

陈小强,陈立锋,黄泓.改进的 ZS 细化算法在 FPC 缺陷定位的应用[J].湖南科技大学学报(自然科学版),2023,38(3):42-48. doi:10.13582/j.cnki.1672-9102.2023.03.006

CHEN X Q, CHEN L F, HUANG H. Application of Improved ZS Thinning Algorithm to FPC Defect location[J]. Journal of Hunan University of Science and Technology (Natural Science Edition), 2023, 38(3):42-48. doi:10.13582/j.cnki.1672-9102.2023.03.006

改进的 ZS 细化算法在 FPC 缺陷定位的应用

陈小强,陈立锋*,黄泓

(湖南科技大学 机电工程学院,湖南 湘潭 411201)

摘要:针对 ZS 细化算法中存在冗余像素,在运用到柔性电路板(Flexible Printed Circuit Board, FPC)短路和断路缺陷定位时,会出现误检和重检的情况,提出了一种改进的 ZS 细化算法.首先,对 FPC 原始图像进行通道分离和二值预处理,运用 ZS 细化算法获取初始细化图像;然后,针对 Z 字型,T 字型和十字型交点处存在的像素冗余,构造 9 个删除模板,改进经典 ZS 细化算法,消除 ZS 细化算法的冗余像素;最后,将改进的 ZS 细化算法、Hilditch 细化算法和经典的 ZS 细化算法应用到 FPC 短路和断路缺陷定位中,并进行对比.结果表明:改进的 ZS 细化算法处理时间比经典的 ZS 细化算法稍长,但只有 Hilditch 细化算法的 1/3 左右;改进的 ZS 细化算法在保证 FPC 骨架连续性的基础上,实现骨架的单一像素化,可以准确快速地定位 FPC 的短路和断路缺陷.

关键词:ZS 细化算法;删除模板;冗余像素;FPC 缺陷定位

中图分类号:TP391.41 文献标志码:A 文章编号:1672-9102(2023)03-0042-07

Application of Improved ZS Thinning Algorithm to FPC Defect Location

CHEN Xiaoqiang, CHEN Lifeng, HUANG Hong

(School of Mechanical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China)

Abstract: Aiming at the redundant pixels in ZS thinning algorithm, there will be a false detection and re-detection when applied to the short circuit and open circuit defect location of flexible printed circuit board (FPC), thus an improved ZS thinning algorithm is proposed. Firstly, the original image of FPC board is processed by channel separation and binary preprocessing, and the initial refined image is obtained by ZS thinning algorithm. Then, aiming at the pixel redundancy at the intersection of Z-shape, T-shape and cross shape, nine deletion templates are constructed to improve the classical ZS thinning algorithm and eliminate the redundant pixels of ZS thinning algorithm. Finally, the improved ZS thinning algorithm, Hilditch thinning algorithm and classical ZS thinning algorithm are applied to FPC short circuit and open circuit defect location, and comparisons are made. Results show that the processing time of the improved ZS thinning algorithm is slightly longer than that of the classical ZS thinning algorithm, but only about one-third of that of Hilditch thinning algorithm. On the basis of ensuring the continuity of FPC skeleton, the improved ZS thinning algorithm realizes the single pixel of FPC skeleton, and can accurately and quickly locate the short circuit and open circuit defects of FPC.

Keywords: ZS thinning algorithm; delete template; redundant pixel; FPC defect location

收稿日期:2021-11-28

基金项目:国家自然科学基金资助项目(61572185);湖南省教育厅重点资助项目(19A170);湖南科技大学博士科研启动基金资助项目(E57110)

*通信作者, E-mail: hmustchen@163.com

柔性电路板(Flexible Printed Circuit Board, FPC)生产过程包括冲切、钻孔、镀铜、显影、蚀刻等几十道工序^[1],其加工过程的复杂性不可避免地会产生短路和断路这2种缺陷,不仅会影响电子设备的质量,甚至还可能造成电子设备的永久性破坏.因此,FPC在使用之前必须进行缺陷检测.

