

电脑刺绣中的字符形状分割算法^{*})

朱逢华 王崇俊 陈世福

(南京大学计算机软件新技术国家重点实验室 南京 210093)

摘要 本文提出了一个应用于刺绣 CAD 系统中的字符分割算法。该算法利用字符图像的全局信息,通过以骨架交点作圆来确定需要添加分割线的范围;利用边界信息最后确定分割线来对字符进行形状分割。实验表明该算法对边界噪声具有较强的抗干扰能力,分割的效果能达到电脑刺绣的要求。

关键词 字符刺绣,图像分割,骨架抽取,候选分割点

A Character Shape Decomposition Algorithm in Embroidery

ZHU Feng-Hua WANG Chong-Jun CHEN Shi-Fu

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

Abstract This paper presents the character decomposition algorithm which is applied to PRECISE DIGITIZING SYSTEM. This algorithm is based on the global information of character image and defines the range of adding decomposition lines through drawing circles around skeleton's cross points. Combined with the edge information, we can define how to segment the character images based on segmentation lines. Our experiments have shown this algorithm can avoid most edge noise and the segmenting effect can approach the demands of automatic computer needlework.

Keywords Character embroidery, Image decomposition, Skeletonize, Candidate decomposition point

1 前言

刺绣是我国的一项传统工艺,历史悠久,源远流长,造型千变万化,多姿多彩,具有很高的艺术价值和收藏价值^[1],以前的刺绣是手工劳动,效率很低,在现代,伴随着电脑绣花机的出现,利用计算机刺绣 CAD 系统^[2,3]设计已成为当今刺绣工业的主流。

文字在刺绣作品中使用得非常频繁,很多衣服、运动包和帽子上都会绣上品牌名称。因此,将文字绣在绣品上是刺绣中经常会使用到的方法。



a) 固定走针方向效果 b) 形状适应方向效果

图 1 两种方向的走针效果对比(环境为 PDS 智能刺绣系统)

文字刺绣可由用户手工输入相应字符轮廓来完成,但由于字体的种类繁多,而且,不断有新的字体出现,用手工输入轮廓的方式相当费时,这样刺绣制作人员就迫切希望能自动地产生相应字符轮廓,来减少绣品制作时间。目前,使用最多的是 TrueType 字体,很多辅助刺绣系统例如天术、Wilcom、田岛、MAGICSTITCH 等都支持将 TrueType 字体的轮廓作为文字刺绣的轮廓使用。但多数系统的效果很差,主要原因在于其只能在固定方向上走针,如图 1-a 所示,当区域在该方

向较宽时,绣线会很长,就显得不太美观,也没什么实用价值;相比而言走针方向与区域形状相适应更符合工艺要求,如图工 1-b 如示。为使走针方向与区域相适应,系统必须首先将字符分割成一系列的带状,然后对每个子区域编针。本文给出一个字符形状的分割算法,来解决字符自动刺绣。

2 字符的形状分割算法

一个字符在人们看来也算是一幅图像,故也可以采用图像形状分割算法,图像形状分割主要有两大类方法,第一类是基于局部边界信息的方法,通过分析边界凹角点来进行分割。文[4]通过连接边界上对应的凹角点将形状分割成若干个简单的部分,其中每个部分都是凸多边形。但是这类方法对边界噪声较为敏感,并且不太适用于较复杂的形状的分析,同时,对于字符,很多边界是平滑的,无法找到相应凹点。第二类方法是基于形状的整体信息,通过分析形状骨架来进行分割。文[5]根据骨架中的片段将形状直接分割成简单的部分。

我们日常使用的字符多数是 TrueType 字体格式。对 TrueType 字体的自动编针,文[6]也给了一个方法,其没有自动分割过程,而是人工预先对字符分割并存入文件,以后使用时再调用,这样就无法适应不断增长的 TrueType 字体。

我们研制的算法是针对 TrueType 字体, TrueType 字体中已经存储有字符的轮廓信息,在我们进行形状分割时为了保持最后刺绣的结果与开始的字符形状保持不变,就必须采用这些轮廓信息。我们的算法结合了前面两类图像分割算法,并利用了 TrueType 字体的特点,使自动分割的结果十分接近人工刺绣分割的结果。

2.1 算法设计思想

^{*})本文工作得到国家 863 计划(863-306-2704-01-3)资助。朱逢华 硕士生,主要研究领域为 CAD & CG 和智能化信息系统。王崇俊 博士,主要研究领域为 CAD 及人工智能。陈世福 教授,博导,主要研究领域为人工智能理论应用。

