

2013 高教社杯全国大学生数学建模竞赛

承 诺 书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》(以下简称为“竞赛章程和参赛规则”,可从全国大学生数学建模竞赛网站下载)。

我们完全明白,在竞赛开始后参赛队员不能以任何方式(包括电话、电子邮件、网上咨询等)与队外的任何人(包括指导教师)研究、讨论与赛题有关的问题。

我们知道,抄袭别人的成果是违反竞赛章程和参赛规则的,如果引用别人的成果或其他公开的资料(包括网上查到的资料),必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺,严格遵守竞赛章程和参赛规则,以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为,我们将受到严肃处理。

我们授权全国大学生数学建模竞赛组委会,可将我们的论文以任何形式进行公开展示(包括进行网上公示,在书籍、期刊和其他媒体进行正式或非正式发表等)。

我们参赛选择的题号是(从 A/B/C/D 中选择一项填写): B

我们的参赛报名号为(如果赛区设置报名号的话):

所属学校(请填写完整的全名): 重庆大学

参赛队员(打印并签名): 1. 杨阳

2. 刘蛰

3. 熊莹

指导教师或指导教师组负责人(打印并签名): 温罗生

(论文纸质版与电子版中的以上信息必须一致,只是电子版中无需签名。以上内容请仔细核对,提交后将不再允许做任何修改。如填写错误,论文可能被取消评奖资格。)

日期: 2013 年 9 月 16 日

赛区评阅编号(由赛区组委会评阅前进行编号):

2013 高教社杯全国大学生数学建模竞赛

编 号 专 用 页

赛区评阅编号（由赛区组委会评阅前进行编号）：

赛区评阅记录（可供赛区评阅时使用）：

评 阅 人										
评 分										
备 注										

全国统一编号（由赛区组委会送交全国前编号）：

全国评阅编号（由全国组委会评阅前进行编号）：

基于多元统计分析的印刷文件碎纸片自动拼接复原数学模型

摘要

本文综合应用系统聚类法, Fuzzy C-Means 聚类法等方法, 建立了 3 个基于多元统计分析的印刷文件碎纸片自动拼接复原数学模型, 对题目所给碎片数据进行了较好的自动拼接复原。

问题一, 利用系统聚类法对图像边界灰度向量进行分类, 建立纵切印刷文件规则碎纸片拼接复原模型, 根据模型设计出完全无人工干预的纵切印刷碎纸片拼接复原算法。首先使用 Matlab 软件对附件 1、2 中的碎纸图像进行数字图像处理, 每张图像由一个灰度数据矩阵表示。对图像边缘进行中值滤波光滑处理, 将滤波后图像的边界列向量单独提取保存, 得到 38 个样本数据。使用系统聚类法对这 38 个样本进行分类, 得到聚类分析结果。将结果分为 19 类, 样本两两匹配。匹配复原图像, 对复原图像增强后保存。整个过程由设计的程序实现, 无人工干预。其中附件 1 中文图像复原后碎片序号从左到右依次为: 8、14、12、15、3、10、2、16、1、4、5、9、13、18、11、7、17、0、6; 附件 2 英文图像复原后碎片序号从左到右依次为: 3、6、2、7、15、18、11、0、5、1、9、13、10、8、12、14、17、16、4。附件 1、2 的碎纸片复原完全图分别见附录 1.1 和 1.2。