目前,FPC缺陷检测方法主要有人工目测法、带电检测法和自动光学检测法这3种^[2].人工目测法需要检测人员在强光下利用肉眼或放大镜进行观察,这种方法存在主观性强、易视觉疲劳、检测效率低和准确性差等问题;带电检测法是一种接触式检测方法,将探针直接连入电路中,通过电路的导通情况来检测短路和断路缺陷,针对不同产品需要重新调整检测设备,难以满足柔性化生产的要求;自动光学检测法利用工业相机采集产品图像,采用图像处理算法来检测缺陷,是一种非接触式检测方法,具有操作简单、检测精度高、可满足柔性化生产等特点^[3].因此,自动光学检测法已成为电路板制造产业缺陷检测技术的重点研究方向之一.

检测算法是实现自动光学检测的核心,目前FPC表面缺陷检测算法有参考法、非参考法和混合法这3种^[4].参考法是通过比较待测板图像与标准板图像的差异来检测缺陷,该算法可以实现大部分缺陷类型的检测,但该方法对标准板图像与待测板图像之间的配准精度要求很高;非参考法是根据FPC板线路和缺陷特征来检测缺陷,不同类型的缺陷需要制定不同的判断准则,单一的判断准则无法适用多种类型缺陷的检测.因此,该算法复杂度较高,图像处理时间长,难以满足实时性要求;混合法综合了参考法和非参考法的特点,即对待检测图像同时使用这2套检测算法进行逻辑并行检测,该方法目前尚不成熟,检测速度慢,对系统硬件性能要求较高,无法满足工业实时检测的要求.如何有效地提高非参考法的检测效率是FPC缺陷检测算法研究热点之一.图像细化技术作为一种常用的图像处理技术,不仅可以对目标对象进行有效的描述,保留目标的轮廓和区域信息,还可以最大限度地减少目标特征的冗余信息,减少图像处理时间^[5].众多学者对细化算法进行了深入的研究,常用的细化算法有Hilditch细化算法^[6-7]、基于模板的OPTA细化算法^[8-9]、Pavlidis细化算法^[10]和ZS细化算法^[11-13]等.ZS细化算法对拐角、T型交叉点以及直线细化结果准确性高,具有连续性、细化速度快等优点,但也存在局部冗余像素、特征信息丢失等问题^[14].

不少研究学者对ZS细化算法进行了改进.如LU和WANG^[15]为了防止图像骨架的端点信息被误删除,将ZS算法中端点判决条件的范围缩小了,有效避免了二像素宽斜线的过度腐蚀,但细化结果存在残留的冗余像素,不能使图像完全细化;齐淑雯等^[16]采用先斜线再十字的方向判别方法对ZS细化算法进行改进,实现对初始细化图像的充分细化,该方法对直线特征细化具有较好的效果,但是由于逻辑较为复杂,导致细化处理时间过长;韩建峰和宋丽丽^[17]采用算术逻辑运算构造辅助判决条件,完善ZS细化算法的删除判决,抑制二像素宽斜线过度腐蚀现象,并增加二次扫描,但是并没有完全删除ZS细化算法处理后的冗余像素;贺继刚和杨晓伟^[18]为了解决ZS细化算法出现信息丢失的问题,在ZS映像算法的基础上提出了基于模板保留的ZS细化算法,减少了计算时间和部分冗余信息,但并没有彻底解决冗余信息的问题.

结合FPC线路图像尺寸大、细微特征多以及定位精度高等特点,利用经典的ZS细化算法定位FPC短路和断路缺陷,但是由于ZS细化算法存在冗余像素,会出现误检和重复定位的情况.为了去除ZS细化算法中的冗余像素点,准确快速地定位FPC短路和断路缺陷,对经典的ZS细化算法进行改进.针对Z字型、T字型和十字型这3种情况下的冗余像素,构造9个删除模板,通过逻辑运算将删除模板与ZS细化算法进行有机融合,有效减少骨架信息的冗余像素,准确定位FPC的短路和断路缺陷.

1 ZS细化算法

1.1 ZS细化算法原理

ZS细化算法是一种基于8邻域的细化算法,8邻域示意图如图1所示.ZS细化算法的对象是二值图像,其前景和背景示意图如图2所示.将二值图像的前景(灰度值为255)置1,二值图像的背景(灰度值为0)置0.ZS细化算法的思想是通过判断前景点 P_0 周围的8个像素点的情况来决定此点是否删除,即将前景像素点的灰度值变为0,从外到内逐层地删除不属于骨架的边缘像素,直到删除所有满足条件的前景像素点.

