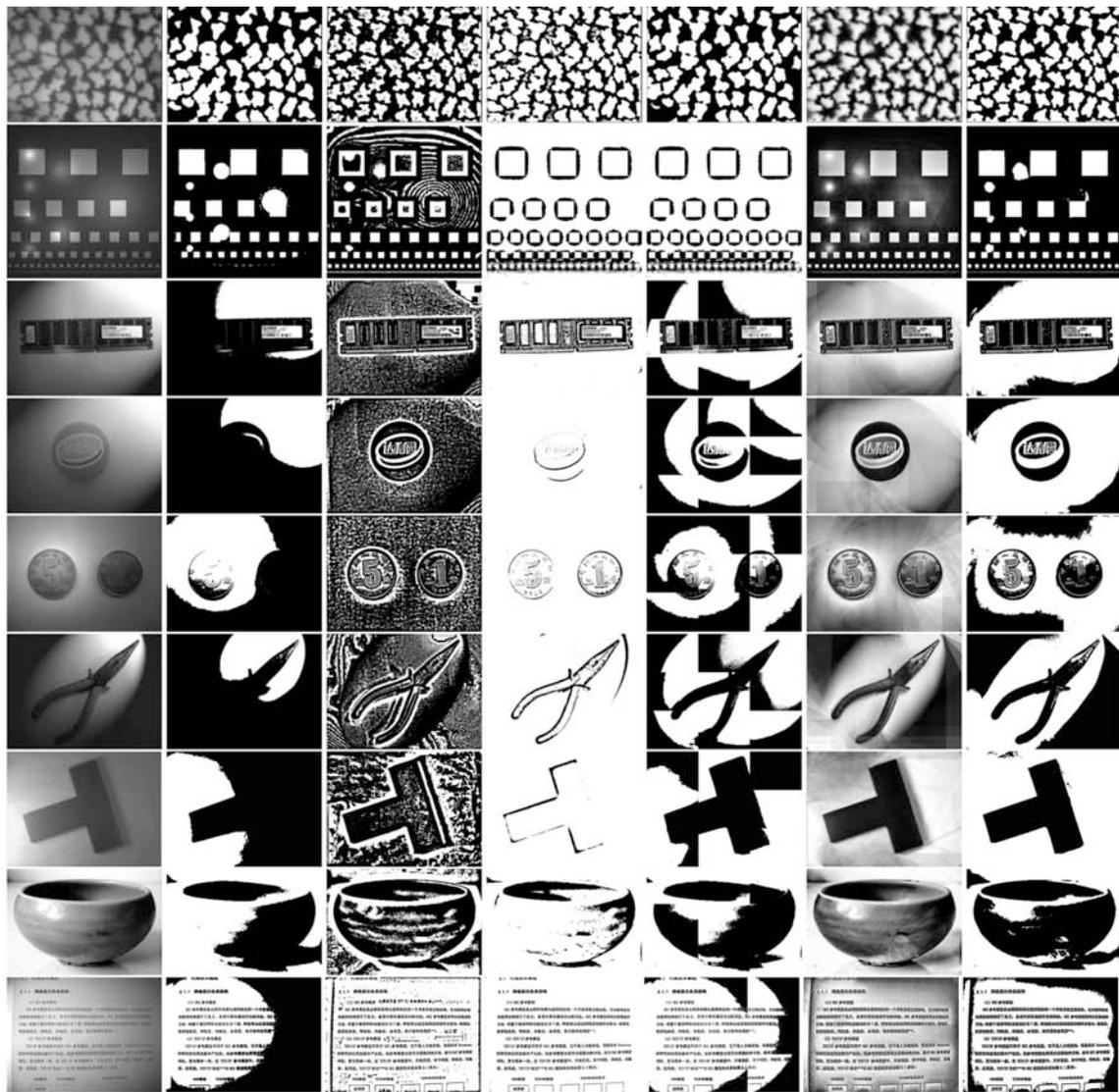
在对比算法的选择上, 我们首先选择二维 Otsu 全局算法<sup>[12]</sup> 作为全局分割算法的代表. Niblack 方法<sup>[2]</sup> 属于早期基础性的局部算法, 而 Sauvola 方法<sup>[5]</sup> 则是目前应用较为广泛的经典邻域法, 因此二

者均可作为局部邻域法的代表. Chou 方法<sup>[8]</sup>是新近提出的基于分块的局部方法, 时效性较强, 适合作为局部分块法的代表. 由于在近年发表的文献中我们还未发现有较全面的综述文章对大多数阈值分割算法进行对比和评判, 因此仅选择了上述 4 种各具代表性的算法作为本文方法的对比算法.