问题二, 在问题一的基础上, 根据印刷文字的特征, 对模型和算法进行改进, 建立横纵切印刷文件规则碎纸片的拼接复原模型。根据模型设计出计算机半自动化的横纵切印刷碎纸片拼接复原算法。首先涂黑碎纸片图像中有文字的部分, 利用 Fuzzy C-Means 聚类法对涂黑图像聚类, 然后利用系统聚类法把由 Fuzzy C-Means 聚类算法聚成一类的图像进行左右边界灰度向量比对, 类似于第一问。将两两聚成一类的图片拼接在一起后使用系统聚类法对两两拼接的图像上下边界进行聚类分析, 得到可靠的两两聚成一类的结果, 并可以得到纸条行与行之间的位置信息。将系统聚类分别不出来的图像, 根据行纸带空白间隙拼接算法, 对行间隔进行比对匹配, 同时得到纸条行与行之间的位置信息。最后由所有已经拼接好的图像和计算机给出的参考信息进行整合, 得碎纸片拼接复原的完全图, 及完全图中图片序列号。其中附件 3 的中文图像复原后的完整图与图片序列号表见附录 1.3; 附件 4 的中文图像复原后的完整图与图片序列号表见附录 1.4。

问题三, 在问题一与问题二模型与算法的基础上, 根据英文印刷特点, 建立英文印刷文字双面打印文件规则碎片的拼接与复原模型。根据模型设计出计算机半自动化的拼接复原算法, 其基本算法与问题二模型算法基本一致, 需要更多的人工干预。首先将附件 5 中的 xxxa 文件与 xxxb 文件分开, 分别导入 Matlab 工作空间。加入正反识别程序算法, 由设计的程序辨识出纸张文件的正反两面后, 按照模型二的算法对纸张文件的正反两面分别拼接复原。

关键词: 系统聚类法 Fuzzy C-Means 聚类法 Matlab 软件 边界灰度向量
空白间隙拼接算法 自动拼接 半自动拼接

一. 问题的重述

破碎文件的拼接在司法物证复原、历史文献修复以及军事情报获取等领域都有着重要的应用。传统上，拼接复原工作需由人工完成，准确率较高，但效率很低。特别是当碎片数量巨大，人工拼接很难在短时间内完成任务。随着计算机技术的发展，人们试图开发碎纸片的自动拼接技术，以提高拼接复原效率。请讨论以下问题：

1. 对于给定的来自同一页印刷文字文件的碎纸机破碎纸片（仅纵切），建立碎纸片拼接复原模型和算法，并针对附件 1、附件 2 给出的中、英文各一页文件的碎片数据进行拼接复原。如果复原过程需要人工干预，请写出干预方式及干预的时间节点。复原结果以图片形式及表格形式表达。

2. 对于碎纸机既纵切又横切的情形，请设计碎纸片拼接复原模型和算法，并针对附件 3、附件 4 给出的中、英文各一页文件的碎片数据进行拼接复原。如果复原过程需要人工干预，请写出干预方式及干预的时间节点。复原结果表达要求同上。

3. 上述所给碎片数据均为单面打印文件，从现实情形出发，还可能有双面打印文件的碎纸片拼接复原问题需要解决。附件 5 给出的是一页英文印刷文字双面打印文件的碎片数据。请尝试设计相应的碎纸片拼接复原模型与算法，并就附件 5 的碎片数据给出拼接复原结果，结果表达要求同上。

二. 模型假设及符号说明

2.1 模型的假设

- (1) 假设题目所给的数据真实可靠；
- (2) 假设所拼接的碎纸片均为规则几何图形；

2.2 符号说明

U ：隶属度矩阵；

V ：聚类中心矩阵；

G ：系统聚类形成的类；

x_i ：第 i 个灰度向量样本 ($i=1,2,\dots,n$)；

X : 观测数据矩阵, 即灰度矩阵;

d_{ij} : 灰度向量样本 x_i 与 x_j 之间的距离;

c : n 个灰度向量样本划分的类数;

D_{KL} : 类 G_K 与类 G_L 间的距离;

m : 灰度向量样本 x_k 属于第 i 类的权重;

$J(U, V)$: 各类中灰度向量样本到聚类中心的加权平方距离和;

