

基于种子段的方向无关的直线矢量化方法*

宋继强 苏丰 陈冀兵 蔡士杰

(南京大学计算机软件新技术国家重点实验室 南京 210093)

(南京大学计算机科学与技术系 南京 210093)

E-mail: sjcai@netra.nju.edu.cn

摘要 准确而高效的矢量化是工程图纸理解的基础,现有方法普遍存在3个不足:(1)通过中心点链拟合生成线段的计算量大;(2)相交或噪音的影响将一条完整的直线先分段矢量化,然后再拼接起来,需要大量遍历操作;(3)对斜线的处理能力明显低于对水平或垂直线的处理能力.该文提出了一种基于种子段的直线整体矢量化方法,此方法可以一次性识别一条完整的直线,既适用于任意方向的直线,又避免了拼接所需的反复遍历操作,性能分析和实验结果均表明,该方法明显提高了直线矢量化的性能.

关键词 直线矢量化, 图纸识别, 种子段, 图像跟踪, 方向无关.

中图法分类号 TP391

准确而高效的矢量化是工程图纸理解的基础.由于直线是工程图中最基本的并且占绝对多数的图形元素,直线矢量化的质量成为评价矢量化方法的重要方面.国内外已经提出了许多直线矢量化方法^[1~5].但现有的矢量化方法普遍存在3个不足,一个是通过中心点链拟合的方法生成线段,计算量大并且对噪音敏感,二是字线粘连、线条相交等因素会引起一条完整直线的分段矢量化,而噪音、缺损等扫描位图中常见的现象造成了每一段拟合线段的方向偏离,则使线段合并等后处理变得十分困难.此外,为合并直线而反复遍历位图点和线段非常耗时,严重降低了直线矢量化的速度并增加了算法的复杂性.再一个不足是对斜线的处理能力明显低于对水平或垂直线的处理能力,使直线矢量化的质量产生方向依赖性.因此,Karl Tombré^[6]在评价现有矢量化方法时指出:“None of these methods works”.他进一步解释这个结论的含义:“Actually, the methods do work, but none of them is perfect”.因此,人们还在继续努力寻找好的直线矢量化方法^[7~11].

本文提出的基于种子段的直线整体矢量化方法(global line vectorization,简称GLV)从根本上解决了这3个问题.GLV通过寻找种子段来获得待处理直线的特征(方向、线宽),然后依据特征方向延伸种子段,使其生长为一条完整的直线,从而避免了中心点链拟合和分段矢量化.同时,由于寻找与延伸种子段的算法都与方向无关,因此本文的直线矢量化方法对任意方向的直线都具有相同的处理能力.

1 现有直线矢量化方法概述及GLV的设计思想

1.1 概述

现有的矢量化方法大致可以分为4类:基于细化的方法^[1,2,7]、基于轮廓的方法^[3]、基于邻接图的方法^[4,5,8,9]和像素跟踪方法^[10,11].

基于细化的方法.采用各种边界腐蚀操作逐层削去图像的外边界像素点,直到留下单像素宽的骨架点并将

* 作者宋继强,1974年生,博士生,主要研究领域为工程图纸扫描图像的分析 and 识别,计算机图形学,CAD.苏丰,1975年生,博士生,主要研究领域为工程图纸扫描图像的分析 and 识别,图像处理,CAD.陈冀兵,1975年生,硕士生,主要研究领域为工程图纸扫描图像的分析 and 识别,文字识别,CAD.蔡士杰,1944年生,教授,博士生导师,主要研究领域为计算机图形学,CAD,人机交互,文档分析识别.

本文通讯联系人:蔡士杰,南京210093,南京大学计算机科学与技术系

本文2000-02-28收到原稿,2000-04-18收到修改稿

其转化成链码表示,然后通过折线拟合将链码转化成矢量(直线段或折线段),这类方法的主要缺点是丢失了线宽信息和交点处的畸变,反复遍历位图和大量的短线段合并使这类方法的速度很慢。

基于轮廓的方法.先提取图像的轮廓,然后找出匹配的轮廓对,在轮廓对之间可以拟合出该段图像的中心线。这种方法虽然可以保持线宽信息,但由于在相交、粘连处无法找到轮廓对,必然会造成一条直线的分段矢量化。

