

自动焊机中芯片识别技术的研究^{*}

钟雪灵¹ 鲍苏苏²

(广东金融学院计算机系 广州 510520)¹ (华南师范大学计算机系 广州 510631)²

摘要 针对我国半导体生产后工序设备中的焊机一人一机手动的现状,本文提出手动焊机自动化改造的软件设计思想。讨论分析目前常用的基于灰度的快速模板匹配技术,着重研究芯片识别中所采用的关键技术——模板匹配。经实验对比发现,序贯相似检测算法(SSDA)具有较高的计算速度和定位精度,比较适合于高要求的工业应用。为进一步满足系统的要求,本文改进了自适应门限序列的 SSDA,将匹配过程分为两步进行,第一步为粗匹配,第二为精匹配,大大提高了识别的精度与速度。

关键词 自动焊机,图形识别,模板匹配,序贯相似检测算法(SSDA)

Research of Chip Recognition in the Automatic Bonder

ZHONG Xue-Ling¹ BAO Su-Su²

(Department of Computer Science, Guangdong University of Finance, Guangzhou 510520)¹

(Department of Computer Science, The South China Normal University, Guangzhou 510631)²

Abstract In semiconductor production, there is one person operating one machine by hands in the later process of bonder. According to this situation, in this paper, the software proposal of reconstruction is brought forward, and the key skill in the chip recognition——pattern match is especially researched. Some commonly used fast image matching algorithms base on gray matching are discussed. By comparison, Sequential Similarity Detection algorithm bears the characteristics of speedy calculation, accurate orientation and anti-jamming, which is fit for the high requirement in industry. In this project, we improve the arithmetic of SSDA and divide the matching process into two steps; the first one is the rough matching and the second one is the precise matching. This greatly improves the precision and pace in recognition.

Keywords Automatic bonder, Image recognition, Template matching, Sequential similarity detection algorithm(SSDA)

1 引言

在半导体的生产工序中,作为前工序的芯片生产过程和后工序的芯片检测和封装过程,都涉及精密工具和设备,且具有非常强的竞争力。据不完全统计,我国承担的半导体后工序,如半导体三极管、二极管、集成电路引线焊接的超声波金丝球焊接工序与超声波金丝球焊接工序,在国际上占有一半以上的份额,且打主力的是中小型企业。这些企业目前使用的超声波铝丝焊机与超声波金丝球焊机多数还靠手工操作,其工作效率和产品合格率都很低下。解决的办法就是实现设备的全自动化、自动化或半自动化,但是,目前全自动化设备的价格昂贵,一般每台都在 70 到 100 万元人民币,对于中小型企业而言,根本无法承受^[1~4]。

本文提出半导体生产后工序中手动焊机自动化改造的软件设计思想以及其中的关键技术。由于系统对精度和速度的要求非常高,并且引线键合现场存在各种干扰,所摄取的图像往往并不十分清楚,特征不十分明显,要准确高速识别芯片并不是件容易的事。本文重点讨论芯片识别中采用的模板匹配算法,而且根据实际情况,对传统的快速模板匹配算法进一步改进,以满足系统的高速度和高精度要求。

2 系统的软件设计^[4~6]

自动焊接系统中图像识别子系统主要是识别芯片与引脚框架,图像定位子系统主要是定位芯片及外引脚框架上焊点的精确位置,提供给焊接机构实现焊接。因此,在系统硬件确定的情况下,软件决定着识别速度和识别精度。

在精度上,焊接指标要求芯片定位精度为 $6 \pm 1 \mu\text{m}$ 。焊接速度指标要求为 4 线/s,除上芯片传送和焊头移动时间,实际焊接度为 250ms/线。主要时间分配为四部分:焊盘识别定位、焊盘焊接、线形拉线、引脚框架焊接。其中中间两项是由焊接工艺决定的,共约需 100ms,因此,要求焊盘识别定位时间不能超过 75ms/线。系统的软件设计如图 1 所示。

