

利用动态链接表的二值图像目标区域分割

田 睿,郝志成,吴 川

(中国科学院长春光学精密机械与物理研究所,吉林 长春 130033)

摘 要: 为了准确、快速地对二值图像中的连通区域进行标识,分割出各个目标物体的位置和面积,提出了一种基于动态链接表的连通区域标记方法。以目标像素线段为基本处理单元,令像素扫描和连通标识同步进行,同时,建立所用连通体的动态链接表,仅需一次扫描便可根据链接关系将图像中所有目标物体分割出来。定义“中空”的 4 邻域模板,通过特定的判断运算对二值图像进行开运算滤波,一次遍历便可去除图像中所有前景和背景噪声。实验证明:动态链接表有效地解决了其他算法中重复标记的问题,对于 512×512 的图像,算法时间约为 3 ms,且该算法不受所标记物体的形状和数目影响。满足了目标分割在工程上的速度快、准确度高、鲁棒性好等要求。

关键词: 目标区域分割; 连通标记; 动态链接表; 快速开运算

中图分类号: TP394.1 **文献标志码:** A **文章编号:** 1007-2276(2011)02-0344-06

Target area segmenting using dynamic chained list in binary images

Tian Rui, Hao Zhicheng, Wu Chuan

(Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Science, Changchun 130033, China)

Abstract: In order to label the connected components fast and exactly in binary images, and segment all the targets, a new method of labeling connected components based on dynamic chained list was proposed. The line of target pixels was taken as the elementary cell. The pixel scanning and connected components labeling were processed synchronously. Simultaneously, the dynamic chained list were built. Based on the chained relation, only once pixel scanning was needed to segment all the targets. The fast open operation needed only once scanning to finish morphologic filter through a “hollow center” cross template and special judgment. Experimental results indicate that the dynamic chained list can solve the problem of labeling repeated effectively. For 512×512 images, The method needs only about 3 ms and is independent of the shape and number of the targets. Therefore, this method can meet the needs of high speed, exactness and robustness in engineering field.

Key words: target area segment; connected component labeling; dynamic chained list;
fast open operation

收稿日期:2010-05-02; 修订日期:2010-07-26

基金项目:中国科学院青年基金资助项目(O76031T070)

作者简介:田睿(1981-),男,助理研究员,博士,主要从事数字图像处理、模式识别等方面的研究。Email:tianrui_81@yahoo.com.cn

0 引言

目标区域分割是目标检测的一个重要环节,目前主要有两种思路:一种是在灰度图像的基础上进行目标分割,比如利用相位一致性检测来进行目标分割^[1],和一些学者提出的基于K-均值聚类 and 分形理论的目标分割算法^[2];而另一种是对阈值分割^[3]后的二值图像进行区域连通与分割。

二值图像作为数字图像的一种特殊类型,由于其目标和背景对比分明,轮廓清晰,从而在几何特征参数的计算中简单有效。当图像内存在多个目标物体时,其二值图像就包含多个连通区域,而且有的区域可能由几个子区域组成。因此,有必要检测每一个区域的连通情况,然后通过标记将它们分别提取出来,这个过程就是目标区域分割。实现二值图像目标区域分割通常有以下3种方法:

(1) 区域生长法 区域生长是一种根据“一致性”的标准把图像分成若干具有最大一致性的区域。基本方法^[4]是首先在需要分割的区域里找一个“种子”作为生长的起点。然后将种子像素邻域中与种子具有相似性质(某种事先定义的生长规则或相似准则)的像素附加到生长区域内。将这些像素作为新的“种子”,反复进行这一操作,直到没有满足生长规则的像素被包括进来为止。停止生长后,所得到的区域就是分割后的目标区域。

(2) 区域分裂与合并 区域分裂与合并是在开始时将图像分割成一系列任意不相交的区域,然后将它们按照一定得原则进行聚合或拆分,以达到区域分割的目的,最后的子区域中可能众多的相邻区域都具有有一致性,这时就需要采用合并操作来弥补这一缺点。这种技术可以用四叉树形式表示^[5]。

(3) 跟踪算法 二值图像中每个取值为“1”的像素 $f(n,m)$ 都被标记一个与其坐标相关的标号,如由 n, m 构成的数。然后,将扫描标记后的图像每十像素的标号改为其邻域内的最小标号。反复执行这个过程,直到不需要作标记更改为止。