基于邻接图的方法.通常是先对位图进行行程编码(run length encoding),然后分析行段生成各种形式的邻接图结构,由位图中的线状区域的行段中点拟合而成的线段作为邻接图中的边,非线状区域(相交或粘连处)作为邻接图中的节点连接相邻的边,最后在图结构上进行线段延伸、合并。这类方法难以处理实际图纸中的复杂情况,并且当图纸质量较差时对节点的分析(如出、入度计算)非常困难。此外,这类方法对行程编码的扫描方向的依赖性导致斜线矢量化的质量较差。

像素跟踪方法.没有将位图数据结构转换成其他紧凑数据结构(如链码,行程编码等)的过程,而是直接进行像素级跟踪。这类方法处理原始位图数据,因而在数据访问灵活性和形态保持上优于前几种方法,但由于现有方法在跟踪时没有明确的方向约束,无法保证正确穿越交点,所以仍然会发生分段矢量化。并且,此方法总是按水平或垂直方向扫描像素,仍然有处理斜线的难度。

以上4类方法虽然在提取和表示图像信息的方式上互相区别,但它们都按照以下4个步骤来完成矢量化一条直线的工作:(1)得到线状区域的中心点链;(2)由中心点链拟合成线段;(3)将短线段拼成长线段;(4)标记该直线的图像数据防止重用。中心点拟合、分段矢量化和方向依赖性是影响现有方法性能的主要原因。

1.2 GLV 的设计思想

GLV 的基本思想是,从无干扰的像素连通区找到待识直线的方向和线宽特征,并沿着初始方向进行像素跟踪,在跟踪中逐步调整方向,从而既可以正确地穿越线上的相交或粘连部分,又可以适用于任意方向的直线。关键问题在于如何确定待识直线的方向以及如何高效地实现任意方向的像素跟踪。

考虑到直线具有自相似性,即它的一部分在方向和线宽上与整体相似,本文提出了直线种子段的概念,用于在像素连通区中找出直线的特征。为方便定义种子段,先作如下约定:

约定 1. 矩形 $ABCD$ 的两短边中点为 E 和 F ,长边中点为 G 和 H ,中心为 O 。

约定 2. $P(R)$ 为闭区域 R 中的像素点数, $B(R)$ 为 R 中的黑像素点数, $W(R)$ 为 R 中的白像素点数。

约定 3. $V(l, i)$ 为直线 l 的路径上第 i 点处在直线的正交方向上的连通黑像素数。

定义 1. 种子段是直线的图像上能够表征其方向和宽度的一组位图,是同时满足以下 4 个条件的矩形连通区(如图 1 所示):

(1) $W(ABCD) < \epsilon$, 其中 ϵ 为线条质量阈值,取值范围为 $P(ABCD) \times 10\%$ 左右,考虑到减少浮点乘除运算,选择 $P(ABCD)/8$,可以用移位操作实现;

(2) $|B(ABEF) - B(CDFE)| < \delta \wedge |B(AGOF) - B(HDFO)| < \delta \wedge |B(GBEO) - B(CHOE)| < \delta$, δ 为中心对称阈值,小于 $\epsilon/2$;

(3) $\forall i, 1 \leq i \leq |EF| \rightarrow V(EF, i) \leq |AD| + \beta$, β 为预定的阈值,以排除噪音的干扰。

(4) $|AB| = K \times |AD|$, K 是一个实验常数,若太小,则不能保证种子段具有直线的特征;反之,则寻找种子段的开销过大,并且难以找到满足条件的种子段。表 1 列出了按不同的 K 值寻找种子段的有效性(为成功识别一条直线所需调整种子段(参见第 2.2 节)的平均次数)和开销(为成功地找到一个种子段的平均搜索长度/ $|AD|$)。通常取 $K=4$,即可在保持线状特征与保证效率之间取得平衡。

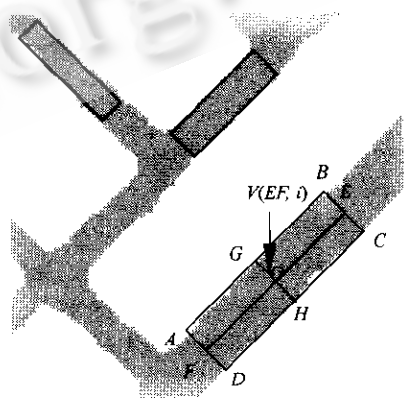


Fig. 1 Definition of seed segment

图 1 直线种子段的定义

Table 1 Experimental result of detecting a seed segment with various *K*
表 1 按不同 *K* 值寻找种子段的实验结果

<i>K</i>	2	3	4	5	6	8	10	12	16	20
Validity ^①	5.3	2.4	1.1	1.1	1.0	1.0	1.0	1.0	1.0	1.0
Cost ^②	2.8	3.7	4.8	6.9	10.2	14.3	20.6	36.4	61.5	137.7

①有效性,②开销.