对比实验结果如图 8 所示, 各算法参数如下: Niblack 法中  $k$  值第 1~8 行为 0.2, 第 9 行为 -0.5. Sauvola 法中  $k$  值第 1 行为 0.05, 其余为 0.2. Niblack 法和 Sauvola 法的邻域窗口大小均为

$30 \times 30$ . Chou 法中各行的分块数均为  $3 \times 3$ , 方差阈值均为 10, 均值阈值均为 128. 本文方法中, 第 1 行和 9 行  $\alpha$  为 10, 第 2, 3, 7, 8 行为 40, 其余为 30, 第 1 行和 2 行设为暗色背景, 其余设为亮色背景.

图 8 第 1 行是化纤长丝横截面图像, 边角处有明显的暗区, 许多相邻目标块之间存有模糊的过渡区域. 二维 Otsu 法限于其全局特性, 丢失了暗区中的若干目标, 且块间粘连严重. Niblack 法和 Sauvola 法的邻域特性导致它们在前景内部误分出空洞或者在背景内部误分出噪声. Chou 法的效果比



(a) 原始图像 (b) 二维 Otsu 法 (c) Niblack 法 (d) Sauvola 法 (e) Chou 法 (f) 变换图像 (g) 本文方法  
 (a) Raw images (b) 2D Otsu method (c) Niblack's method (d) Sauvola's method (e) Chou's method (f) Transformed images (g) Our method

图 8 本文算法和二维 Otsu 法、Niblack 法、Sauvola 法以及 Chou 法的对比实验结果

Fig. 8 The comparison experiment results using various methods, including 2D Otsu method, Niblack's method, Sauvola's method, Chou's method, and our method

前三种要好,但由于  $3 \times 3$  的分块过大,块内部存在的不均光照导致 Chou 法丢失了左下角和右下角暗区内的目标块. 本文算法中,变换图像将图像各部分的光照进行了均衡,提升了目标块的亮度,因此分割效果较好,不仅暗区的目标块都能显现,块的形状也还原得较准确.

第 2 行是人工合成的方块图像,加入了数个点状光源效果. 此图像的分割上,二维 Otsu 法丢失了下方的大量小方块,中部点状光照也有明显残留. Niblack 和 Sauvola 法依然由于邻域特性导致了前景或背景中的大量误分. Chou 法比二维 Otsu 法多保留了下方暗区的一排小方块,中部偏右的一块亮斑也被抹除,但许多小方块的形状仍存在缺失. 本文方法则保留了所有的方块,方块形状较为完整,各个亮斑的残留面积也控制得较小.

第 3~8 行显示了在不同光照条件下不同物体图像的实验效果. 与前两幅图像类似,二维 Otsu 法始终缺乏对不均光照的适应力. 而 Niblack 法在背景的处理上误分太多,前景内部也遗留了大量噪声. Sauvola 法在边缘处的细节上处理得很好,背景中的阴影大部分都能消除,但是前景内部的误分非常严重,如第 3 行图像内存长方形颗粒的内部,第 6 行图像铁钳的内部,第 7 行 T 型物体的内部等. Chou 法则均出现了分块间不连续的情况,效果较差,这也说明该法对参数的依赖性很高. 本文方法对第 3~6 行图像在边角处遗留了大量阴影,但是目标周围的背景均被准确分割,目标的形状和细节也能完整显现. 对第 7 行和 8 行图像的分割,本文方法则明显比其他方法准确.

第 9 行是一幅印刷体文本图像,二维 Otsu 法仍旧丢失了右边大部分前景,Niblack 法基本未受到不均光照的影响,文本分割的较准确,但是背景中仍含有大量噪声. Sauvola 法的效果则非常优秀,不仅没有受到任何不均光照的影响,而且字体非常清晰,背景也很干净. Chou 法右侧阴暗区域的处理比二维 Otsu 法好,但是仍存在不连续现象. 本文算法由于面向波动的特性,在边角遗留了一些阴影,虽然效果比 Chou 法好,但字体笔画较粗,细节不如 Sauvola 法清晰. 总体来看, Sauvola 法对于分割文本不均光照图像较本文算法和其他算法存在明显的优势.