我们的算法是先把字符轮廓填充成图像,并抽取骨架,然后找出骨架中分枝的交点,以分叉点为圆心,分叉点到轮廓的最短距离为起始半径作圆,并不断地扩大这半径,直到这个圆与轮廓的交点数大于等于分枝数的2倍。然后再从这些交点中得到候选分割点,并依据一定规则选取一些分割点连接起来,最终将字符分割开。我们的算法主要步骤如下:

(1)读取 TrueType 字体的轮廓信息,并且将字符填充成图像。

(2)提取字符图像的骨架。骨架提取算法十分重要,本文在 2.2 节给出了该算法。

(3)以骨架交点作圆,求出与字符轮廓的交点,求出分割的候选点。这是此算法的核心,本文将在 2.3 节中详细描述该算法。

(4)根据第三步所求出的分割候选点,及字符轮廓的各种信息选取最适合的划分辅助线,将字符划分成一系列带,或者是面。

2.2 骨架抽取算法

在我们的算法中骨架的主要作用是确定分枝数,因此我们主要关心的是三个及以上分枝的骨架的交点。为方便叙述,我们称骨架上的像素点为骨架点,把一个骨架点的按八连通相邻的骨架点称为该骨架点的邻居,把一个骨架点所拥有的邻居数称为此骨架点的度,相应地可称这个骨架点为几度点,如图 2 中骨架点 A 的度为 3,点 A 被称为 3 度点,骨架点 B 的度为 4,点 B 被称为 4 度点。1 度点和 2 度点均不是分枝的交点,3 度及其以上有可能是骨架分枝的交点。在我们的算法中采用骨架点的度来确定分枝数,并要求一个 $n(n \geq 3)$ 分枝只对应一个 n 度点。

本文所采用的是现在比较流行的顺序细化的方法^[7],很多图像处理库都包含了这个提取骨架的方法,这里就不再赘述。由于我们是依据骨架点的度来确定分枝数,但是抽取出来的八连通的骨架一般不符合我们要求,如图 3,表中标 1 的点为骨架点,按八连通的定义点(2,2)(3,2)(3,3)的度均为 3,但实际上这里仅有一个三分枝,所以需要去掉两个三度点,为了适应后继算法中一个 $n(n \geq 3)$ 分枝仅对应一个 n 度点的要求,我们预先对骨架作了一个过滤操作。骨架过滤算法如图 4。



图 2 骨架点的度

1	0	0	0	0
0	1	0	0	0
0	1	1	0	0
0	1	0	1	0
0	1	0	0	1

图 3 多余的骨架分枝点

(1)置全局处理过标志为 true,置所有 $n(n \geq 3)$ 度点为未处理。

(2)如果全局处理过标志为 true 则置全局处理过标志为 false,并置所有 $n(n \geq 3)$ 度点为未处理,转(3),否则此算法成功结束。

(3)选择一个未处理过的 n 度点,如果都已处理转(2),否

则转(4)。

(4)置此骨架点为已处理,判断此骨架点与其邻居是否有公共的邻居点,如果有转(5),否则转(3)。

(5)将此骨架点的与其有共同邻居的邻居点删掉,并将这被删掉的邻居点的邻居作为此骨架点的新邻居,但是要注意的是所有原来已是此骨架点的邻居及其本身不能作为其邻居点;置全局处理过标志为 true,转(3)。

图 4 骨架过滤算法

通过上述过滤操作,便可使一个 $n(n \geq 3)$ 分枝只对应一个 n 度点,虽然,由于算法最后保留的这个 n 度点可能并非原来的分枝交点,但是其位置偏移并不大,而且,我们后面的算法对这种偏移也能容忍。

2.3 求候选分割点算法

在经过骨架提取与过滤后,我们可以得到所有度大于等于 3 的骨架点,根据骨架的最大球特性, $n(n \geq 3)$ 度点,可以找到 n 个离其距离最近的轮廓点。并且,在这些轮廓点中根据一定规则选取一些连接起来,即可将原图像分割开来。我们把这些轮廓点称为候选分割点。由于骨架提取是采用图像方式处理,这样我们所得到的骨架点就可能并不是理论上的最大球点,使我们在找这些候选分割点时出现了困难。为了解决这个问题,我们提出了基于骨架交点作圆的算法来找出这些候选分割点。对每一个 $n(n \geq 3)$ 度的骨架点找其候选分割点的算法如图 5 所示。