种子段的长轴方向和短轴长度分别反映了待识直线的方向和线宽.种子段沿其长轴的两个方向生长成一条完整的直线,因为延伸的方向已经确定,种子段的生长不受直线上的相交或粘连的影响.为避免重复识别属于已被识别直线的图像数据,在识别出一条直线后就应立即将该直线的数据从图像数据中抽去而不影响后继的识别.按照以上设计思想,GLV 矢量化一条直线分 3 个步骤:(1) 搜索种子段;(2) 种子段生长为一条直线;(3) 擦除该直线的位图.这种处理方法完全不同于现有方法的先从中心点链拟合线段再拼接成直线的做法,并且不依赖于某一具体方向,因而可以克服现有方法的不足.

2 GLV 核心算法的实现

2.1 搜索种子段

从一个黑像素点 *c* 开始按定义寻找种子段的算法如下(如图 2 所示).

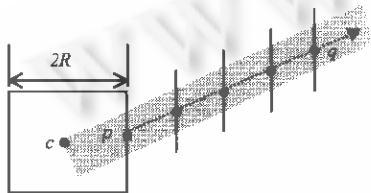


Fig. 2 Seed segment detection

图 2 搜索种子段的过程

约定 4. *r* 为搜索步长,scope 为最大搜索范围,均根据图纸扫描分辨率确定.初始设 $R=r$.

Step 1. 以黑像素点 *c* 为中心作一个边长为 $2 \times R$ 的正方形窗口,从上边界开始,按顺时针方向取窗口边界上的长度接近线宽范围的连续黑像素段为候选段.

Step 2. 若有未处理的候选段(长度为 *L*),以中点 *p* 作为初始点来寻找种子段.若无未处理的候选段,如 $R < \text{scope}$,则 $R=R+r$,

并跳至 Step 1;若 $R \geq \text{scope}$,结束寻找,返回失败标志.

Step 3. 从 *p* 开始,按 \overrightarrow{cp} 确定的方向(上、下、左或右),以 *r* 为步长测正交方向上的黑像素段长度,长度接近或小于 *L* 的黑像素段称为规则段,否则为不规则段.设 *n* 为连续测得的规则段数;*m* 为连续测得的不规则段数.设当前测得的规则段中点为 *q*,如果 $n \times r \geq K \times L$,则进入 Step 4;否则,如果 $m \times r > \text{scope}$ 或 $|cp| > \text{scope}$,则结束寻找,返回 Step 2.经过几个不规则段以后,在又进入规则段时,将 *p* 定位到 *q*,继续寻找,使搜索可以跳过小范围的噪音、缺损或交点.

Step 4. 根据种子段定义检查 *p, q* 之间的黑像素连通区是否合格.如果合格,则结束寻找,返回成功标志;否则,返回 Step 3 继续寻找.

该算法可从图像中任意一个黑像素点开始,寻找(如果搜索范围内存在)距离最近的一个任意方向直线的种子段.找到的种子段的长轴方向可作为该直线的初始方向,短轴长度可作为线宽的初值.

2.2 种子段的生长

种子段的生长由基于 Bresenham 扫描转换算法的定向跟踪技术来实现. Bresenham 算法用于快速生成任意方向的直线路径上的点的序列.跟踪路径从种子段的长轴开始,沿着长轴的方向朝两个相反的方向延伸.只要满足下面两个条件,种子段就可以生长:

条件 1. 跟踪路径上连续白点段的长度小于虚线间距的阈值;

条件 2. 跟踪路径上连续黑点段上每一点处的与路径正交方向的连通长度接近或大于种子段线宽.

条件 1 可以保证跳过小的位图缺损,并具有跟踪虚线的能力.条件 2 保证延伸的位图连通区中包含待识直线的特征,并能通过相交、粘连区域.向其中一个方向跟踪直线的过程描述如下(如图 3 所示).

约定 5. 路径段表为记录路径上连续的黑白段的链表,作用于两个方向的跟踪过程,初始化为空;*L* 和 *R* 是跟踪路径两侧与路径点连通的黑像素点的计数器,作用于一个方向的跟踪过程.