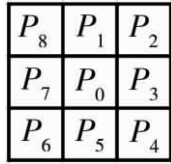


图1 8邻域

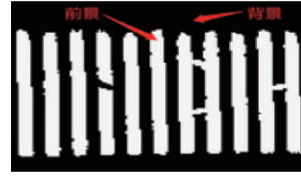


图2 前景和背景

ZS 细化算法包含 2 个子迭代过程,流程图如图 3 所示.设 N 为遍历次数,初始时 N 为 0,然后 N 进行累加,每次加 1, N 为偶数时执行偶迭代子过程, N 为奇数时则执行奇迭代子过程.2 个子迭代过程循环进行像素删除操作(将前景像素点变为背景像素点),直到 2 个子迭代过程都不存在需要删除的前景像素点.每个子迭代过程,都需要遍历整幅图像所有前景像素点,对于满足子迭代条件的前景像素点进行保存,当一次遍历过程结束后统一删除前景像素点.

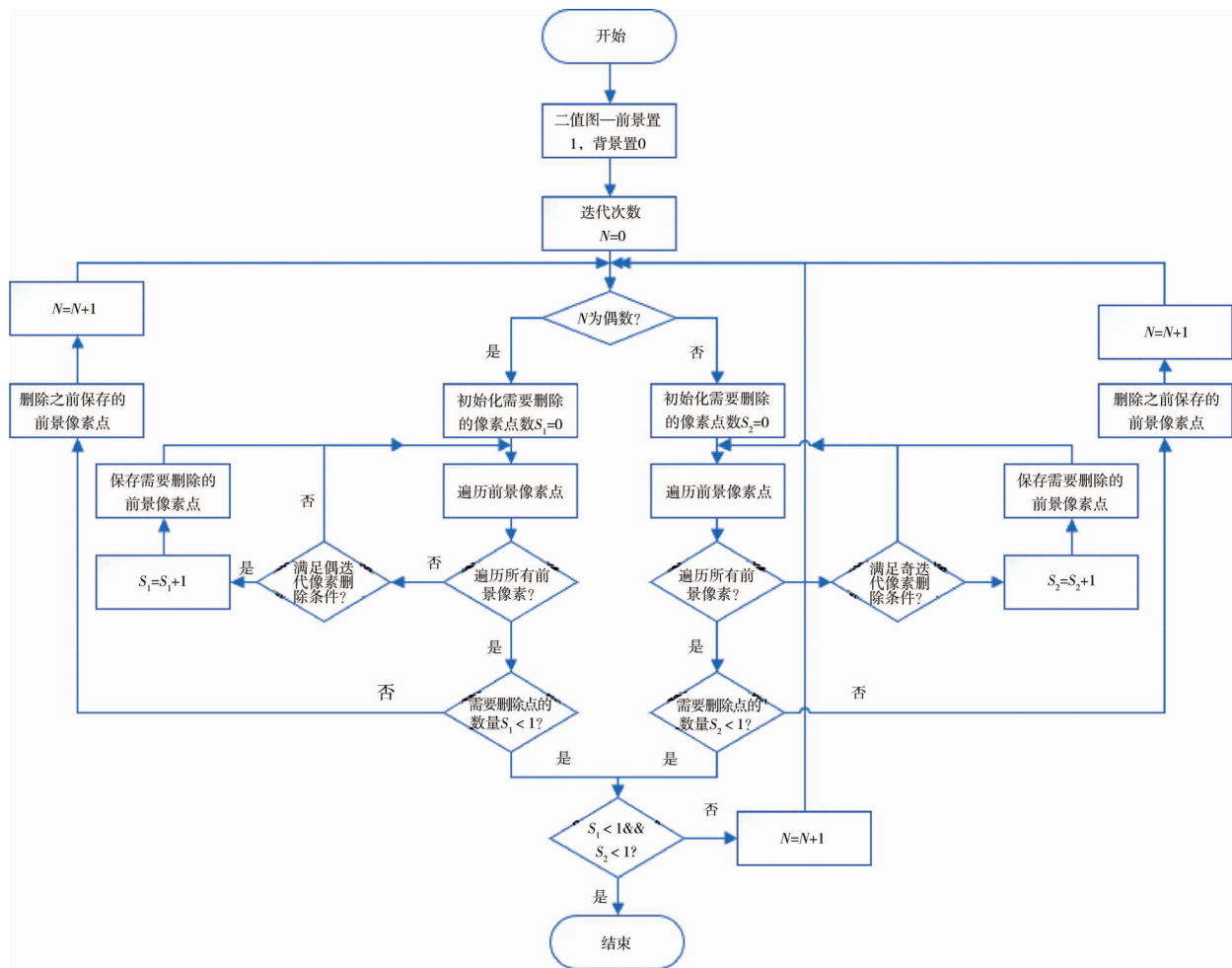


图3 ZS 细化算法流程

为了防止删除骨架的局部特征和破坏骨架的连通性,设 $A(P_0)$ 为前景像素点 P_0 的 8 邻域像素点值为 1 的个数, $A(P_0)$ 的取值范围为 0~8.若 $A(P_0)=0$,则表示 P_0 是孤立点,不可以删除,否则会删除图像的特征信息;若 $A(P_0)=1$,则表示 P_0 是端点,不可以删除,否则也会删除图像的特征信息;如果 $A(P_0)=7$ 或 8,表示 P_0 是内部点,不可以删除,否则会破坏图像的连通性.因此设置式(1)和式(2)这 2 个条件:

$$A(P_0) \geq 2; \tag{1}$$

$$A(P_0) \leq 6. \tag{2}$$

为了只删除图像的外层像素点,防止删除骨架像素,设 $B(P_0)$ 为前景像素点 P_0 的 8 邻域像素值以顺时针或者逆时针方向由 0 变 1 的次数, $B(P_0)$ 的取值范围为 0~4.若 $B(P_0)=0$,则表示 P_0 是内部点,不可

以删除;若 $B(P_0) = 2, 3$ 或 4 , 则 P_0 可能是内部像素点, 不可以删除. 因此设置式(3)条件.