(1)计算骨架点与轮廓上每条 Bezier 曲线的起始点的距离,选其最小的一个赋值给 radius。

(2)判断在以这个骨架点为圆心、radius 为半径的圆的内是否还有其他度大于等于 3 的骨架点,如果有,则取这些点的几何中心点为新骨架点,新骨架点的度为这个圆内所有度大于等于 3 的骨架点度的和减去这些骨架点的总数。删除这个圆内的度大于等于 3 的骨架点,即仅保留这个新的骨架点,并转(1)。

(3)求出以此骨架点为圆心、radius 为半径的圆与所有轮廓的交点。

(3)假如交点数大于等于 $2 * \text{骨架点的度}$,并且交点数为偶数则转(5),否则转(4)。

(4)搜索数加 1,假如搜索数大于 5 则宣告失败,否则 radius = radius * 1.2 转(2)。

(5)把同一轮廓上的点按一定规则每两个点组成一个对,在这两个点之间找出所有凹角点,如果没有这样的点,则取这两个点在轮廓上的中间点,如果有多个这样的点则取离骨架点距离最近的一个点,只有一个则取这唯一的点。

(6)判断所选点的合法性,删除不合法的点,直到点的个数等于骨架点的度,如果小于点的骨架点的个数则二次搜索数加 1,假如二次搜索数大于 2 则宣告失败,否则, radius = radius * 1.1, 转(2)。

(7)经过合法性判断的点则为全部候选分割点。

图 5 寻找候选分割点算法

在算法中所提到每两个点组成一个对的规则是,判断同一轮廓上第一个点和最后一个点在轮廓上的中点离骨架点的距离是否小于 radius,如果是则将第一和最后一个点组成一个对,相应地把第二、第三个点组成一个对,后面的类推,如果不是,则将第一、第二个点组成一对,相应,把第三、第四个点组成一个对,后面的类推。

在算法中所提到的合法性判断,主要是针对三度点,规则是判断骨架点是否在所选的三个点所组成的三角形内,如果在则是合法的。而删除点的规则是看是否有三个点组成的三角形包含骨架点,把多余的点删掉。

2.4 选取分割线算法

在找到候选分割点后,我们在这些候选分割点中选取一些点连接起来形成分割线,这样我们就可以将字符分割成带状系列了。

因为字符图像的骨架的分枝交点的度多数为 3 度或 4 度,大于 4 度的骨架交点很少出现,故我们在选择分割点分割字符时将依据骨架点的度数分成 3 度点、4 度点,及大于 4 度点三个大类来考虑。

三度点是最常出现的骨架点,对于三度点一般只选取一条分割线,即在三个候选分割点中选取两个点连接起来。考虑到 TrueType 字符的特点,向内凸的点即凹点一般即是分割点,所以我们在选择分割点时,主要是先择凹点,而且由于我们在前面已求出候选分割点,这与仅考虑边界凹点的算法相比,就更少受边界噪声的干扰。对于没有凹点的情况,我们再考虑曲率,曲率大的点一般对应弯曲程度比较大的点。下面,将列出选出这两个分割点的各种情况:

(1)如果在这三个点中有两个点的夹角 \cos 值大于阈值(经实验取 -0.86),则直接选取这两个点作为分割点。

(2)如果只有一个点大于阈值,则选取此点作为尖角点,并选取另一个曲率较大的点作为修改点,然后执行修正算法。

(3)如果三个点均不大于阈值,则选择两个曲率最大的点作为分割点。

对于(2)条所提的修正算法,主要是为了使分割的形状更符合手工刺绣,后面实验结果中的 abd 三个字符都用到了这个算法,这个算法描述如图 6。

(1)在尖取点的两条边中选取一条与尖角点和修改点所连成的线段夹角最小的一条边。并记尖角点同修改点的连线为正确边。

(2)向原修改点方向延长所选取尖角点的边,求出与原修改点所在轮廓的交点。

(3)次数加 1,如果次数大于 10 转(6)判断所求交点与原

交点的中间是否有向区域外凸的点,如果有则不合法转(5),否则为合法转(4)。

(4)如果所求点是合法的并且是第一次所求,则转(6),如果不是第一次,将尖角点与所求的点的连线作为正确边,并求错误边与正确边的所构成的角的角平分线与原修改点所轮廓的交点,转(3)。