Step 1. 用 Bresenham 算法生成种子段长轴方向的正交方向路径 $V_0[i](i=0, \dots, n)$. 路径的中点在 O (种子段的中心点) 并且 $n=3 \times W$ (W 是种子段的宽度). 初始化 L 和 R 为 0.

Step 2. 按种子段的长轴方向用 Bresenham 算法逐点生成从 O 开始的跟踪路径. 每生成一个点, 检查该点对应的像素点是否为黑点. 如果是, 则进入 Step 3, 否则, 进入 Step 4.

Step 3. 累计连续黑点的长度, 如果长度大于噪音阈值, 则将上一白段 (如存在) 的长度记入路径段表. 检查在当前路径点 P 处的正交连通长度. 这时, P 点处的正交方向路径 V_P 不必重新生成, 而可以从 V_0 平移变换得到, $V_P[i]=V_0[i]+offset(i=0, \dots, n, offset=P-O)$.

从中点 $V_P[n/2]$ 开始向两侧检查 V_P 上的黑段连通长度, 如果两侧的总长度近似 W , 则将两侧的长度分别累加到 L 和 R . 当 L 与 R 的差值大于 L 与 R 中的较小值时, 表明种子段的方向与实际直线方向存在偏差, 需要在进行调整后重新跟踪. 种子段的动态调整确保种子段能按相应位图延伸到最大长度. 实际上, 不必在跟踪路径上的每一点都做正交检测, 每隔几点做一次可加速跟踪 (间隔大小取决于图纸的扫描分辨率).

Step 4. 累计连续白点的长度, 如果长度大于噪音阈值, 则将上一黑段 (如存在) 的长度记入路径段表. 如果长度大于虚线间距的阈值, 则结束当前方向上的跟踪.

整个跟踪过程以种子段的方向为参数, 没有涉及任何具体方向, 因此对任意方向的直线都具有相同的处理能力. 另一方向的跟踪过程与此类似, 只是在向路径段表中记入段的方式有所不同. 一个方向添加到表尾, 另一方向添加到表头, 并且两个方向的第 1 段要合并成一个黑段. 当向种子段的两个方向跟踪结束后, 路径段表中记录了整个延伸路径从一端到另一端的顺序的黑、白段长度. 如果只存在一个黑段, 则该段是一条实线; 否则, 通过分析黑、白段之间是否存在与虚线模式类似的规律性, 就可以实现对虚线的一次性识别. 当一条直线被完整地识别后, 准确的线宽可以由整个路径上近似于 W 的正交段长度的平均值计算得到.

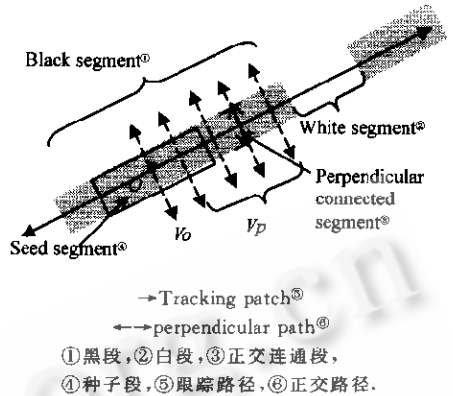


Fig. 3 Direction guided tracking of line
图 3 定向跟踪直线

2.3 擦除已识别直线的位图

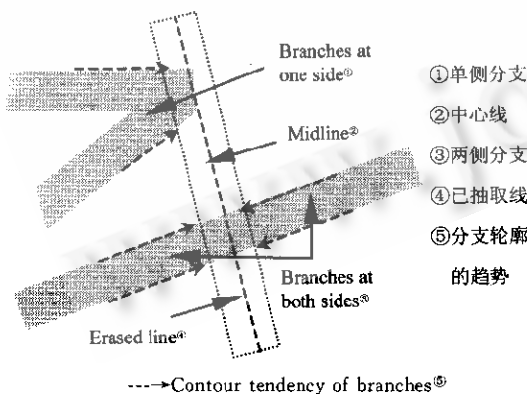


Fig. 4 Erasion of a recognized line
图 4 擦除已识别直线的示意图

一侧擦除至直线的中心线, 另一侧用轮廓的趋势计算.

擦除已识别直线的位图后, 与该线相交的直线都减少了一个交点, 从而简化了后继线条的识别.