可见,为实现准确而可靠的焊接,需要一个精确而高速的图像识别定位软件支持,因为焊盘相对于芯片来说比较小,成像质量较差,人眼也往往不易辨别,对计算机来说判识难度更大,引脚框架同样也如此。但是,对每种型号的芯片而言焊盘相对芯片的位置是固定的,引脚框架也固定,因而可以首先由人工引导测定各个焊点在芯片坐标(以芯片中心为原点)中的坐标,并保存下来,利用模板匹配识别芯片,修正预存的焊点坐标为精确的焊点的坐标。这样既可以保证较高的识别准确率,又可以定位芯片上和对应的引脚框架上焊点的位置。

^{*} 基金项目:广东省教育厅自然科学基金(Z03021)。钟雪灵 硕士,研究方向为图形与图像处理、模式识别。鲍苏苏 博士,教授,硕士研究生导师。

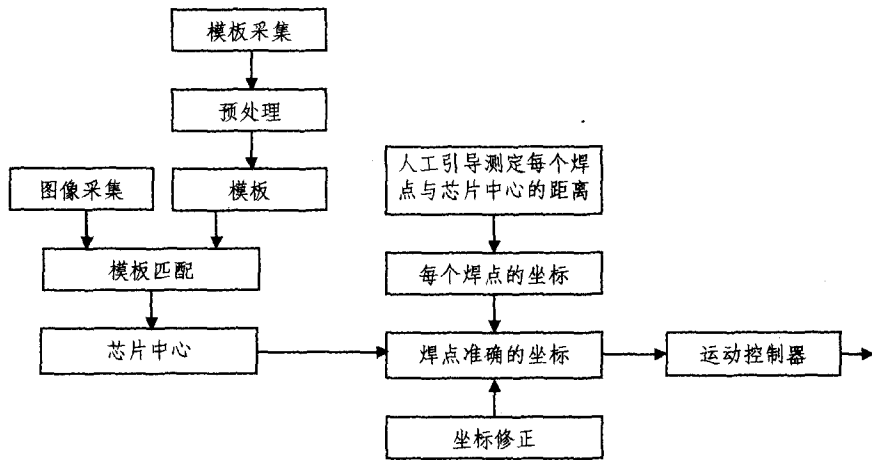


图1 软件设计示意图

3 快速模板匹配算法

模板匹配方法是一种基于灰度的匹配方法,其原理简单,易于软硬件实现,并且实时性较好,是当前研究最成熟、应用最广泛的方法之一。

通过分析发现,主要有两种途径可以减少算法的计算量:一是减小模板与场景图像的相关计算;二是减小模板在场景图像上的搜索空间。目前,较为常用的快速匹配方法有:FFT

相关算法、序贯相似检测算法(SSDA)、幅度排序相关算法、多分辨率塔型结构算法(MPSA)等^[7~9]。

表1是采用以上四种快速模板匹配算法进行芯片识别的一组实验数据,实验采用 Pentium 4 处理器,256M DDR 内存的计算机。利用 C++ Builder 实现上述快速算法。实验选取大小为 640×480 实拍芯片图像作为场景图像,芯片模板尺寸为 100×100。

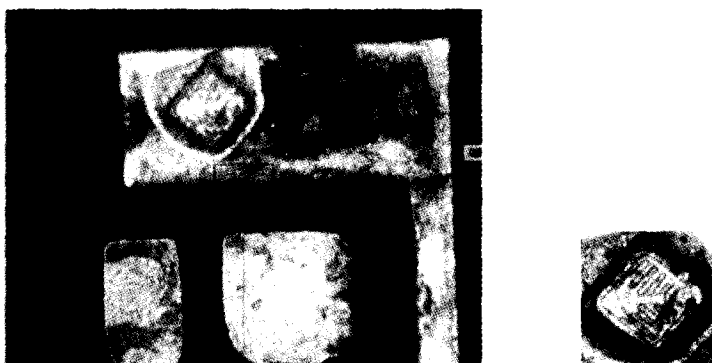


图2 场景图和模板

表1 算法比较结果

算法	原始位置	匹配位置	匹配时间(ms)
FFT	(246, 210)	(244, 213)	14438
幅度排序		(245, 210)	1827
SSDA		(246, 211)	232
MPSA		(241, 212)	914

