基于 0-1 规划的规则中文文件碎片自动拼接技术①

蓝　洋, 和　亮

(西安外事学院 工学院, 西安 710077)

摘　要: 为了实现规则中文文件碎片的拼接, 研究了规则碎片文件中汉字文本的特征, 提出了文件碎片中文本行信息的提取方法, 定义了基于 L^1-norm 的碎片边界差异度概念, 建立了基于 0-1 规划的文件碎片拼接模型, 并运用聚类分析降低了算法复杂度. 与现有同类算法相比, 本文的算法无需使用人工干预即可完成正确拼接.

关键词: 规则碎片拼接;　0-1 规划; 聚类分析; 文本特征提取; L^1-norm

Automatic 0-1 Programing Based Reassembly of Fragmented Chinese Documents

LAN Yang, HE Liang

(College of Engineering, Xi'an International University, Xi'an 710077, China)

Abstract: In this thesis, feature of Chinese characters in regular fragments of document is studies and a method of extracting line-information of text is proposed. By defining the concept of L^1-norm based differences between adjacent fragments, we develop a reassembly algorithm base on 0-1 programming and reduce the algorithm complexity by using cluster analysis. Compared with existing method, our reassembly method can fulfill the reassemble of given fragmented Chinese text effectively and efficiently without artificial supplementary.

Key words: regular fragments reassembly; 0-1 programing; cluster analysis; text feature extraction; L^1-norm

文件碎片的拼接与复原在情报获取、文件修复中具有重要应用[1]. 在没有计算机辅助的情况下, 仅依靠人工可以完成少量碎片的拼接, 但是耗时相对较长. 当碎片规模增大时, 人工拼接的难度与人力工时均大大增加, 因此, 设计恰当的算法实现计算机拼接可以解决这一问题. 现实中的文件碎片拼接大致分为两类: 不规则形状的碎片拼接[2-6](需依赖边界轮廓和碎片上的信息)和规则切割碎片的拼接[7-10], 后者所针对的碎片没有明显边界形状信息, 因而必须依赖碎片上的信息进行拼接.

本文针对第二类拼接问题, 对规则中文文件碎片的全自动拼接技术进行了研究. 首先, 对碎片进行了数字化与特征分析, 运用聚类算法对依据文字行信息对数据进行了分组, 降低了拼接算法复杂度. 然后, 定义了基于 L^1-norm 的碎片边界差异度概念, 并据此建立了以总差异度最小为目标的 0-1 规划模型. 最后再次使用行信息完成了文件的复原.

1　文件碎片的数字化

本文所使用的碎片样本来自一页印刷的中文文件, 被碎纸机切割成 209 张尺寸相同的碎片, 碎片内所印刷的中文文字大小一致, 字间距、行间距均为常数, 文字按行水平排列. 为了利用计算机进行碎片拼接, 对文件碎片实物进行扫描得到其图片文件, 再利用软件将每个图片文件转化成尺寸为 180×72 的矩阵(后文称其为碎片矩阵)并编号. 碎片矩阵中的元素值对应图片文件中像素的灰度值, 矩阵元素的取值区间为[0,255]. 图 1 所示为一张文件碎片对应的碎片矩阵.

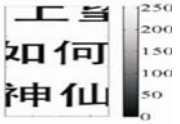

图 1　文件碎片对应的碎片矩阵

① 基金项目:陕西省教育科学"十二五"规划项目(SGH13481)
收稿时间:2014-10-24;收到修改稿时间:2014-12-01

2 碎片的边界差异度定义

规则切割的文件碎片不具有可识别的轮廓信息, 因而必须依靠图片中的文字信息进行拼接. 本文根据 L^1 范数的概念, 利用两个图片边界像素的差异大小判断是否应该进行拼接, 后文中称为边界差异度. 设第 i 个碎片矩阵的最后一个列向量(即第 i 个碎片右边缘的一列像素灰度值)记为 r_i, 第 j 个碎片矩阵的第一个列向量(即第 j 个碎片左边缘的一列像素灰度值)记为 l_j, 两碎片左右拼接(碎片 i 在左边, 碎片 j 在右边), 其横向拼接的边界差异度定义为

$$h_{ij}=\sum_{k=1}^{180}|r_{ik}-l_{jk}|,i,j=1,2,\ldots,209 \tag{1}$$

其中, r_{ik} 和 l_{jk} 是向量 r_i 和 l_j 的第 k 个元素. 同理, 也可以定义两张上下纵向拼接的边界差异度为

$$v_{ij}=\sum_{k=1}^{180}|u_{ik}-d_{jk}|,i,j=1,2,\ldots,209 \tag{2}$$

其中, u_{ik} 和 d_{jk} 是向量 u_i(第 i 个碎片的上边缘对应的向量)和 d_j(第 j 个碎片的下边缘对应的向量)的第 k 个元素. 根据定义式可以看出, 边界差异度取值越小, 表明两个边界正确拼接的概率越高.