$$B(P_0) = 1. \quad (3)$$

为了使获得的骨架位于图像的中心位置, 需要对称的删除非骨架像素. 在满足式(1)~式(3)这3个条件的前提下, 为了删除右下边界和左上角以及左上边界和右下角, 设置了式(4)和式(5), 以及式(6)和式(7)这2组条件. 同时满足式(4)和式(5)这2个条件, 则 $P_1 = 0$ 或 $P_7 = 0$, 或 $P_3 = 0$ 且 $P_5 = 0$, 于是删除了右下边界和左上角的非骨架像素. 同时满足式(6)和式(7)这2个条件, 则 $P_3 = 0$ 或 $P_5 = 0$, 或 $P_1 = 0$ 且 $P_7 = 0$, 于是删除了左上边界和右下角的非骨架像素.

$$P_1 P_3 P_7 = 0; \quad (4)$$

$$P_1 P_5 P_7 = 0; \quad (5)$$

$$P_1 P_3 P_5 = 0; \quad (6)$$

$$P_3 P_5 P_7 = 0. \quad (7)$$

偶迭代子过程中需要满足式(1)~式(5)这5个条件, 奇迭代过程中需要满足式(1)、式(2)、式(3)、式(6)、式(7)这5个条件.

1.2 ZS 细化算法缺点

ZS 细化算法结果如图4所示, 然后通过判断细化后骨架的节点(骨架像素点的8邻域内有3个或3个以上的骨架像素)和端点(骨架像素点的8邻域内只有1个骨架像素)位置来定位 FPC 板的短路和断路缺陷. 将细化后骨架的节点和端点位置显示在 FPC 原图上, 绿色的点代表节点, 红色的点代表端点, 如图5所示. 如果检测准确, 节点只会出现在线路的短路处, 端点只会出现在线路的端点和断路处. 由图5可以看出: 节点不仅在 FPC 线路的端点处出现, 如图5中1处所示, 还在短路处重复出现, 如图5中的2, 3, 4, 5处所示, 造成误检和重复检测的问题. 这是由于 ZS 细化后的骨架存在冗余像素, 如图4中的1, 2, 3, 4, 5处所示.