经过多次实验,四种匹配算法中,SSDA 匹配精确最高,能满足系统对精度的要求(坐标误差±2)。需要说明的是 SSDA 算法随着目标位置的不同,每幅图像所需的时间也不相同,这里只是取平均值。此外,在速度上 SSDA 也有较大的优势,但 SSDA 仍不能满足系统对实时性的要求(250ms/线)。因此,在下一节中将改进 SSDA 算法,来满足系统对速度和精度的要求。

4 序贯相似检测算法(SSDA)^[7~12]

最简单的模板与场景图像中某一子图像对应像素之间的比较是点-点之间的距离度量,通常取为一对像素间的绝对差

$$\epsilon(u, v, j, k) = |S^{u,v}(j, k) - T(j, k)| \quad (1)$$

其中 $S^{u,v}(j, k)$ 和 $T(j, k)$ 分别表示场景图像中位置 (u, v) 的子图像和模板的第 (j, k) 个像素值,而 $j, k = 1, 2, \dots, N$ 。

$\epsilon(u, v, j, k)$ 称为误差分量。它实际上是附加噪声第 (j, k) 个分量取绝对值的结果。因为,当匹配时,附加噪声基本上是由测量噪声决定的,当不匹配时,它是由测量噪声和背景噪声组成的,所以,不匹配时的误差分量 ϵ 比匹配时的大得多。由此可见, ϵ 是一个反映匹配与否的主要指标。因此,如果把各个像素对的误差累加起来,则随着累加次数的增加将获得越来越多的信息,所以,有可能对累加和进行序贯判决,而且为了使每一次累加产生新的信息,各个像素对是按随机次序取出来进行比较的。换言之,应产生一个随机而非重复的、表示各个像素对位置的坐标序列 $(j_s, k_s), s = 1, 2, \dots, N^2$ 。

显然,当比较进行到 n 时,各次误差的累加和为:

$$E_{u,v}(n) = \sum_{s=1}^n |u,v(j_s, k_s) - T(j_s, k_s)| \quad (2)$$

其中 $E_{u,v}(n)$ 表示场景图像位置 (u, v) 上 n 次比较的累积误差,并称为累加次数。

由此可见,累积误差 $E_{u,v}(n)$ 是随着累加次数 n 的增加而

单调递增的,而且,在不匹配的位置上,由于两图不相似,随着 n 的增加而增长得很快,如图中累积误差曲线 A 和 B 所示。而在匹配位置 (u^*, v^*) 上,由于两图相似,因此 $E_{u^*, v^*}(n)$ 随着 n 的增加而增长缓慢,如图 2 中曲线 C 所示。

因此,如果选择一个简单的固定门限 T ,并规定累积误差一旦超过该门限 T 就停止计算的话,那么,在各个不匹配的位置 (u, v) 将提前结束计算,所以,在这些位置上,累加次数(记为 $n(u, v)$)就小,而在匹配位置 (u^*, v^*) 则需要经过很多次累加后, $E_{u^*, v^*}(n)$ 才能达到固定门限值 T 。因此,把记录下来的累加次数 (u, v) 为最多的那个试验位置判决为匹配位置 (u^*, v^*) 。这种匹配算法称之为采用固定门限的 SSDA 算法,因为它对于不匹配位置上的试验,只付出了较少的计算量,所以提高了相关处理速度。

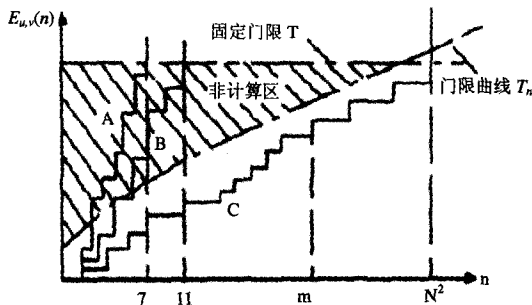


图 3 累积误差曲线和门限

如果用一个单调递增的门限序列 T_n 代替上述固定门限 T 的话,那么在各个不匹配位置上停止计算所需要的累加次数将进一步减少。因此,采用门限序列的 SSDA 算法将具有更快的处理速度,而且,门限序列曲线越靠近匹配时的累积误差曲线,则处理速度就越快。但是,由于像素的绝对误差 ϵ 本身是一个随机变量,因此,若门限序列选的太低,则会使匹配概率减小。所以,对于上述 SSDA 算法来说,如何选择一个好的门限序列,是一个很关键的问题。