3 基于0-1规划的数学模型

理论上, 由裁剪得到的两相邻边界应无差异, 因此, 正确拼接时的碎片边界差异度总和应为零. 但考虑到图像分辨率及像素灰度值的变化, 实际切割中很可能存在边界差异, 所以, 从概率角度而言, 最优拼接方式是使得全部拼接边界的差异度总和最小的方式.

文献[10]提出了 0-1 规划的数学模型来描述此问题, 其目标函数为

$$\min\sum_{i=1}^{209}\sum_{j=1}^{180}(h_{ij}x_{ij}+v_{ij}y_{ij}) \tag{3}$$

其中,

$$x_{ij}=\begin{cases}0,\text{表示第}i\text{个碎片和第}j\text{个碎片未左右拼接}\\1,\text{表示第}i\text{个碎片和第}ji\text{个碎片左右拼接}\end{cases}$$

$$y_{ij}=\begin{cases}0,\text{表示第}i\text{个碎片和第}j\text{个碎片未上下拼接}\\1,\text{表示第}i\text{个碎片和第}ji\text{个碎片上下拼接}\end{cases}$$

从该模型的目标函数可以看出, 此模型同时考虑了所有碎片的横向拼接与纵向拼接, 是一个二维拼接问题, 但由于该模型需要设置的变量较多, 约束条件复杂, 模型求解时需要的内存空间很大, 因此会导致普通配置的计算机在求解该模型时会出现内存溢出. 为了解决模型求解的内存问题, 一个很自然的想法是将大规模数据进行分组, 将二维碎片拼接问题降解成为一维碎片拼接问题, 即对碎片仅进行横向的行拼接或者纵向的列拼接. 例如, 对一组横向拼接的碎片而言, 0-1 规划的数学模型被简化为:

$$\begin{cases}\min\sum_{i=1}^{N}h_{ij}x_{ij}\\ s.t.\ \sum_{i=1}^{N}x_{ij}\le 1\\ \sum_{j=1}^{N}x_{ij}\le 1\\ \sum_{i=1}^{N}\sum_{j=1}^{N}x_{ij}\le N-1\end{cases} \tag{4}$$

其中, x_{ij}=1 表示第 i 个碎片和第 j 个碎片左右拼接, x_{ij} =0 表示两者未拼接, N 为待拼接碎片的数量.

在运用模型(4)进行碎片拼接之前, 必须先将全部碎片进行分组. 由于文件碎片是文本文件的规则切割, 可以保证文字的行与横向切割方向水平、与纵向切割方向垂直, 因此利用每张碎片中文字的行信息即可进行准确分组.

4 行信息特征提取与行聚类

为了进行准确分类, 需要确定有效的行信息特征. 通过观察给定文件碎片可以发现以下特点:

1)每张碎片的高度可以包含三行文字, 有些碎片仅被印刷了一行或者两行文字;

2)碎片的上下端有可能仅含被切断的半行文字;

3)大多数碎片中每行完整文字的高度相等, 经测试, 文字高度为 41, 行高为 68;

4)个别碎片中的行仅包含如“一”这样的文字, 其高度小于 41.

其中第 1)、2)和 4)给行分类带来了行信息的随机性, 采用文献[10]的方法将导致部分特殊碎片无法正确分类. 经分析可知, 由于文字的行高是常数, 所以只要在碎片中包含一行完整的文字, 就可以生成碎片上其它行的信息. 本文选用首行中轴线位置(在坐标系中的纵坐标)来表示碎片的行信息特征. 那么, 本应被左右拼接的两张碎片的首行中轴线应该重合. 本文采用滑窗求和法获取碎片的首行中轴线位置, 计算的方法介绍如下.

记碎片文件的行标记向量为 $V=(v_1,v_2,\ldots,v_{180})^{\mathrm{T}}$, v_i=0 表示碎片矩阵第 i 行元素全为 255(即空白行), v_i=1 表示第 i 行中有小于 255 的元素(即该行中有文字的黑

色像素). 如图 2 所示, 图 2(a)为碎片矩阵, 图 2(b)为行标记向量 V, 图 2(c)为滑窗求和向量 $W=(w_1, w_2,\ldots, w_{108})^{\mathrm{T}}$, 其计算公式为

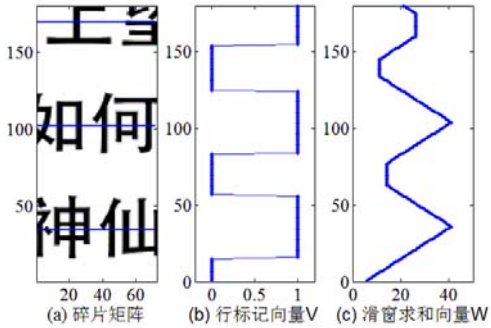

图 2 碎片文字的首行中轴线位置的提取

$$w_k=\sum_{l=k-20}^{k+20} v_l,\ k=1,2,\ldots,180 \tag{5}$$

滑窗的窗口宽度为 41(文字的高度). 则每张碎片首行文字中轴线的位置 Mid 用表达式(6)进行计算:

$$Mid=\mathrm{mod}(\arg\max_k(w_k),68) \tag{6}$$

其中, $\arg\max(w_k)$ 表示使得 w_k 取到最大值的 k. 如图 2(a)中的第一个横线记为首行中轴线, 其纵轴坐标为 170.