三. 问题分析

3.1 问题一的分析

附件 1 与附件 2 中分别为来自同一页印刷文件的碎纸机纵切破碎纸片, 形状规则。将碎纸片拼接复原, 就是将切成的纸条按文字书写规则左右相互匹配在一起, 复原图像。首先对附件中的 bmp 格式图片进行数字图像处理^[3], 将图片导入 Matlab 元胞数组, 每张图片由一个灰度数据矩阵表示, 考虑到图像在图像的采集、传递和转换过程中会加入一些噪声, 使图像模糊、失真和有噪音。而图像灰度值不能突变, 只能渐变。可以使用中值滤波^[1]保护边缘信息, 平滑噪音。将滤波后图像的边界列向量单独提取保存, 得到 38 个样本数据。使用系统聚类法对这 38 个样本进行分类, 建立印刷文件规则纵切碎纸片拼接复原模型, 得到聚类分析结果。将结果分为 19 类, 每一类的样本相互匹配, 即可复原图像。

具体步骤如下:

- (1) 数字图像处理: 对附件碎纸片图像进行采样和量化, 将图像数据导入 Matlab 工作空间, 每张图像由一个 1980×72 的灰度数据矩阵表示, 共有 19 个这样的矩阵。将这 19 个矩阵存放到 1×19 的元胞数组中;
- (2) 图像中值滤波处理;
- (3) 提取样本数据: 将滤波后图像的边界灰度列向量单独提取保存, 存入 2×19 的元胞数组中, 第一行为每张图像的左边界, 即灰度数据矩阵的第 1 列, 第二行为每张图像的右边界, 即灰度数据矩阵的第 72 列, 共 38 个样本数据。
- (4) 系统聚类分析: 使用系统聚类法对这 38 个样本进行分类, 得到聚类分析结果。
- (5) 复原图片并保存: 将结果分为 19 类, 每一类的样本相互匹配, 由于完整图的左边界灰度向量与右边界的灰度向量均为 0 向量, 其相似度很高, 在聚类图中可直接观测到。由此可设计算法程序, 自动复原图像, 得到完整的图片。对复原图片进行增强后保存图片。

3.2 问题二的分析

对于既纵切又横切的印刷文件规则碎纸片拼接复原问题, 在问题一的基础上, 对模型和算法进行改进。纵切又横切的碎纸片既要考虑左右匹配, 也要考虑上下匹配。并且碎纸片四个边的灰度信息相当少, 直接采用问题一的模型拼接效果会很差。可以从印刷文字的特征进行考虑。对于印刷文件, 印刷文字按行规则排列, 相当于将一个个高度相等的方块排成一行。但印刷文件列与列之间没有这种规律, 所以此处首先考虑将出现字体笔画的行完全涂黑, 形成一条黑带, 然后使用 Fuzzy C-Means 聚类法^[3]将这些黑带碎

纸片集按照一定的准则划分为若干个子集。对附件 3 的中文纵横切碎纸片，由于每页纸被切为 11×19 个碎片，所以可以将这些黑带碎纸片集按照一定准则划分为 11 个子集，即分为 11 行。印刷文件横向文字与文字之间紧密挨在一起，由于文章书写要求，标点符号等规则，从纵向看很混乱，没有特别的规律可言，但是在纵切方向，即碎片左右两边界，在对分为同行的碎纸片边界灰度进行匹配分类时，其灰度信息是比较可观的。可以使用模型一中的系统聚类法将属于各行的碎纸片左右匹配，复原完整的行纸带。之后的工作便是将 11 条行纸带使用系统聚类的方法拼接复原。此处有一个问题，即印刷文字行与行之间存在宽度基本一致的空白间隙，在横切的时候，很有可能切到很多空白间隙，此时仍使用模型一的算法，即纸带边界灰度向量的系统聚类法，将不能实现纸带的行与行之间的正确匹配。但是由于文字的每行与行之间的宽度基本一致，可以由此设计算法，使得相匹配的行纸带的上边界空白间隙宽度与下边界空白间隙的宽度之和在一个数值范围内，当宽度和在这个范围内时，认为其相互匹配。由此可以寻找到正确的匹配，将碎纸片拼接复原。