累积误差是一个单调递增函数,所以门限序列也应该是单调递增的,为了尽可能减少 SSDA 算法的计算量,同时又保证它具有较高的匹配,希望所设计的门限序列适当高于匹配时的累积误差曲线。这意味着 SSDA 算法的门限序列可以根据匹配时的累积误差的均值和标准偏差推导出来。此外,还有一种自适应门限序列法,就是在匹配过程中能够自动地选择门限序列。具体方法如下:在搜索区域 G 内任选一点,计算该点处窗口图像与实测图像的误差累加和,将其作为自适应门限序列的初值。然后按任意选定的顺序选择第二点。在第二点的计算过程中,如果累加和超过了自适应门限的初值,则中止计算。并在下一个点的累加和计算过程中,仍采用原来的自适应门限值;如果所有 N^2 个点的累和仍小于原来的自适应门限值,则用这个误差累加和来代替原来的自适应门限值。这样一直进行下去,直到自动地计算出门限序列。自适应门限对于某一基准点的累加运算来说,它是不变的,但对于不同的基准点,它所取的门限则是变的。即随着基准点数目的增加,门限值递减。自适应门限的一个最大优点是具有与穷举搜索的平均绝对差算法相同的匹配定位精度,同时运算量又小得多。

采用自适应门限序列的序贯相似性检测算法,其运算量的大小与搜索的顺序有关,并且是随机的,尤其取决于初始搜索点的选择。一般来说,如果初始点十分接近配准点,则运算

量较少;反之,若初始搜索点选择不当,则运算量就较大。

5 本系统采用的匹配算法^[14]

5.1 自适应门限序列的 SSDA 算法改进

序贯相似度检测算法(SSDA)通常有三种:固定门限的 SSDA 算法、单调递增门限序列的 SSDA 算法及自适应门限序列的 SSDA 算法。对于固定门限法和单调递增门限序列法来说,门限的选取十分困难。门限选取得好,处理速度快、匹配准确。门限选取过大,处理速度提不高;选取过小,匹配又不准确。因此在实际应用中大都采用自适应门限序列的 SSDA 算法。

但在实际应用中发现,自适应门限序列的 SSDA 算法仍不能满足某些对实时性要求的应用场合,因此文中对其作了进一步的改进,在保持算法准确性的同时还极大提高了算法的实时性。

本算法将自适应门限序列的 SSDA 算法的匹配过程分为两步进行,第一步为粗匹配,第二为精匹配。

(1)粗匹配过程 在此过程中要首先确定粗匹配门限初值,利用式(2)计算 $E_{0,0}$,在计算过程中选取模板和场景图像隔 a 行和隔 b 列的像素参与计算。接下来在场景图像 S 的每一个像素 (u, v) 上利用公式(2)计算 $E_{u,v}$ 。门限 T 的初始值为 $E_{0,0}$,如果在计算像素 (u, v) 的 $E_{u,v}$ 过程中, $E_{u,v}$ 累加超过门限 T 时停止计算像素 (u, v) ,转而计算下一像素点;如果计算完像素 (u, v) 的 $E_{u,v}$, $E_{u,v}$ 不大于门限 T 时更新门限 T 并且将此像素的位置坐标赋予相关匹配点,门限 T 按下述公式取值。

$$T = \begin{cases} T & T > E_{u,v} \\ E_{u,v} & T \leq E_{u,v} \end{cases} \quad (3)$$

这样在遍历完实测图像 $E_{u,v}$ 后,即可找到粗相关匹配点,所要注意的是在计算过程中应选取场景图像和模板图像中隔 a 行隔 b 列的像素参与计算。

(2)精匹配过程 在此过程中要首先确定精匹配门限初值。假定粗匹配过程得到的粗相关匹配点为 CPoint,利用公式(2)计算 CPoint 点的 $E_{u,v}$,并以此 $E_{u,v}$ 作为精匹配门限初值。

接下来在以粗匹配点为矩心,尺寸为 $2a \times 2b$ 的精相关匹配区域内对上一步得到的粗相关匹配点作修正。修正方法如下:根据公式(2)计算精相关匹配区域内每一像素的 $E_{u,v}$,并按上一步叙述的方法和式(3)终止计算和更新精匹配门限值。