以上方法都是基于扫描像素点的,而且逻辑判断多,通常还需要多次反复操作,故而算法时间长、速度慢。文中采用基于扫描像素线段的连通体检测方法,标号确定仅在检测出一条扫描像素线段时才进行,即

以目标像素线段作为基本处理单元。该方法可检测出各种复杂类型的连通体,而且避免了基于点的连通体检测方法中一些“1”像素点可能被重复扫描的现象。

1 连通域标记

1.1 线段编码

线段编码是用来存储被分割物体的一种逐行处理技术。首先对二值图像进行逐行扫描,每遇到一行中由“0”像素值到“1”像素点的跳变时,就赋予这个“1”像素点一个新的标号(标号从“1”开始,依次增加)。而该“1”像素点后面与之相连续的所有“1”像素点都赋予相同的标号,直到遇到“0”像素点或这一行扫描结束。这个过程可以用图1所示的例子来说明。如图中所示,二值图像中每行所有由连续的“1”像素组成的线段都被赋予了一个编码值“ n ”,称这样的线段为编码线段 n 。

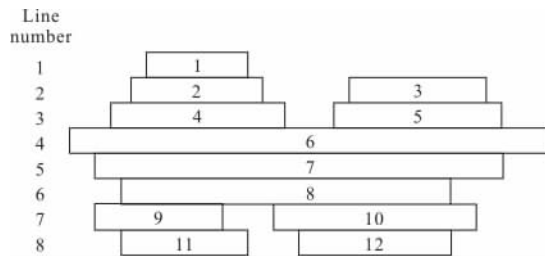


图1 线段编码原理示例

Fig.1 Coding principle of line

1.2 连通标记

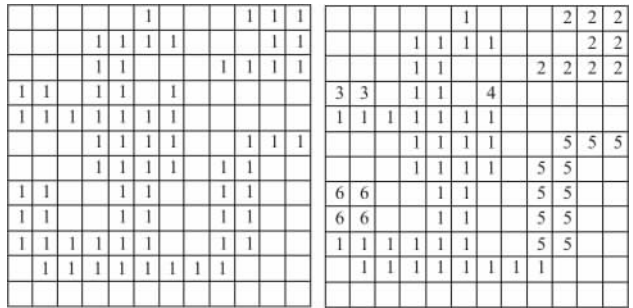
首先,设 AB 和 CD 分别为二值图像中第 i 行和第 $i+1$ 行的两目标像素线段(即线段中的像素值为“1”),其端点坐标为: $A(x_a, i), B(x_b, i), C(x_c, i+1), D(x_d, i+1)$ 。若有

$$x_a \leq x_c \leq x_b \quad \text{或} \quad x_c \leq x_a \leq x_d \quad (1)$$

则称 AB 与 CD 连通。

传统的连通标记算法是根据上述定义,对输入的二值图像进行逐行检测。当遇到第一段编码线段(即编码线段1)时,将其标记为“1”,生成连通体1(记为 CON_1);若第 $i+1$ 行的某一编码线段 n 仅与第 i 行中标记为“ k ”的编码线段连通(即与 CON_k 连通),就将编码线段 n 同样标记为“ k ”,并将其归入 CON_k 中;若编码线段 n 与多个连通体相连通,就把它归入标号最小的连通体中;若编码线段 n 与任何连通体都不连通,就赋予它一个新的标记“ $k+1$ ”,生成一个新的连通

体—— CON_{k+1} 。图 2 为按照上述方法对一幅二值图像进行标记的结果。图 2(a)是输入二值图像,其中标有“1”的方格表示二值图像中的 1, 空的方格表示二值图像中的 0;图 2(b)是标记的结果,方格中的数字表示像素所属连通体的标号。



(a) 二值图像 (b) 标记结果
(a) Binary image (b) Labeling result

图 2 连通标记结果

Fig.2 Result of connected components labeling

可以看出:实际上为同一连通体,但标记结果却包含若干不同标号的子连通体。如图 2(b)中的连通体 1, 分别由标号为{1,3,4,5,6}的多个子连通组成。所以,需要在第一次扫描像素线段的过程中,记录连通线段之间的连通关系,然后进行第二次扫描,对不同标号的连通体进行归并,以确定对应的实际连通体。但这种传统方法的执行效率非常低,尤其是对于重复性标记发生率较高的情况,效率就更低。建立动态链接表,可以将重复标记的操作融入到第一次扫描过程中,以达到一次性扫描标记的目的。