3.3 问题三的分析

问题一与问题二所建立的两个碎纸片拼接复原模型是相互嵌套的，问题三模型将包含这两个问题建立的模型算法。首先将附件 5 中的 xxxa 文件与 xxxb 文件分开，分别导入 Matlab 工作空间。设计程序辨识 xxxa 文件中的图片是否为纸张文件的同一面，如果是同一面，则按照模型二的算法对文件的 a、b 两面分别拼接复原。如果不是同一面，则将另外设计程序，找出纸张文件的正反两面后，按照模型二的算法对纸张文件的正反两面分别拼接复原。

四. 模型的准备

4.1 数据的处理

本题目所给数据全部为 bmp 图像数据，首先应将这些数据转换为程序可读数据，进行数字图像处理。对附件碎纸片图像进行采样和量化，将图像数据导入 Matlab 工作空间，每张图像由一个灰度数据矩阵表示。其中附件 1、2 分别有 19 个这样的矩阵，矩阵大小为 1980×72 。将这 19 个矩阵存放到 1×19 的元胞数组中，数据结构转换如图 1 所示。其他附件的图像数据做同样转换。

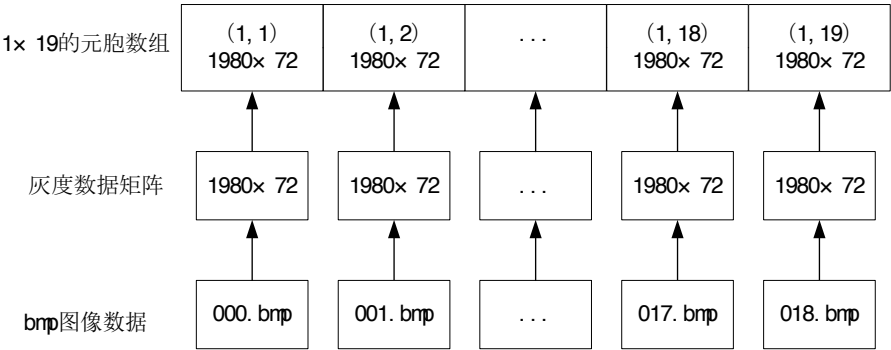


图 1 附件 1 数据结构转换图

由于图像在图像的采集、传递和转换过程中会加入一些噪声，使图像模糊、失真和有噪音。而图像灰度值不能突变，只能渐变。由此使用中值滤波保护边缘信息，平滑噪音。在对图像进行中值滤波处理后，将图像的边界灰度列向量单独提取保存，存入两行的元胞数组中，第一行为每张图片的左边界，即灰度数据矩阵的第 1 列，第二行为每张图片的右边界，即灰度数据矩阵的第 72 列。其中附件 1 图像的边界灰度列向量可存入 2×19 的元胞数组中，共 38 个样本数据，数据提取如图 2 所示。对附件 2 的图像数据做同样处理。其他附件中的图像数据在进行数据结构转换后将根据具体模型算法具体处理，在后面的模型建立与求解中将具体描述。

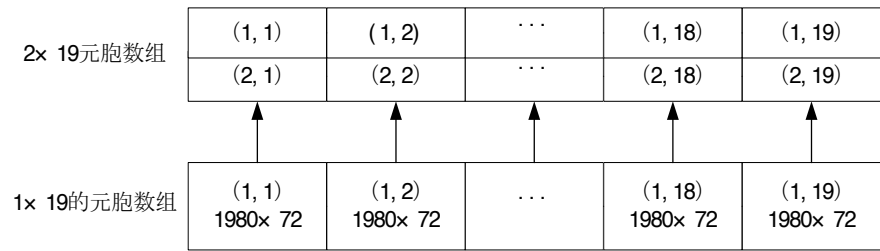


图 2 附件 1 数据提取图

4.2 数据分析与模型预测

对附件 1 中 000.bmp 图像数据进行分析知，000.bmp 图像数据的右边与 006.bmp 图像数据的左边能够匹配在一起。在进行中值滤波处理后，作图比较 000.bmp 图片的右边界灰度向量（即像素）与 006.bmp 的左边界灰度向量，如图 3 所示。（程序见附录）