综上所述,对于工业检测图像,二维 Otsu 法限于其全局特性无法适用于不均光照图像的分割. 虽然邻域法对边缘细节的处理非常优秀,但限于邻域特性,Niblack 法容易在前景和背景内部造成严重误分, Sauvola 法在前景内部经常出现大面积的缺失. Chou 法依赖于分块数,方差阈值和均值阈值 3 个参数,使用时需要多次调整才能得到较满意的效果. Chou 建议采用支持向量机训练参数<sup>[8]</sup>,但这需要大

量的先验知识,实用性较低. 此外,分块间不连续的缺陷依然很难克服. 本文算法由于是面向波动的分割,无需分块,因此避免了该问题,并且参数设置也比较容易. 从实验结果上看,对于一些含有强烈阴影的图像,本文方法虽然不能完全去除边角区域的阴影,但是包含前景的关键区域均分割得较为准确,前景的形状和内部细节也都能完整地保留. 对于含有较弱不均光照的图像,本文效果则均强于其他算法.

对于文本图像,由于邻域算法对边缘较为敏感,因此更适合分割宽度在邻域范围内的狭长目标,例如文字对象,所以本文方法的效果逊于邻域法,尤其是 Sauvola 法. 此外,由于对灰度波动较为敏感,如果图像含有大量噪声,背景纹理较复杂或受到其他非前景物体的干扰,本文方法的分割效果有时并不令人满意. 图 9 是一幅背景纹理中含有大量噪声的不均光照图像,本文方法在右上角产生了许多误分.

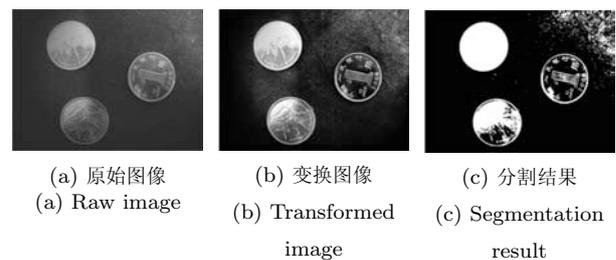


图 9 本文方法对含背景噪声图像的分割效果

Fig. 9 The segmentation result of image that contains background noise using our method

### 3.3 分割准确率和算法耗时实验

目前对图像分割质量的评判还没有一种合适的量化标准,一般依赖研究者的主观判断. 现阶段文献中使用较多的标准是误分错误率 (Misclassification error, ME)<sup>[14]</sup>,其数值按下式计算:

$$ME = 1 - \frac{|B_O \cap B_T| + |F_O \cap F_T|}{|B_O + F_O|} \quad (21)$$

其中,  $B_O$  与  $F_O$  指图像标准二值掩码 (Ground truth, GT) 中的背景与前景,该掩码图像是按照研究人员的观察标准人工制作而成,图 10 给出了图 8 第 1~9 行图像的标准掩码.  $B_T$  与  $F_T$  为分割结果的背景与前景,  $|\cdot|$  为取集合元素总数操作. 即 ME 等于分割结果与掩码之间交集的补集面积占总面积的比例,该值越低说明分割效果越好. 尽管利用 ME 来衡量各算法的分割质量并不是非常准确,但是至少从侧面反映出了各算法的效果.

表 1 给出了二维 Otsu 法、Niblack 法、Sauvola 法、Chou 法和本文算法对图 8 中 9 幅图像的 ME 值及处理时间,最后一行给出了对图像库 90 幅图像

的平均 ME 值及处理时间. 可以看出, 本文算法在第 1, 2, 7, 8 行图像的 ME 值均比其他算法低, 这与图 8 中的效果一致. 而尽管第 3~6 行图像中本文算法对前景目标的形态分割较好, 但由于背景中阴影的影响, 本文算法的 ME 值比 Sauvola 法高. 对第 9 行文本图像仍是 Sauvola 法的 ME 值最低. 在图像库 ME 平均值上, 本文算法明显低于二维 Otsu 法、Niblack 法和 Sauvola 法, 略低于 Chou 法.