1.3 动态链接表

动态链接表定义如下:设 TAB 是一个一维数组,其数组元素 $TAB[k]$ 和数组下标 k 表示两个具有连通关系的连通体 ($i=1,2,3,\dots,N,N$ 是连通体总数目)。且 TAB 满足以下关系:

$$\begin{cases} TAB[i]>i \\ TAB[i]>TAB[j] & i>j \text{ 且 } i\neq N \end{cases} \quad (2)$$

当连通体 CON_i 与多个连通体相连通时,则将这这几个连通体的关系在 TAB 中按照从小到大的顺序存放,例如:当与 CON_2 相连通的有 $CON_1, CON_4, CON_6, CON_7$ 4 个连通体时,有 $TAB[1]=2, TAB[2]=4, TAB[4]=6, TAB[6]=7, TAB[7]=7$ 。

链接表的赋值是在逐行扫描图像,对目标像素线段进行编码的同时完成的。以图 2(a)的二值图像为

例,建立动态链接表的过程如下:

(1) 对二值图像进行逐行扫描,依据线段编码的原理对扫描行的像素进行线段编码,并给新发现的“孤立线段”赋予新的标号,产生新的连通体。同时,初始化动态链接表 $TAB[i]=0$ 。例如,图中第一行生成的 CON_1, CON_2 ,第四行生成的 CON_3, CON_4 ,第六行生成的 CON_5 以及第九行生成的 CON_6 ,链接表如表 1 所示。

表 1 链接表

Tab.1 Chained list

i	1	2	3	4	5	6
$TAB[i]$	0	0	0	0	0	0

(2) 当检测到扫描行的某一编码线段仅与 CON_i 连通时,就将其标记为“ i ”,并归入 CON_i 。

(3) 当检测到扫描行的某一编码线段与多个连通体相连通时,将其归入标号最小的连通体。若所有这些连通体在链接表中都未曾记录过,即 $TAB=0$,则在链接表中记录下它们的连通关系。例如,图中第五行的编码线段同时与 CON_1, CON_3, CON_4 连通,就将其归入到 CON_1 ,同时建立动态链接表,如表 2 所示。

表 2 链接表

Tab.2 Chained list

i	1	2	3	4	5	6
$TAB[i]$	3	0	4	4	0	0

(4) 当两个或两个以上的连通体相连通,而且在这些连通体中至少有一个已经在链接表中记录过(即 TAB 值不为 0)时,以记录过且标号最小的连通体作为接入点,依次将这些连通体连入到链接表中(包括所有记录过的连通体)。例如,图中第十行 CON_1 与 CON_6 连通,则此时的链接表如表 3 所示。

表 3 链接表

Tab.3 Chained list

i	1	2	3	4	5	6
$TAB[i]$	3	0	4	6	0	6

当扫描到第十一行时, CON_1 与 CON_5 连通,则此时的链接表如表 4 所示。

表 4 链接表

Tab.4 Chained list

i	1	2	3	4	5	6
$TAB[i]$	3	0	4	6	0	6

在此种情况下,当记录过的连通体多于一个时,链接的结果有可能出现 $TAB[i] = TAB[j] (i < j)$,此时只要令 $TAB[i] = j$,即可使链接表满足公式(2)。经过以上4步,即可成功建立动态链接表。最后只要依据表中的链接关系,按照标号“i”从大到小的顺序反向合并,就可以实现连通标记的归并,进而达到目标区域分割的目的。用此种方法对图2(a)的二值图像进行连通域标记后如图3所示。

					1			2	2	2
			1	1	1			2	2	
			1	1				2	2	2
1	1		1	1		1				
1	1	1	1	1	1	1				
			1	1	1	1			1	1
			1	1	1	1			1	1
1	1			1	1				1	1
1	1			1	1				1	1
1	1	1	1	1	1				1	1
1	1	1	1	1	1	1	1			