(5)将所求点与尖角点的连线记为错误边,并求错误边与正确边所构成的角的角平分线与原修改点所在的轮廓的交点转(3)。

(6)将确定正确边的两点作为分割点,并退出。

图 6 三度点分割线修正算法

在字符中,四度及其以上点所表现出来的形式一般是比较简单的,所以,我们在处理时也采用比较简单的方法。对四度点,一共会找到四个分割点,我们把这四个点分成两个对,每一个对连接成一条分割线,而分割的原则是这两条分割线不相交即可。对于四度以上的点,我们将每个分割点首尾连接即可。这种分割线的选择方式比较简单,对任意形状来说,可能会出现很大的错误,但是对于字符来说是比较合适的,基本上均能满足实际需要。

3 实验结果

图 7 为 Arial regular 字体的部分字符应用本算法后的分割结果,从图 7-c 中可以看出该算法对字符分割有很好的效果,与人工的分割结果非常接近。图 7-d 是字符分割后在我们研制的 PDS 智能刺绣系统中的刺绣结果,可以看出刺绣的效果也非常好。

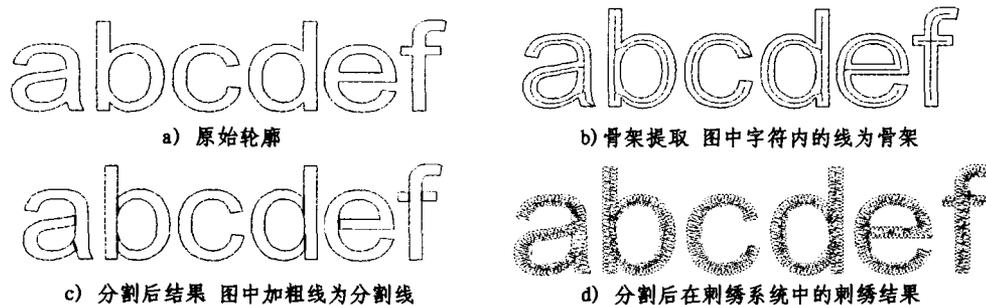


图 7 Arial regular 字体部分字符的分割及刺绣结果

我们对多种字体作了实验,实验的结果表明该算法对多数字符都能做比较好的分割,特别是对有凹角的字符分割成功率可达到 99%,在实验中只有极少数字符无法分割或者分割的效果不是很好;对于没有明显凹角的字符,多数也能分割得比较好,但是有少数字符的分割效果与人工分割的结果还有一定的差距,主要原因在于算法中对于无明显凹角的分割点采用了求曲率的近似算法,有时候并不能很好地表现出整体的转角信息,在下一步的工作中,我们准备将字符转换成近似的多边形来解决这个问题。

结论 本文提出了一个用于刺绣 CAD 中字符的自动分割算法。算法通过以骨架交点作圆来求出所有候选分割点,并从这些候选分割点之间的连线中选择一条作为最终的分割线,同时,为使其更符合手工刺绣分割的要求,还对三分枝的情形作了一些优化。实验表明该算法对区域边界噪声具有较好的抗干扰能力,对骨架提取的要求也并不是太高,同时,经过一些改进,也可以用于带状比较多的图形。该分割算法已

应用于我们研制的智能化电脑刺绣 CAD 系统,已达到实用化,效果很好。

参考文献

- 1 魏国志. 面向对象的字符自动编针设计与实现:[南京大学硕士学位论文]. 2002
- 2 李红兵,陈世福,陈兆乾,等. 智能电脑刺绣集成环境. 计算机辅助设计与图形学学报, 2000, 12(1): 1~5
- 3 宋晓丹,罗子频,武藤幸好小宫量平. 刺绣仿真的建模与实现. 计算机辅助设计与图形学学报, 2003, 40(1): 88~93
- 4 Siddiqi K, Kimia B B. Parts of Visual Form: Computational Aspects. IEEE Transaction on Pattern Analysis and Machine Intelligence, 1995, 17(3): 239~251
- 5 Abe K, Arcelli C, Hisajima T, Ibaraki T. Parts of Planar Shapes. Pattern Recognition, 1996, 29(10): 1703~1711
- 6 杨育彬,郭磊,陈世福,陈兆乾. 字符自动刺绣编针技术的研究. 计算机研究与发展, 2003, 40(1): 88~93
- 7 Suzuki S, Abe K. Sequential thinning of binary pictures using distance transformation. In: Proc. of the 8th Intl. Conf. on Pattern Recognition, 1986. 289~292