这样在遍历完精相关匹配区域后即可最终给出精确匹配点。所要注意的是在按式(2)计算过程中场景图像和模板图像中每一个像素都参与计算。

5.2 实验结果及分析

实验采用 Pentium 4 处理器,256M DDR 内存的计算机。利用 C++ Builder 实现上述快速算法。实验选取大小为 640×480 实拍芯片图像作为场景图像,芯片模板尺寸为 100×100 。取 $a=8, b=32$ 。表 2 所示为其中一组实验结果。

表 2 实验结果

序号	原始位置	匹配位置	匹配时间(ms)
1	(243, 213)	(244, 213)	21.2
2	(146, -182)	(146, -182)	23.2
3	(-124, 256)	(-125, 256)	19.8
4	(-219, -265)	(-218, -266)	19.7

(下转第 284 页)

- 9 胡海洋,马晓星,陶先平,等. 反射中间件的研究与进展. 计算机学报,2005,28(9): 1407~1420
- 10 Hopcroft J E, Ullman J D. Introduction to Automata Theory, Languages, and Computation. Wesley Publishing Company,1979
- 11 Blair G S, Coulson G, Robin P, et al. An architecture for next generation middleware. In: Proceedings of the Middleware'98, LakeDistrict, UK,1998, 191~206
- 12 Coulson G, Blair G S, Clarke M, et al. The design of a highly configurable and reconfigurable middleware platform. Journal of ACM Distributed Computing,2002,15(2): 109~126
- 13 Fabio K, Fabio C, Gordon B, et al. The case for reflective middleware. Communications of the ACM,2002,45(6): 33~38
- 14 潘爱民,徐辉. Linux 平台下的组件对象模型及其实现. 小型微型计算机系统,2004,25(9):1707~1711
- 15 Chan T S, Chuang Siu-Nam. MobiPADS: A Reflective Middleware for Context-Aware Mobile Computing. IEEE Transactions on Software Engineering, 2003,29(12): 1072~1085
- 16 Welling G, Badrinath B. An Architecture for Exporting Environment Awareness to Mobile Computing. IEEE Transactions on Software Engineering, 1998,24(5): 391~400
- 17 Haahr M, Cunningham R, Cahill V. Supporting CORBA Applications in a Mobile Environment (ALICE). In: 5th Int Conf. on Mobile Computing and Networking (MobiCom). ACM Press, August 1999
- 18 Joseph A D, Tauber J A, Kaashoek M F. Mobile Computing with the Rover Toolkit. IEEE Transactions on Computers, 1997, 46(3)
- 19 Angin O, Campbell A, Kounavis M, et al. The Mobiware Toolkit: Programmable Support for Adaptive Mobile Networking. Personal Communications Magazine, Special Issue on Adapting to Network and Client Variability. IEEE Computer Society Press, August 1998
- 20 Davies N, Friday A, Wade S, et al. L2imbo: A Distributed Systems Platform for Mobile Computing. ACM Mobile Networks and Applications (MONET), Special Issue on Protocols and Software Paradigms of Mobile Networks,1998,3(2)
- 21 Perkins C. Ad-hoc Networking. Addison-Wesley, Jan. 2001
- 22 黄罡,王千祥,梅宏,等. 基于软件体系结构的反射式中间件研究. 软件学报,2003,14(11):1819~1826
- 23 陶先平,吕建,马晓星. ARTeMIS: 基于 agent 的多模式协同中间件. 南京: 南京大学软件新技术国家重点实验室: [技术报告 BG2001012]. 2003
- 24 Lalini V, Mayur D, Shivajit M, et al. Design & implementation of a composable reflective middleware framework. In: Proceedings of the IEEE International Conference on Distributed Computer System, Arizona,2001. 644~653
- 25 Coulson G, Blair G S, Clarke M, et al. The design of a highly configurable and reconfigurable middleware platform. Journal of ACM Distributed Computing,2002,15(2): 109~126
- 26 Capra L, Emmerich W. CARISMA: Contextaware reflective middleware system. IEEE Transactions on Software Engineering, 2003,29(10): 929~945
- 27 Wall A, Norstr C. A Component Model for Embedded Real-Time Software Product-Lines. In: 4th IFAC Conference on Field Bus Systems and Their Applications, Nancy(France), 2001
- 28 Genßler T, Nierstrasz O, Schönhage B. Components for Embedded Software The PECOS Approach. In: Proc International Conference on Compilers, Architectures and Synthesis for Embedded Systems,2002
- 29 Vandewoude Y, Berbers Y. Run-time evolution for embedded component-oriented systems. In: Werner B, ed. Proceedings International Conference on Software Maintenance, IEEE Computer Society,2002. 242~245
- 30 van Ommering R. Building Product Populations with Software Components. International Conference on Software Engineering (ICSE2002), Orlando, Florida, US, May2002
- 31 Clarke M, Blair G S, Coulson G, et al. An Efficient Component Model for the Construction of Adaptive Middleware. In: IFIP/ACM International Conference on Distributed Systems Platforms (Middleware'2001), Heidelberg, Germany, November 2001. 160~178

