藏文历史文献粘连字丁串数据库及改进的滴水算法

# 1.简介

班禅作品全集

The complete works of Panchen Lama

字丁串的切分研究在字丁识别系统中具有至关重要的作用，是一个比较传统但仍未完全解决的重要课题，这方面的研究工作从二十世纪八十年代就已经开始。目前，对于数字、英文、中文的切分识别已经取得了不错的效果，在邮政编码识别、银行支票阅读、文本识别等方面有着重要的应用，很少有关于藏文粘连字丁串的研究。在字丁串的切分任务中，粘连字丁串的处理是最为核心也是最复杂的问题。对于粘连字丁串的切分可大致分为两大类，第一类隐式切分（Implicit segmentation），以滑动窗口的方式从左到右遍历，得到一组序列，然后基于动态规划或隐马尔科夫模型等，选择最佳的识别效果。第二类显示切分（Explicit segmentation），这种切分方式又可进一步分为两类，一类是弱切分，另一类为过切分。（1）弱切分，弱切分算法主要特征是考虑局部信息，最终只会产生一条切分路径，适用于粘连不严重的情况，代表算法有投影[4][5]、滴水算法[6]、蓄水池算法[10]等。这些算法切分效率高，在粘连不严重的情况下能够取得不错的效果。投影在粘连严重（投影峰值点不在粘连区域）的情况下会造成切分错误，滴水算法及其改进算法主要是模拟水滴在重力作用下落的过程，水滴下落的路径形成切割路径，初始降落点的选择对于是否能正确切分具有决定性作用。蓄水池算法认为在字丁串粘连处会形成坑槽，即蓄水池，根据所要切分字丁的特点制定合适的规则将水库穿透即可得到最佳分割路径。（2）过切分，过切分算法会考虑所有可能存在的切分路径，会产生多条切分路径。可大致分为三类：基于前景分析；基于背景分析；基于组合分析。首先利用前景、背景提取轮廓和骨架信息，然后提取所需的特征点，基于这些特征点在粘连字丁串上得到多个候选组合，最后结合字丁识别器、几何模型、语言模型，筛选出最佳的分割路径。相对于弱切分算法，过切分算法切分准确度高，在粘连严重的字丁串中仍能表现出很好的性能。过切分算法核心思想是将粘连的字丁串切分成多个小部件，这样得到的小部件是单个字丁或单个字丁的一部分。原则上所得到的小部件会小于经隐式切分所形成的候选部件，通过组合不同的小部件，形成候选字丁模式，结合语言模型、几何信息对候选模式进行打分或者动态规划，最后得到最优切分路径。

在过切分方法中，常对粘连字丁串的前景骨架、轮廓和背景骨架进行分析，也有学者从多个角度对粘连字丁串进行综合分析，试图找到最佳的切分路径。可将粘连字丁串切分方式大致分为三种：基于前景分析；基于背景分析；基于组合分析。

为了避免因数据集不同造成切分算法对粘连字丁串的切分效果有差异，分别有学者建立了粘连字丁串数据库。其中有代表性的有：粘连手写数字串数据库、中文粘连字丁串数据库（CASIA-HWDB-T）。表一显示了两个数据库的规模及相关的存储信息。粘连手写数字串数据库中的数据是根据NIST SD19中单个的数字合并得到的，作者选取了2000图片进行合并得到了273,452张粘连手写数字串。考虑到人工合成粘连字丁串，可能与真实手写字丁串有差异，许亮根据已经标记好的脱机手写文本数据库CASIA-HWDB，抽取了其中的所有粘连字丁串构成了新的数据库。CASIA-HWDB-T包含56,469个粘连字丁串，其中大部分是单粘连丁符串，1,818个是多粘连字丁串。

在这篇文章中，参考文献【】【】，我们通过对藏文历史文献进行连通区域分析，建立了藏文历史文献粘连字丁串数据库。它由X张两个字丁的粘连和X张三个字丁的粘连数据组成，存在多种粘连方式。另外，我们在我们的数据库上利用改进的算法进行了一些实验。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

# 2.数据库

To the best of our knowledge, no similar data have been built so far.

到目前为止，还没有公开发表过任何关于藏文粘连字丁串的数据库。下面，我们将介绍我们数据库的建立及标注信息。

## 2.1采集数据

藏文文字通常以一个字母为中心，其余字母以此为基础前后附加和上下叠写，组合成一个完整的字表结构，一个藏字最多可由6个辅音字母组成，如图1所示，在进行藏文切分识别时通常以一个字丁作为识别单位（上下叠写构成的单位）。我们对藏文历史文献《》进行扫描，得到原始的扫描图像。在粘连字丁串抽取中，首先对图像进行人工切分成行图像；然后对行图像进行二值化和降噪处理；最后通过连通区域分析构成了我们的粘连字丁串数据库的原始数据集。

在连通区域分析时，考虑到字丁存在上下叠写的情况，结合XX提出的算法，对联通区域进行合并。首先通过四邻近对行图像进行连通区域标记，并保存每个连通区域的边框和区域内的像素，记：上、下、左、右边界分别为、、、，假设两个连通区域边界分别为（，，，）和（，，，），且<，按照公式（1）（2）和（3），计算，，和，其中ovlp表示两个字丁重叠的长度，span表示两个字丁总的长度，dist表示两个字丁的质心的距离。

两个连通区域是否合并由nmovlp度量，其中w1和w2分别表示两个连通区域的宽度。

若nmovlp > 0，则表示两个连通区域需要合并。将整个行图像处理完成后，统计字丁长宽比的均值。若某个字丁的长宽比大于均值的1.3倍，则初步判定为粘连字丁串。图2展示了从行图像中提取的粘连字丁串。可以看出，初步提取得到的字丁串是不准确的，在后续的标记中，我们会筛选出粘连的字丁串并进行相关的标记。

## 2.2数据库数据

经过进一步的筛选，我们删除了单个字丁和重叠字丁，确定了最终的数据集。通过我们的观察，数据库中绝大部分为两个字丁的粘连。不同的字丁串存在一处或两处粘连，根据粘连形式的不同，我们将藏字粘连形式分为了六类，表格展示了粘连字丁串不同的粘连形式及对应的例子。根据粘连点和粘连字丁个数，我们将数据库分为三个子集：单粘连字丁串、单粘连多字丁串、多粘连字丁串，图展示了三个子数据库的粘连字丁串图像。

为了更好的评价算法切分的准确性，我们在每个图像对应的XML文件中标注了如下信息：粘连字丁串的类别，图像的高度、长度，平均笔画宽度，以及候选粘连点。通过连接候选粘连点可构成最佳的切分路径。