图3 动态链接表连通标记结果

Fig.3 Labeling result of dynamic chained list

除了上面介绍的基于动态链接表的连通标记算法外,一些学者还提出了采用其他手段的标记算法,例如引入特殊的数据结构来组织第一次扫描产生的结果,以实现重复标记点的快速查找^[6]。采用 $2 \times J$ 阶(J 为发现的具有连通关系的标号对总数)分析表来表示连通关系^[7]。在该表中,一旦发现连通关系就需添加标号对,当不同标号的连通部分的归属情况非常复杂时,如多次嵌套或包含,则连通标号对的数目将很大,而且求映射表时还需多次进栈和出栈操作,运算将较费时;采用 $n \times n$ (n 为检测到的不同标号的连通部分数目)阶二值矩阵来描述连通关系^[8];将递归的思想引入连通标记算法中^[9]。虽然这两种方法都可以有效标记出连通区域,但缺陷是内存资源的浪费,特别是在图像较大或连通部分数目较多的情况下,矩阵将十分庞大。这样,内存资源浪费巨大,而且最终的归属关系要经过不断的行列跟踪扫描才能确定。

2 二值图像滤波

由于采集图像过程中的噪声、光照的突然变化以及摄像机光圈自动调节的结果,二值图像中不可避免地会留下一些噪声点。噪声点与运动目标区域的区别在于:运动目标区域表现为若干像素点组成的具有一

定形状的空间,而噪声点则表现为相对孤立的像素点的较小集合。根据这一特点,选择形态滤波器对二值图像进行滤波处理。

2.1 形态开运算

数学形态学中最主要的两种运算就是膨胀($A \oplus B$)和腐蚀($A \ominus B$)。几乎所有的形态分析运算都来源于这两种基本的运算。将膨胀和腐蚀两种操作按不同的顺序组合起来就得到了形态开运算和形态闭运算,其定义如下:

$$\text{开运算: } A \circ B = (A \ominus B) \oplus B$$

$$\text{闭运算: } A \cdot B = (A \oplus B) \ominus B \quad (3)$$

开运算是先做腐蚀,再做膨胀。它类似非线性低通滤波器,能够消除与结构元素相比尺寸较小的亮细节,而保持图像整体灰度值和大的亮度区域基本不变,其主要作用:(1)消除细小对象;(2)在细小粘连处分离对象;(3)在不改变形状的前提下,平滑对象的边缘。二值图像中滤波操作的噪声均为小的“亮斑”,所以开运算正符合要求。

2.2 快速开运算

通常,在二值图像 $D(x, y)$ 作形态开运算时,按照其定义,用一个结构模板 $m(x, y)$ (通常为 3×3) 先对 $D(x, y)$ 做腐蚀运算,然后再做膨胀运算。这样就需要用 $m(x, y)$ 对图像 $D(x, y)$ 作两次遍历搜索,这对分辨率较大的数字图像而言,在运算时间上是一种浪费。而文中采用的快速开运算方法,只需一次遍历就可以完成形态滤波。

设 $D(x, y)$ 是未经形态滤波的二值图像, $D'(x, y)$ 是经快速开运算后的输出图像。则快速开运算可通过以下步骤实现,如图4所示。

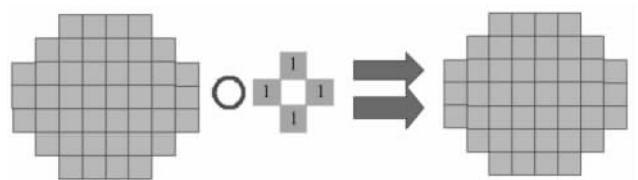


图4 快速开运算算法示意图

Fig.4 Sketch of fast open operation

(1) 定义模板。首先定义一个十字形的4邻域模板 $m(x, y)$, 令中心点为“空”, 即中心点不参与运算, 而4个邻域点为“1”, 模板形状如图5所示。

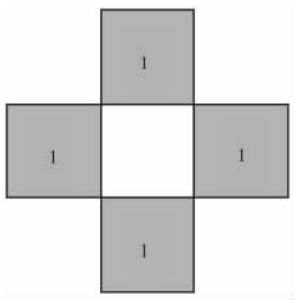


图 5 “中空”十字模板
Fig.5 Cross template with “hollow center”

(2) 遍历搜索。用模板 $m(x,y)$ 按照从上到下,从左至右的顺序去遍历搜索二值图像 $D(x,y)$:当遇到 $D(x,y)=1$ 的点时,令 $m(x,y)$ 中心点与 $D(x,y)$ 对齐, $m(x,y)$ 其他各元素逐一和 $D(x,y)$ 中的对应像素进行“与”运算。

(3) 图像输出。判断只有当 $m(x,y)$ 中 4 个“1”点与 $D(x,y)$ 中对应像素相“与”的结果都为 1 时,则输出 $D(x,y)$ 中与模板 $m(x,y)$ 对应的 5 个像素都置为 1(包括模板中心“空”点),否则都置为 0。