2.4 相交直线的处理

事实上, 不必每次都通过在黑像素连通区中搜索种子段来矢量化一条直线. 直线之间的相交关系可被用来

对于一条孤立的直线段, 按线宽擦除位图的操作很容易实现. 但当直线与其他图形或文字相交时, 交点的多样性使问题复杂化. 事实上, 由于与其相交的对象尚未识别, 要准确计算交点处应该保留的部分是不实际的. 因而 GLV 采用了一个基于对交点分支位图的轮廓检测的近似算法 (如图 4 所示).

Step 1. 沿直线的路径测试路径正交方向上的连通长度, 将路径划分为有交点段和无交点段. 无交点段按线宽完全擦除; 有交点段的处理转入 Step 2.

Step 2. 检测交点处直线的中心线两侧的分支情况及其局部轮廓.

Step 3. 若为两侧分支, 则用两边轮廓的趋势计算出近似的擦除边界; 若为单侧分支, 则将无分支的

加速矢量化. 擦除已识别直线时检测了该直线各交点处的相交情况, 对于仅有一条交线的交点, 可以较容易地检测到交线的方向和线宽, 从而跳过搜索种子段的步骤, 直接进行定向跟踪直线.

3 性能对比与实验结果

直线矢量化方法的性能主要由以下几个方面来衡量: 直线参数的识别精度、对各方向直线的适应性和计算复杂度. 表 2 从理论上对 GLV 与现有方法的性能进行了分析和对比.

Table 2 Performance comparison between GLV and existing vectorization methods

表 2 GLV 与现有直线矢量化方法的性能对比

Method ^①	GLV	Existing vectorization methods ^②
Precision of line parameters ^④	High, because it detects line width and adjust midline according to the entire line ^⑤	Thinning-based methods lose line width. Precision is degraded by other methods because parameters are averaged from each segment ^⑥
Impact of line orientation ^⑦	Independent, because process lines of any orientation in a same way ^⑧	Thinning-based and contour-base methods are adaptive to line orientation. Other methods are inconvenient to process slant lines ^⑨
Computational complexity ^⑩	Low, because it is based on Bresenham algorithm which mostly uses addition, subtraction and shift operations ^⑪	High, because line fitting uses a lot of distance calculation, and combining collinear lines need a lot of list searching and angle calculation ^⑫

①方法, ②对比项目, ③现有直线矢量化方法, ④直线参数的识别精度, ⑤整条线范围测试线宽, 调整中心线, 精度较高, ⑥基于细化的方法会丢失线宽; 其他方法依据各段的参数来推算合并线的参数, 不够准确, ⑦对各方向直线的适应性, ⑧与方向无关, 对任意方向直线的处理方法完全相同, ⑨基于细化和基于轮廓的方法对方向的适应性较好; 其余方法处理斜线比处理水平、垂直线复杂得多, ⑩计算复杂度, ⑪低, 因为基于 Bresenham 算法, 主要是加、减和移位操作, ⑫高, 因为拟合直线段使用大量距离计算, 寻找和连接共线的线段需要大量链表遍历操作和角度计算.

从以上对比可以得出结论, GLV 比现有的直线矢量化方法具有明显的优点. GLV 通过种子段获得直线的方向和宽度特征, 然后经定向跟踪一次性识别整条直线, 从根本上避免了分段识别后再合并时所需的大量遍历、共线性判断与拼接操作, 简化了矢量化过程. 定向跟踪排除了粘连、交叉对识别的影响, 并通过擦除仅仅对应于已识别直线的位图的方法使图像数据逐步简化, 从而较好地解决了粘连、相交的分离难题. 因此, 其直线矢量化的速度和精度应该比现有方法有较大幅度的提高.

本文提出的基于种子段的直线矢量化方法已用 VC++6.0 在 PC 机上实现, 并通过了近百张工程图纸的扫描位图测试. 该方法对直线数量在 2 000 条左右的 A0 图纸中直线部分的自动矢量化只需 1 分钟 (CPU 为 PII233, 内存为 128M), 比基于细化、行程编码的矢量化方法的速度提高了一个数量级. 直线的识别率在 90% 以上, 识别精度 (方向、线宽和端点定位) 较高, 交互修改的工作量很小. 表 3 给出了具有代表性的一些中等质量图纸的直线矢量化实验结果.