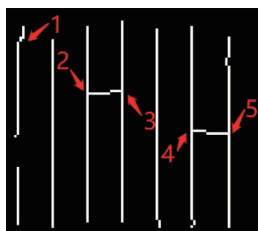


图4 ZS 细化算法结果

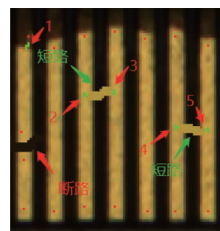


图5 ZS 细化后端点和节点分布

通过对 ZS 细化算法原理的分析, 发现 Z 字型、T 字型和十字型的交点处存在冗余像素, 如图6所示, P_2 处都为冗余像素. 冗余像素的产生是因为非骨架像素不满足式(3)的条件, 也就是 $B(P_0) = 1$, 所以没有被删除.

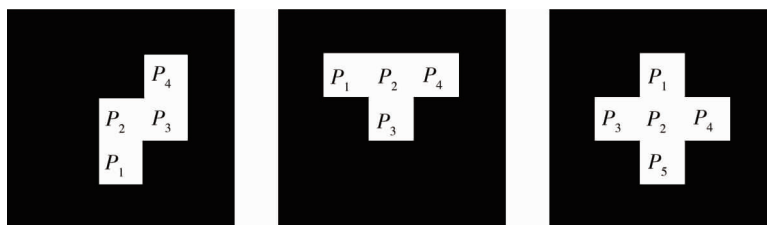


图6 Z 字型、T 字型、十字型交点处的冗余像素

2 改进的 ZS 细化算法

分析 Z 字型、T 字型和十字型这3种情况的骨架信息可知, 奇迭代和偶迭代过程中, $B(P_0)$ 的值为 $2, 3$ 或 4 时, 也存在需要删除的冗余像素. 为此, 针对 Z 字型骨架设置了4个删除模板, 如图7所示; 针对 T 字型骨架设置了4个删除模板, 如图8所示; 针对十字型骨架设置了1个删除模板, 如图9所示. 删除模板中

的1代表前景像素点,0代表背景像素点,X代表既可以是前景像素点也可以是背景像素.当前景像素点8邻域的情况与这9个删除模板中的任意一个相同时,则将此前景像素点删除,也就是将此前景像素点的灰度值置0.将9个删除模板转化为数学公式如式(8)~式(16)所示.其中, $P_1 \sim P_8$ 的取值为1或0,是前景像素点或背景像素点;&&代表前后2个等式都要成立;!代表对其进行取反,即0变为1,1变为0.

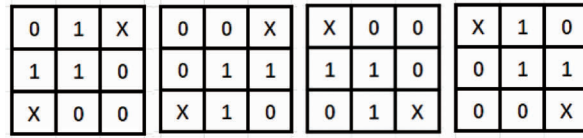


图7 Z字型删除模板

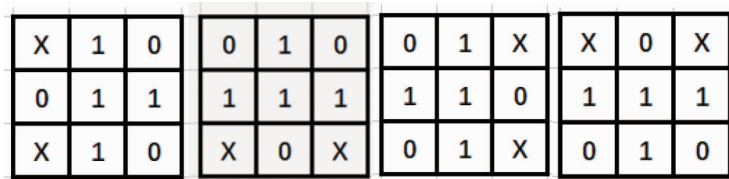


图8 T字型删除模板

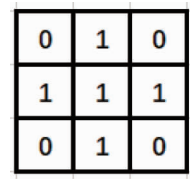


图9 十字型删除模板

$$P_1 P_7 = 1 \&\& (! P_3) (! P_4) (! P_5) (! P_8) = 1; \tag{8}$$

$$P_3 P_5 = 1 \&\& (! P_1) (! P_4) (! P_7) (! P_8) = 1; \tag{9}$$

$$P_5 P_7 = 1 \&\& (! P_1) (! P_2) (! P_3) (! P_6) = 1; \tag{10}$$

$$P_1 P_3 = 1 \&\& (! P_2) (! P_5) (! P_6) (! P_7) = 1; \tag{11}$$

$$P_1 P_3 P_5 = 1 \&\& (! P_2) (! P_4) (! P_7) = 1; \tag{12}$$

$$P_1 P_3 P_7 = 1 \&\& (! P_2) (! P_5) (! P_8) = 1; \tag{13}$$

$$P_1 P_5 P_7 = 1 \&\& (! P_3) (! P_6) (! P_8) = 1; \tag{14}$$

$$P_3 P_5 P_7 = 1 \&\& (! P_1) (! P_4) (! P_6) = 1; \tag{15}$$

$$P_1 P_3 P_5 P_7 = 1. \tag{16}$$

改进前的 ZS 细化局部放大图如图 10 所示,当加入 9 个删除模板后改进的 ZS 细化局部放大图如图 11 所示.由图 11 可以看出:改进后的 ZS 细化算法可以实现骨架的完全细化,消除了冗余像素.改进的 ZS 细化算法不仅形成了单个像素宽度的骨架,还没有造成局部信息丢失的情况.