通过以上方法, 可以获得所有碎片的首行中轴线位置坐标, 其结果如图 3 所示, 图中每个点的横坐标是碎片序号, 纵轴坐标即为该碎片的首行中轴线的位置. 从图中可以看出, 首行中轴线具有非常明显的聚类特征, 设置 10 个门限(如图中的横线所示), 即可将全部碎片进行分组. 从分组结果可知, 209 张碎片被分为 11 组, 每组内有 19 张碎片. 与文献[10]所采用的方法相比, 本文的聚类方法无需人工干预, 可以一次完成正确分组.

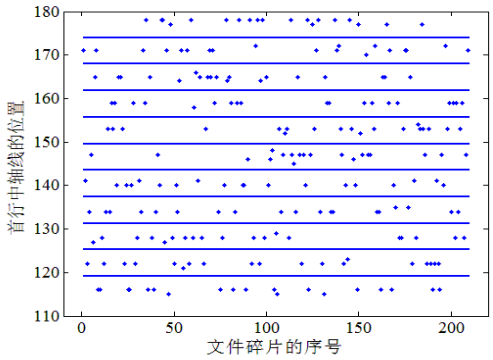

图 3 文件碎片的首行文字中轴线位置与聚类门限

在得到正确分组之后, 对每组碎片利用模型(4)进行横向的行拼接, 即可获得所有分组的正确拼接结果.

5 文件的拼接复原

在得到正确的分组和每个分组的拼接结果(行碎片)之后, 需要对 11 个行碎片进行拼接, 以复原原始文本文件. 由于在切割文件时, 横向切割线有较高的概率位于两行文字之间的空白行间距, 此时无法利用行碎片的边界像素值进行匹配, 因而, 在不采用其他辅助手段的情况下, 应用 0-1 规划无法实现行碎片的正确拼接. 本文提出了一个搜索策略, 利用已有的首行中轴线信息, 可完成最终文件的拼接复原.

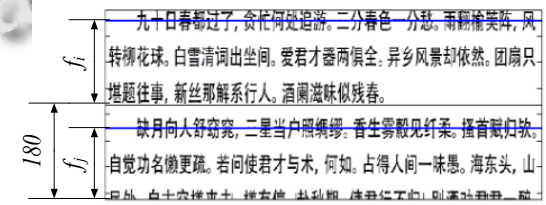

图 4 行碎片的拼接示意图

参考图 4, 设第 i 个与第 j 个行碎片的首行中轴线位置分别为为 f_i 和 f_j, 若第 i 个行碎片与第 j 个行碎片应该被上下相邻拼接, 则两个行碎片首行中轴线间的距离应为行高的整数倍, 即应该满足表达式

$$f_i+(180-f_j)=68\times N \tag{7}$$

其中, N=3, 行高为 68. 仿真结果表明, 此方法可自动完成正确的行碎片拼接.

6 总结

本文对待拼接的文件碎片进行了特征分析, 获得了碎片的边界向量, 提取了碎片上中文文字的行特征信息, 先利用行特征对所有碎片进行了分组, 然后对每个分组利用边界向量差异度进行了基于 0-1 规划的碎片拼接, 最后将拼接好的分组碎片再次利用行信息进行拼接, 最终复原为原始文件. 本文提出的拼接算法快速有效, 只要能够保证每个碎片中至少有一行完整文字, 即碎片高度不小于行高与文字高度之和, 本算法均能在无人工干预的情况下自动完成拼接.

参考文献

1 杨洛斌.形状匹配技术在文物复原中的研究与应用[学位论

文].西安:西北大学,2002.
2 贾海燕,朱良家,周宗潭,等.一种碎片自动拼接中的形状匹配方法.计算机仿真,2006,23(11):180–183.
3 罗智中.基于线段扫描的碎片边界检测算法研究.仪器仪表学报,2011,23(2):289–294.
4 谢萍,马小勇,张宪民,等.一种快速的复杂多边形匹配算法.计算机工程,2003,29(16):177–181.
5 朱延娟,周来水.二维非规则碎片的匹配算法.计算机工程,2007,33(24):7–9.
6 张欣卜,彦龙,朱良家,等.物证复原系统中的碎纸轮廓提取技术研究.计算机仿真,2006,23(11):184–187.
7 Efthymia T, Ioannis P. Automatic color based reassembly of fragmented images and paintings. IEEE Trans. on Image Processing, 2009, 19(3): 680–690.
8 Nasir M, Anandabrata P. Automated reassembly of file fragmented images using greedy algorithms. IEEE Trans. on Images Processing, 2006, 15(2): 385–393.
9 罗智中.基于文字特征的文档碎纸片半自动拼接.计算机工程与应用,2012,48(5):207–210.
10 沈鸿平,章毅鹏,王义康,基于 0-1 规划模型的规则中文碎片拼接复原研究,电子科技,2014,(27):13–16.