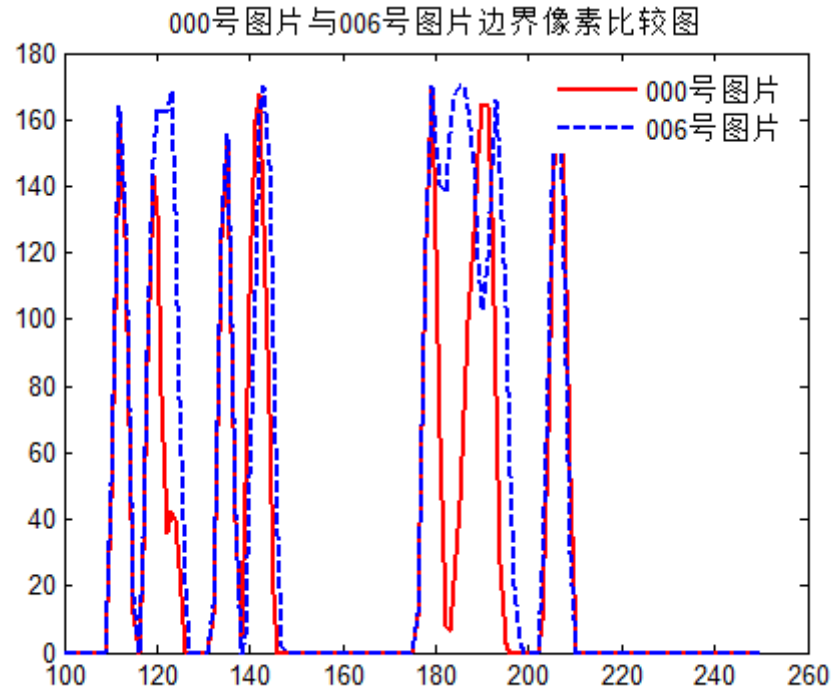


图 3 000. bmp 图片与 006. bmp 图片边界像素比较图

分析图 3 可知，两灰度向量在汉字笔画匹配的地方元素值相等或相近，当然也存在

对应向量元素差异很大的现象。但是，在汉字笔画匹配的地方，对应向量元素基本上没有差异。由此预测，可建立多元统计分析模型，采用多元统计分析中的系统聚类法对图片边界灰度向量样本进行分类，使样本两两匹配。由于完整图的左边界灰度向量与右边界灰度向量均为 0 向量，其相似度很高，在聚类图中可直接观测到。由此可设计算法程序，拼接复原图像，得到完整的图片。对进行滤波处理后的图像得到的完整图进行增强后保存，未进过滤波处理的图像得到的完整图直接保存。

五. 模型的建立与求解

5.1 问题一模型的建立与求解

5.1.1 纵切印刷文件规则碎纸片拼接复原模型的建立

在数据处理后，使用系统聚类法对样本进行分类，建立基于多元统计分析^{[3][4]}的纵切印刷文件规则碎纸片拼接复原模型。

(1) 聚类分析中的距离：

聚类分析中常用的距离有 Minkowski 距离，Lance 距离，Mahalanobis 距离，斜交空间距离等。此处采用系统聚类法中的 Mikowski 距离。

第 i 个样本 x_i 与第 j 个样本 x_j 之间的 Minkowski 距离定义为：

$$d_{ij}(q) = \left[\sum_{k=1}^P (x_{ik} - x_{jk})^q \right]^{1/q}, i = 1, 2, \dots, n; j = 1, 2, \dots, n$$