图10 经典的 ZS 细化局部放大



图11 改进的 ZS 细化局部放大

3 几种细化算法在 FPC 缺陷定位的应用

在 Visual Studio 2017 编程环境下对 100 张 200 万像素的 FPC 彩色图像进行处理.用 Hilditch 细化算法、ZS 细化算法和改进后的 ZS 细化算法分别对 FPC 二值图像进行处理,平均处理时间、端点数和节点数如表 1 所示.细化前的二值图如图 12 所示,不同细化算法的处理结果如图 13~图 15 所示.由图 13~图 15 可以看出:Hilditch 细化后的骨架存在很多毛刺,经典的 ZS 细化后的骨架毛刺消失了,但是仍然存在非单像素宽的骨架,改进后的 ZS 细化后的骨架不仅没有毛刺存在,也不存在非单像素宽的情况.

表 1 几种细化算法对比

	处理时间/ms	端点数	节点数
Hilditch 细化算法	372	702	1 299
ZS 细化算法	75	669	141
改进后的 ZS 细化算法	118	669	29



图 12 二值图

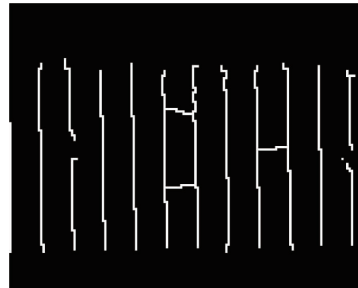


图 13 Hilditch 细化算法结果

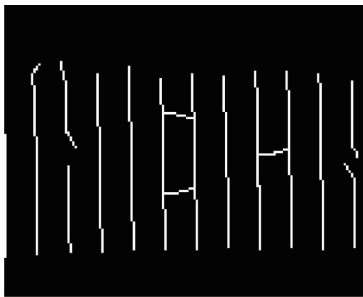


图 14 ZS 细化算法结果

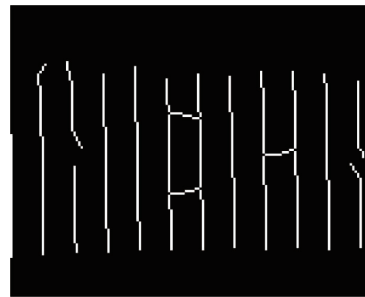


图 15 改进后的 ZS 细化结果

将 Hilditch 细化算法、ZS 细化算法和改进后的 ZS 细化算法获得的骨架的端点和节点位置显示在 FPC 线路原图上,绿色的点代表节点,红色的点代表端点,如图 16~图 18 所示.检测准确的情况下,节点只会出现在线路的短路处,端点只会出现在线路的端点和断路处.由图 16~图 18 和表 1 可以看出:Hilditch 细化算法会产生非常多的误判和重复定位情况;ZS 细化算法也会出现少量的误判和重复定位情况;改进的 ZS 细化算法没有出现误判和重复定位情况,可以准确找到 FPC 的短路和断路位置.