图 10 图 8 中 9 幅图像的标准二值掩码图像

Fig. 10 The ground truth images of the 9 images in Fig. 8

9 幅图像的分辨率分别为: 第 1 行化纤截面图像为 768 像素  $\times$  576 像素, 第 2 行方块图像为 423 像素  $\times$  422 像素, 第 7 行 T 型物体图像为 507 像素  $\times$  384 像素, 第 8 行陶碗图像为 322 像素  $\times$  244 像素, 其余图像均为 800 像素  $\times$  600 像素. 图像库平均分辨率为 772 像素  $\times$  581 像素. 从表 1 可以看出, 正如第 2.3 节所述, 本文算法耗时与图像大小成正比, 虽然明显高出二维 Otsu 法和 Chou 法, 但与 Niblack 法、Sauvola 法相比则减少许多. 从绝对耗时上看, 除了对运算速度要求非常高的场合, 本文算

法低于 1 s 的耗时还是可以从一定程度上满足实际应用的需要.

## 4 结果与讨论

本文从图像灰度曲面入手, 针对不均光照下的工业检测图像提出了一种基于多方向灰度波动变换的自适应阈值分割算法. 该算法首先依据像素在灰度波动中的位置, 将原始图像转换为由八维波动向量构成的灰度波动变换矩阵, 然后采用 PCA 算法将八维向量压缩为一维, 生成了灰度波动变换图像. 最后利用一维 Otsu 全局阈值选取算法对变换图像实施分割. 实验表明该算法能够较大程度地降低不均光照对分割工业检测图像的影响, 改善分割效果.

本算法相对于其他算法的优越性总结如下: 1) 无需对图像分块, 仅依据灰度波动来分割; 2) 参数设置较简单, 仅需设定波动幅度阈值和布尔型背景色参数; 3) 对不均光照下工业检测图像的分割效果普遍强于第 3 节列举的几种分块法和邻域法; 4) 运算速度比邻域法快.

但是该算法也存在一些不足: 1) 本文方法减弱不均光照的效果是有限度的. 例如有些图像背景中存在灰度波动大于前景目标波动的强阴影, 从而导致大量的背景误分, 如图 8 第 3~6 行图像. 2) 由于面向波动的特性, 本算法对振幅较大的高频信号比较敏感, 当噪声的波动幅度大于波动阈值时, 噪声就会被保留下来, 如图 9. 因此本算法的抗噪性较差, 只能通过中值或均值滤波预处理来去噪. 3) 算法的执行速度仍然有待提升.

所以, 寻找分割效果更为优秀, 抗噪性更强, 且不以精度为代价的快速波动向量分割方法, 是今后算法的主要改进目标. 而如果仅对某一具体类别的图像进行分割, 还可以考虑利用支持向量机或人工神

表 1 本文算法与其他分割算法的 ME 值和耗时对比

Table 1 Comparison of ME values and CPU time between our method and other methods

ME 值/耗时	二维 Otsu 法	Niblack 法	Sauvola 法	Chou 法	本文方法
化纤截面图像	0.1269/15 ms	0.0915/1 156 ms	0.0941/1 160 ms	0.0946/15 ms	0.0102/734 ms
方块图像	0.0678/16 ms	0.1537/469 ms	0.6165/468 ms	0.0339/15 ms	0.0218/250 ms
内存图像	0.5325/16 ms	0.5359/1 250 ms	0.0673/1 266 ms	0.3115/16 ms	0.1842/781 ms
瓶盖图像	0.6265/16 ms	0.6377/1 266 ms	0.0981/1 250 ms	0.4027/15 ms	0.1201/766 ms
硬币图像	0.4841/15 ms	0.6132/1 265 ms	0.0874/1 266 ms	0.4420/16 ms	0.2691/781 ms
铁钳图像	0.6867/16 ms	0.5958/1 250 ms	0.0853/1 266 ms	0.5539/16 ms	0.3893/797 ms
T 型物体图像	0.4320/15 ms	0.4644/516 ms	0.2248/515 ms	0.2512/15 ms	0.0145/265 ms
陶碗图像	0.2278/16 ms	0.2675/187 ms	0.3863/172 ms	0.1105/16 ms	0.0696/94 ms
印刷文本图像	0.4274/16 ms	0.0550/1 250 ms	0.0114/1 266 ms	0.2164/15 ms	0.2093/797 ms
图像库平均值	0.2915/15 ms	0.4097/1 219 ms	0.3089/1 222 ms	0.1977/12 ms	0.1665/736 ms