Table 3 Experimental results of line vectorization by GLV

表 3 一些图纸的直线矢量化测试结果

Drawing ^①	Size ^② (pixel2, Mbytes)	Line number ^③	Process time (s) ^④	Miss ^⑤	False alarm ^⑥	Recognition rate ^⑦ (%)
Stairs	3358×3478, 1.427	377	9.1	9	17	92.8
Ds34	7200×4692, 4.124	564	14.3	18	36	90.3
Details	16215×11856, 23.48	1670	53.8	29	93	92.5
Framing	16128×11806, 23.23	1238	40.6	20	79	92.0

①图纸名, ②尺寸, ③直线总数, ④处理时间 (秒), ⑤漏识数, ⑥误识数, ⑦识别率.

4 结束语

本文提出了一种无需拟合和拼接的直线矢量化方法. 通过使用种子段和定向跟踪技术, GLV 能够一次性识

别整条直线(包括虚线),并消除了方向依赖性.本文还从理论上对比和分析了该方法与现有矢量化方法的性能,最后给出的实验结果证实了该方法直线矢量化的速度和精度比现有方法有较大程度的提高.

参考文献

- 1 Clement T P. The extraction of line-structured data from engineering drawings. *Pattern Recognition*, 1981, 14(1):43~52
- 2 Datta A, Parui S K. A robust parallel thinning algorithm for binary images. *Pattern Recognition*, 1994, 27(9):1181~1192
- 3 Hori O, Tanigawa S. Raster-to-vector conversion by line fitting based on contours and skeletons. In: *Proceedings of the International Conference on Document Analysis and Recognition*. Tokyo: IEEE Computer Society Press, 1993. 272~281
- 4 Roseborough J B, Hiroshi Murase. Partial eigenvalue decomposition for large image sets using run-length encoding. *Pattern Recognition*, 1995, 28(3):421~430
- 5 Tan Jian-rong, Peng Qun-sheng. A global line recognition approach to scanned image of engineering drawings based on graphics constraint. *Chinese Journal of Computers*, 1994, 17(8):561~569
(谭建荣,彭群生.基于图形约束的工程图扫描图像直线整体识别方法.计算机学报,1994,17(8):561~569)
- 6 Tombre K. Analysis of engineering drawing: state of the art and challenges. In: Tombre K, Chhabra A K eds. *Graphics Recognition—Algorithms and Systems*. Berlin: Springer-Verlag, 1998. 257~264
- 7 Jasson D T, Vossepoel A M. Adaptive vectorization of line drawing images. *Computer Vision and Image Understanding*, 1997, 65(1):38~56
- 8 Zou Rong-jin, Cai Shi-jie, Zhang Fu-yan. Line interpolation method and error estimation based on run length coding. *Journal of Software*, 1997, 8(supplement):404~410
(邹荣金,蔡士杰,张福炎.基于行程编码的直线拟合方法及其误差估计.软件学报,1997,8(增刊):404~410)
- 9 Wang Jiu-he, Ou Zong-ying, Xia Xiao-dong. A global algorithm for recognition of lines in scanned engineering drawings. *Journal of Image and Graphics*, 1998, 3(11):912~917
(王金鹤,欧宗瑛,夏晓东.工程扫描图像的直线整体识别算法.中国图象图形学报,1998,3(11):912~917)
- 10 Chiang J Y, Tse S C, Leu Y C *et al.* A new algorithm for line image vectorization. *Pattern Recognition*, 1998, 31(10):1541~1549
- 11 Dori D, Liu W. Sparse pixel vectorization: an algorithm and its performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, 21(3):202~215

A Seed Segment Based Orientation Independent Line Vectorization Method

SONG Ji-qiang SU Feng CHEN Ji-bing CAI Shi-jie

(State Key Laboratory for Novel Software Technology Nanjing University Nanjing 210093)

(Department of Computer Science and Technology Nanjing University Nanjing 210093)

Abstract Accurate and efficient vectorization are the foundation of engineering drawing understanding. The existing vectorization methods have three main shortages: (1) A line segment is fitted using a chain of medial axis points so that the computational complexity is high; (2) A lot of time-consuming searching and judging operations are needed to combine the separated segments caused by noise or intersections; (3) The capability of processing slant lines is much lower than that of processing vertical or horizontal lines. In this paper, the authors propose a seed segment based global line vectorization method, which vectorizes an entire line in one step so that the combining step is avoided. Furthermore, it processes the lines of any orientation in the same way. Both the performance evaluation and experimental results show that the performance of line vectorization is significantly improved using the proposed method.

Key words Line vectorization, engineering drawing recognition, seed segment, image tracking, orientation independent.