其中 q 为正整数。特别的：

当 $q = 1$ 时，称 $d_{ij}(1) = \sum_{k=1}^P |x_{ik} - x_{jk}|$ 为绝对值距离。

当 $q = 2$ 时，称 $d_{ij}(2) = \left[\sum_{k=1}^P (x_{ik} - x_{jk})^2 \right]^{1/2}$ 为欧氏距离。

当 $q \rightarrow \infty$ 时，称 $d_{ij}(\infty) = \max_{1 \leq k \leq p} |x_{ik} - x_{jk}|$ 为切比雪夫距离。

在本问的系统聚类法中采用的是 Minkowski 距离中 $q = 2$ 的情况，即样本间的欧式距离。

(2) 系统聚类的基本思想

聚类开始时将 n 个边界灰度向量样品各自作为一类，并规定样品之间的距离和类与类之间的距离，然后将距离最近的两类合并成一个新类，计算新类与其它类之间的距离，重复进行两个最近类的合并，每次减少一类，直至所有的样品合并为一类。最后形成一个表示亲疏关系的动态聚类树，从动态聚类树上面可以清晰的看出应将样品分为几类，以及每一类所包含的样品。以 G 表示类，假定 G 中有 m 个元素（即样品），用列向量 $x_i (i = 1, 2, \dots, m)$ 来表示， d_{ij} 表示样本 x_i 和样本 x_j 间的距离， D_{KL} 表示类

G_K 与类 G_L 之间的距离。类与类之间用不同的方法定义距离，就产生了不同的系统聚

类方

法。系统聚类方法有最短距离法，最长距离法，中间距离法，重心法，类平均法，离差平方和法。在该问题中采用离差平方和法。

离差平方和法把方差分析的思想用在聚类上，同一个类内的离差平方和小，而类间的离差平方和应当大。类中各元素到类重心的平方欧氏距离之和称为类内离差平方和。设某一步 G_K 与 G_L 聚成一个新类 G_M ，则 G_K 、 G_L 、 G_M 的类内离差平方和分别为：

$$W_K = \sum_{x_i \in G_K} (x_i - \bar{x}_K)^T (x_i - \bar{x}_K)$$

$$W_L = \sum_{x_i \in G_L} (x_i - \bar{x}_L)^T (x_i - \bar{x}_L)$$

$$W_M = \sum_{x_i \in G_M} (x_i - \bar{x}_M)^T (x_i - \bar{x}_M)$$

它们反映了类内元素的分散程度。 G_K 与 G_L 聚成一个新类 G_M ，类内的离差平方和会有所增加，即 $W_M - (W_K + W_L) > 0$ ，若 G_K 与 G_L 靠近，则增加的离差平方和较小，于是定义 G_L 与 G_K 的平方距离为：

$$D_{KL}^2 = W_M - (W_K + W_L) = \frac{n_K n_L}{n_M} (\bar{x}_K - \bar{x}_L)^T (\bar{x}_K - \bar{x}_L)$$

类间平方距离的递推公式为：

$$D_{MJ}^2 = \frac{n_J + n_K}{n_J + n_M} D_{KJ}^2 + \frac{n_J + n_L}{n_J + n_M} D_{LJ}^2 - \frac{n_J}{n_J + n_M} D_{KL}^2$$

(3) 模型算法流程

- 1) 将碎片图像数据导入 Matlab 工作空间，数据保存为 mat 格式；
- 2) 对数据进行处理；
- 3) 使用系统聚类法对 38 个样本进行分类：
 - [1] 调用 Matlab 中的 pdist 函数计算构成样本对的样本之间的欧式距离；
 - [2] 调用 linkage 函数，利用离差平方和法创建系统聚类数；
 - [3] 调用 dendrogram 函数作聚类树形图；
 - [4] 调用 cluster 函数，由系统聚类树矩阵创建聚类，并输出聚类结果。