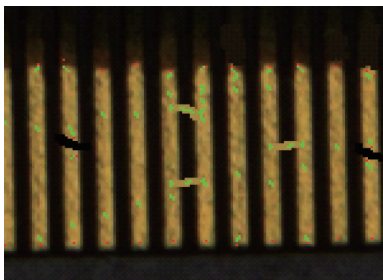


图 16 Hilditch 细化后端点和节点分布

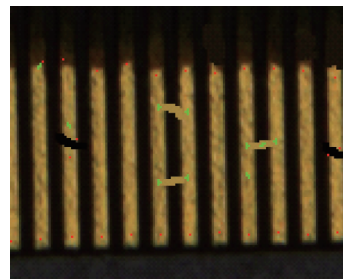


图 17 ZS 细化后端点和节点分布

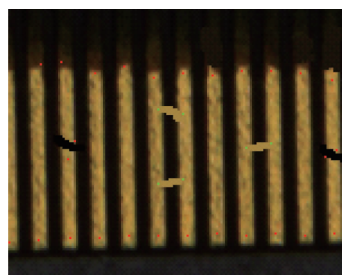


图 18 改进的 ZS 细化后端点和节点分布

对比上面3种细化算法结果可以看出:Hilditch细化算法产生了较多的毛刺,而且处理速度比较慢,处理时间是ZS细化算法的5倍左右,最重要的是存在较多的冗余像素,没有实现完全细化;经典的ZS细化结果不会产生毛刺,但是仍然存在少量的冗余像素,也没有实现骨架的完全细化;改进后的ZS细化算法由于加入了9个删除模板,虽然细化算法速度比经典的ZS细化算法速度要慢一点,但是比Hilditch细化算法速度要快很多,可以满足实际生产速度的要求.改进后的ZS细化算法不仅保留了图像原有的局部信息特征,还实现了骨架的完全细化,消除了冗余像素.

4 结论

1)改进后的ZS细化算法在处理速度上比经典的ZS细化算法稍微慢了一些,但是Hilditch细化算法的3倍左右,可以满足实际生产的高效率要求.

2)改进后的ZS细化算法不仅解决了Z字、T字和十字型交点处的冗余像素,还保留了图像原有的局部信息特征,实现了骨架的完全细化.

3)改进后的ZS细化算法可以实现FPC短路和断路缺陷的准确定位.

参考文献:

- [1] 原振方.基于机器视觉FPC缺陷检测系统的研究与应用[D].武汉:湖北工业大学,2016.
- [2] 刘晓婷.大幅度柔性印刷线路板缺陷在线视觉检测系统[D].沈阳:沈阳工业大学,2016.
- [3] 洪专婷.基于机器视觉的FPC缺陷检测的方法研究及其系统实现[D].成都:电子科技大学,2019.
- [4] 吴冬冬.基于深度学习的高密度柔性电路板缺陷检测研究[D].广州:华南理工大学,2019.
- [5] 王夏妮.汉字细化算法的研究[D].西安:西安理工大学,2016.
- [6] JIA Y, LI Y. Improving Hilditch thinning algorithms for text image[C]//International Conference on E-Learning, E-Business, Enterprise Information Systems, and E-Government-Hong Kong, IEEE. China, 2019: 76-79.
- [7] NASERI M, HEIDARI S, GHEIBI R, et al. A novel quantum binary images thinning algorithm: a quantum version of the Hilditch algorithm[J]. Optic International Journal for Eight and Electron Optics, 2017, 131: 678-686.
- [8] CHEN N, ZENG Y. An improved OPTA fingerprint thinning algorithm based on neighborhood searching [C]//2012 International Conference on Computer Science and Information Processing, 2012: 48-51.
- [9] 王家隆,郭成安.一种改进的图像模板细化算法[J].中国图象图形学报,2014,9(3):297-301.
- [10] BATAINEH B. An iterative thinning Algorithm for binary images based on sequential and parallel approaches[J]. Pattern Recognition & Image Analysis, 2018, 28(1): 34-43.
- [11] ZHANG T Y, SUEN C Y. A fast parallel algorithm for thinning digital patterns[J]. Communications of the ACM, 1984, 27: 236-239.
- [12] 张翠芳.并行细化算法的改进[J].计算机应用技术,2016,26(6):69-71.
- [13] 吴选忠.快速并行细化算法的扩展[J].福建工程学院学报,2016,4(1):89-93.
- [14] 牟少敏.一种改进的快速并行细化算法[J].微电子学与计算机,2013,30(1):53-55.
- [15] LU H E, WANG P S. A comment on a fast parallel algorithm for thinning digital patterns[J]. Communications of the ACM, 1986, 29(3): 239-242.
- [16] 齐淑雯,陈爱军,刘磊.基于ZS细化改进算法的仪表盘指针读数方法[J].中国计量大学学报,2020,31(1):44-49.
- [17] 韩建峰,宋丽丽.基于算术运算的改进细化算法[J].现代电子技术,2013,36(15):13-16.
- [18] 贺继刚,杨晓伟.基于模板保留的快速并行细化算法[J].计算机应用与软件,2017,24(12):26-29.