经网络进行样本学习来选取波动阈值  $\alpha$ 。另外,除了本文提到的 Otsu 算法,目前许多全局自适应分割算法也都能够适用于灰度波动变换图像,这也为本算法今后的进一步发展提供了有利的条件。

## References

- 1 Bernsen J. Dynamic thresholding of gray-level images. In: Proceedings of the 8th International Conference Pattern Recognition. Paris, France: IEEE, 1986. 1251–1255
- 2 Niblack W. *An Introduction to Digital Image Processing*. New Jersey: Prentice Hall, 1986. 115–116
- 3 Taxt T, Flynn P J, Jain A K. Segmentation of document images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1989, **11**(12): 1322–1329
- 4 Sauvola J, Pietikainen M. Adaptive document image binarization. *Pattern Recognition*, 2000, **33**(2): 225–236
- 5 Kim I J. Multi-window binarization of camera image for document recognition. In: Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition. Washington D. C., USA: IEEE, 2004. 323–327
- 6 Huang Q M, Gao W, Cai W J. Thresholding technique with adaptive window selection for uneven lighting image. *Pattern Recognition Letters*, 2005, **26**(6): 801–808
- 7 Tsai Y H. A new approach for image thresholding under uneven lighting conditions. In: Proceedings of the 6th IEEE/ACIS International Conference on Computer and Information Science. Melbourne, Australia: IEEE, 2007. 123–127
- 8 Chou C H, Lin W H, Chang F. A binarization method with learning-built rules for document images produced by cameras. *Pattern Recognition*, 2010, **43**(4): 1518–1530
- 9 Moghaddam R F, Cheriet M. A multi-scale framework for adaptive binarization of degraded document images. *Pattern Recognition*, 2010, **43**(6): 2186–2198
- 10 Chen Q, Sun Q S, Heng P A, Xia D S. A double-threshold image binarization method based on edge detector. *Pattern Recognition*, 2008, **41**(4): 1254–1267
- 11 Otsu N. A threshold selection method from gray-level histogram. *IEEE Transactions on Systems, Man, and Cybernetics*, 1979, **9**(1): 62–66
- 12 Wang Hai-Yang, Pan De-Lu, Xia De-Shen. A fast algorithm for two-dimensional Otsu adaptive threshold algorithm. *Acta Automatica Sinica*, 2007, **33**(9): 968–971  
(汪海洋, 潘德炉, 夏德深. 二维 Otsu 自适应阈值选取算法的快速实现. *自动化学报*, 2007, **33**(9): 968–971)

13 Sezgin M, Sankur B. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 2004, **13**(1): 146–168

14 Jolliffe I T. *Principal Component Analysis (Second Edition)*. New York: Springer-Verlag, 2002. 11–76



**魏 巍** 吉林大学计算机科学与技术学院博士研究生。2008 年获吉林大学计算机科学与技术学院硕士学位。主要研究方向为图像处理与模式识别。

E-mail: win10wei@163.com

(**WEI Wei** Ph. D. candidate at the College of Computer Science and Technology, Jilin University. He received his

master degree from Jilin University in 2008. His research interest covers image processing and pattern recognition.)



**申铨京** 吉林大学计算机科学与技术学院教授。1990 年在哈尔滨工业大学计算机科学与工程系获得博士学位。主要研究方向为多媒体技术、计算机图像处理、智能测量系统和光电混合系统。

E-mail: xjshen@jlu.edu.cn

(**SHEN Xuan-Jing** Professor at the College of Computer Science and Tech-

nology, Jilin University. He received his Ph. D. degree from Harbin Institute of Technology in 1990. His research interest covers multimedia technology, computer image processing, intelligent measurement system, and optical-electronic hybrid system.)



**千庆姬** 吉林大学物理学院副教授。1983 年在吉林工业大学获得学士学位。主要研究方向为多媒体技术、光电检测技术、无线传感器网络、应用光学和光谱分析。本文通信作者。

E-mail: qianqj@jlu.edu.cn

(**QIAN Qing-Ji** Associate professor at the College of Physics, Jilin University. She received her bachelor degree from Jilin University of Technology. Her research interest covers multimedia technology, photoelectricity testing technology, wireless sensor networks, applied optics, and spectrum analysis. Corresponding author of this paper.)