经过上面 3 步的遍历搜索后,得到的输出图像即为快速开运算滤波图像。与普通开运算结果不同的是:快速开运算滤波后,会使原图像中目标区的 4 个顶点被滤掉,不过这对后续的目标提取不会造成大的影响,通常可以忽略不计。

2.3 快速开运算算法分析

对于一幅分辨率为 $M \times N$ 的图像,用 4 邻域模板进行一次普通开运算滤波,在程序中共需要做 $2(M-1) \cdot (N-1)$ 次循环操作, $6(M-1)(N-1)$ 次“与”操作和 $10 \cdot (M-1)(N-1)$ 次判断操作;而快速开运算做一次滤波只需要做 $(M-1)(N-1)$ 次循环操作, $4(M-1)(N-1)$ 次“与”操作和 $4(M-1)(N-1)$ 次判断操作。所以,快速开运算的使用能大大节约计算时间,提高算法效率。

下面是普通开运算和快速开运算针对一组图片滤波运算时间的数据比较,如表 5 所示。实验平台是

表 5 开运算与快速开运算滤波时间比较

Tab.5 Comparison of filtering time between open and fast open operation

Size of image/Pixel	Time of open operation/ms	Time of fast open operation/ms
128×128	0.372	0.154
256×256	1.605	0.598
512×512	6.412	2.205

以 TMS320C6416DSP 为核心处理器的跟踪器,其工作频率为 600 MHz。

3 实验结果及讨论

3.1 目标分割效果实验

图 6 为利用文中的目标区域连通法进行目标区域分割的实际结果,其中图 6(a1)~(a4)为图像序列中不同时刻的二值差分图像,采用快速开运算对其进行了滤波处理。图 6(b1)~(b4)分别为图 6(a1)~(a4)对应时刻的目标区域分割结果(实验平台是以 TMS320C6416DSP 为核心处理器的跟踪器,其工作频率为 1 GHz)。

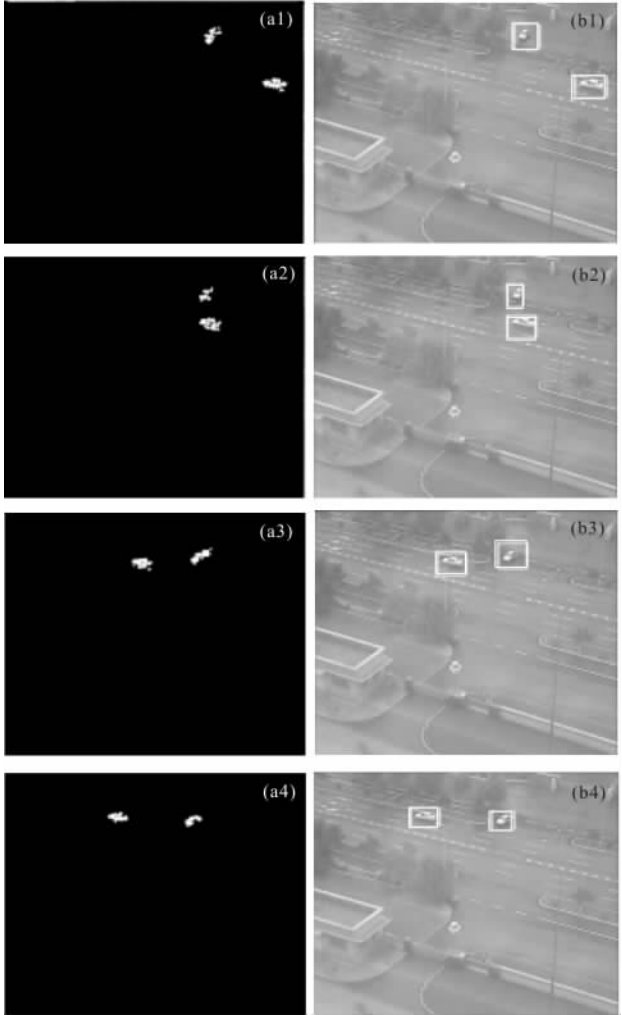


图 6 目标区域分割结果

Fig.6 Result of target area segmentation

从图 6(a1)~(a4)中可以看出,4 幅二值图像中的目标区域并不均匀连贯,不仅有许多凹凸之处,个别地方还有断裂的孤立干扰点。但是文中算法的检测结果却能有效地滤除这些干扰,并准确标记出各个目标