(上接第 248 页)

本算法具有较强的抗干扰能力,这是因为干扰在图像中表现为孤立的像素点,而本算法利用到大量的像素点的空间排列及统计信息,所以孤立的像素点对本算法结果影响不大,故本算法具有较强的抗干扰能力。甚至在背景较干净条件下,可以不用作滤波前处理。

改进后的 SSDA 算法,进一步提高匹配精度,并且大大减少了算法的运算时间,完全能满足系统对精度与速度的要求。同时,算法也具有较强的抗干扰能力。

结束语 半导体后工序生产设备是非常复杂的机电一体化系统。除借鉴国外已有的技术外,必须自主研究和解决其中的关键技术问题。本文着重研究芯片识别中所采用的关键技术——模板匹配,讨论分析目前常用的基于灰度的快速模板匹配技术,即 FFT 相关算法、序贯相似检测算法(SSDA)、幅度排序相关算法、多分辨率型结构算法。本文改进了自适应门限序列的 SSDA 算法,将匹配过程分为两步进行,第一步为粗匹配,第二为精匹配,大大提高了定位的精度与速度。经过改造而成的自动焊机,可以做到一人多机,在生产效率上有较大幅度的提高。

参考文献

- 1 胡跃明,等. 半导体生产后工序设备中的若干视觉检测与伺服系统设计技术[J]. 广东自动化与信息工程,2003,24(1):7~9

- 2 王延凤,何惠阳,孙宝玉,等. 全自动金丝球焊机的 CAD/CAE 设计研究[J]. 光学精密工程,2002,10(5):466~470
- 3 鲍苏苏,等. 多步进电机控制算法研究[J]. 东华大学学报,2003,29(5):83~85
- 4 刘立峰,汤建华. 全自动金丝球焊机焊接定位系统设计研究[J]. 光学精密工程,2002,10(6):70~74
- 5 姜凯. 全自动金丝球焊机中芯片图像实时定位方法研究[D]:[中国科学院硕士研究生论文]. 2003
- 6 王耀南,等. 计算机图像处理与识别技术[M]. 高等教育出版社,2001
- 7 何斌. Visual C++ 数字图像处理[M]. 人民邮电出版社,2001
- 8 Banks S. Signal Processing, Image Processing and Pattern Recognition[M]. Englewood Cliffs, NJ: Prentice Hall, 1996
- 9 Cha Sung-Hyuk. Efficient Algorithms for Image Template and Dictionary Matching. Journal of Mathematical Imaging and Vision [J]. 2000,12:81~90
- 10 Castleman K R. Digital Image Processing[M]. Prentice Hall,1996
- 11 郑南宁. 计算机视觉与模式识别[M]. 国防工业出版社,1998
- 12 贾云得. 机器视觉[M]. 科学出版社,2002
- 13 Zhang Tianxu, Sang Nong, Wang Guoyou, et al. An effective method for identifying small objects on a complicated background [J]. Artificial Intelligence in Engineering, 1996(10)
- 14 米长为,刘晓利,徐明友,等. 基于 SSDA 的改进算法[J]. 弹箭与制导学报,2004,24(1):85~87
- 15 李俊山,等. 图像匹配技术研究[J]. 微电子学与计算机,2002(2)