区域,分割出每一个目标。

3.2 算法运行时间实验

文中算法主要目的就是减少算法时间,提高实时性。为了验证这一指标,分别对3幅含有不同目标个数的二值图像采用不同的算法进行连通区域标记,图像的大小均为512×512。实验平台与实验一相同,实验结果如表6所示。

表6 4种算法运行时间比较

Tab.6 Comparison of running time among several algorithms

Target number	Region growing algorithm/ms	Region divide and union algorithm /ms	Tracking algorithm /ms	Presented algorithm /ms
1	3.203	2.685	4.203	3.014
4	10.605	7.633	14.605	3.167
8	35.412	24.368	50.412	3.289

可以看出:文中方法不仅能有效地分割出目标,而且只需执行一次像素线段扫描,其算法时间基本与图像中目标的数目和形状无关。而其他3种方法的时间都会随目标数目和形状的改变而成倍增加,特别是跟踪算法在处理小而凸的目标时,收敛速度较慢。

4 结 论

针对以往二值图像中连通体标记和目标分割算法中需要多次扫描图像和重复标记的问题^[10-12],提出一种在像素扫描过程中建立动态链接表来标记连通区域的方法。该方法以编码后的像素线段为基本处理单元,将所有已扫描过的连通体关系记录在链接表中,并在扫描过程中实时修改链接表。一次扫描结束后,便可根据链接表分割出所有的连通体。文中还定义了一种“中空”的4邻域模板,并利用特殊的判断运算实现快速开运算。实验证明:该方法能准确分割出图像中的所有目标物体,并且有较好的鲁棒性,算法复杂度和运算时间不会因为连通体形状的改变和数目的增多而增加。与快速开运算结合运用可以大大减少二值图像目标分割的算法时间,对于512×512的图像,整个算法时间(包括快速开运算滤波)仅用5ms左右,完全满足工程中实时处理的要求。

参考文献:

- [1] Hui Ying, Luo Haibo. Phase congruency based on target segmentation and tracking method [J]. *Infrared and Laser Engineering*, 2007, 6(36): 984-986. (in Chinese)
- [2] Zhang Kunhua, Yang Xuan. Segmentation extended target in complex backgrounds based on clustering and fractal [J]. *Optics and Precision Engineering*, 2009, 7(17): 1665-1671. (in Chinese)
- [3] Zhang Huaizhu, Xiang Changbo, Song Jianzhong, et al. Application of improved adaptive genetic algorithm to image segmentation in real-time [J]. *Optics and Precision Engineering*, 2008, 2(16): 333-337. (in Chinese)
- [4] Castleman K R. Digital Image Processing [M]. Beijing: Electronic Industry Press, 2002. (in Chinese)
- [5] Gonzalez R C. Digital Image Processing [M]. Beijing: Electronic Industry Press, 1998. (in Chinese)
- [6] Suzuki K, Horiba I, Sugie N. Fast connected-component labeling based on sequential local operations in the course of forward raster scans followed by backward raster scan [C]// Proceedings of 15th International Conference on Pattern Recognition, 2000: 434-437.
- [7] Li Qiang, Zhang Guilin. Algorithm for fast labeling in binary images [J]. *Chinese Journal of Computers*, 1989 (3): 238-240. (in Chinese)
- [8] Zhang Shusheng. A fast detecting approach to binary image connected components with line-based label propagation [J]. *Journal of Computer Research and Development*, 1994, 31(10): 51-54. (in Chinese)
- [9] Xu Zhengguang, Bao Donglai, Zhang Lixin. Pixel labeled algorithm based on recursive method of connecting area in binary images [J]. *Computer Engineering*, 2006, 32(24): 186-188. (in Chinese)
- [10] Sang Hongshi, Fu Yong, Zhang Tianxu, et al. A connected components labeling algorithm for multi-value images that suitable for realization in VLSI [J]. *J Huazhong Univ of Sci & Tech (Nature Science Editions)*, 2005, 33(9): 5-8. (in Chinese)
- [11] Chen Hua, Ye Dong, Chen Gang. Digital image correlation search method based on genetic algorithm [J]. *Optics and Precision Engineering*, 2007, 15(10): 1633-1637. (in Chinese)
- [12] Wang Jing. Segment labeling algorithm of connecting area in binary images and its realization [J]. *Infrared and Laser Engineering*, 2010, 39(4): 761-765. (in Chinese)