5.1.2 印刷文件规则纵切碎纸片拼接复原模型的求解

- (1) 对附件 1 中文规则碎片图像边界灰度数据采用欧式距离离差平方和法聚类结果：

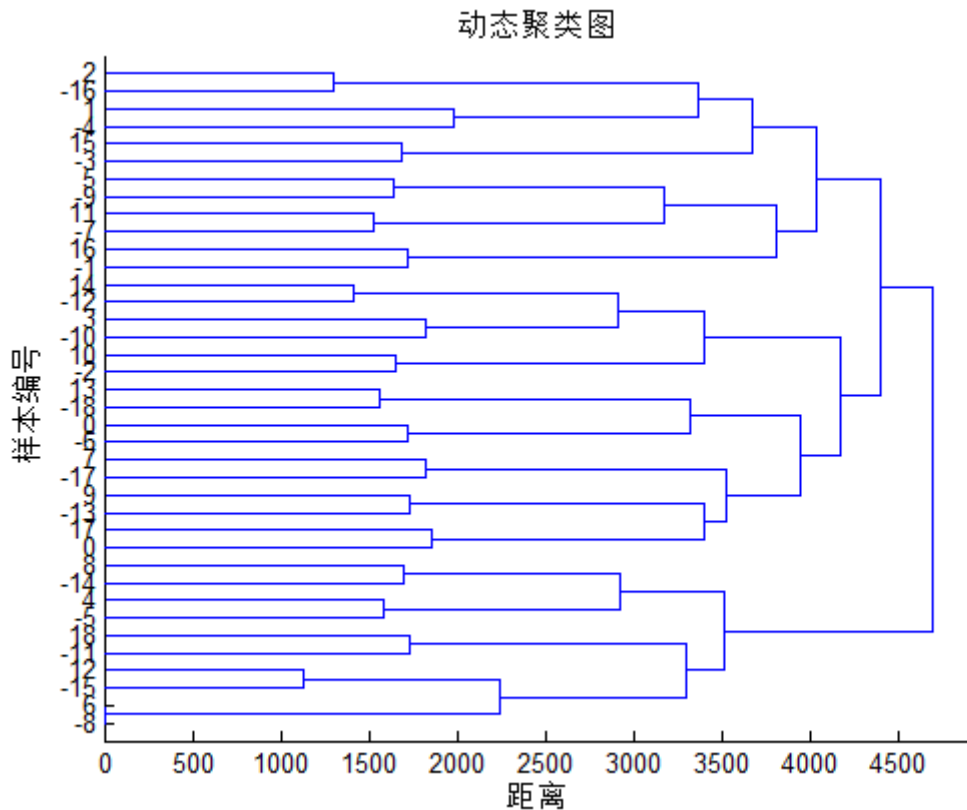


图 4 中文规则碎片边界灰度动态聚类图

注：负号表示该样本为图像左边界灰度向量数据，无负号表示该样本为图像右边界灰度向量数据。

设计程序时将聚类结果分为 19 类，结合图 4 可知分类结果如表 1 所示：

表 1 中文规则碎片边界灰度聚类分类结果

类别	1	2	3	4	5	6	7	8	9	10
右边界	6	8	14	12	15	3	10	2	16	1
左边界	-8	-14	-12	-15	-3	-10	-2	-16	-1	-4
类别	11	12	13	14	15	16	17	18	19	
右边界	4	5	9	13	18	11	7	17	0	
左边界	-5	-9	-13	-18	-11	-7	-17	0	-6	

分析聚类结果知，-8 与 6（即 008.bmp 图像左边界与 006.bmp 图像右边界）相似度极高。由常识知，这是因为碎纸片完全图的左边界与右边界的边界灰度向量均为 0 向量。所以 008.bmp 碎纸图片应该位于该页纸的最左边，006.bmp 碎纸图片应该位于该页纸的最右边。由此可设计程序 Resultshow.m（见附录）将同一类灰度数据向量样本自动匹配在一起，输出碎纸图片完全图。（见附录 1.1）附件 1 中文碎片的拼接结果序号如表 2 所示。

表 2 附件 1 中文碎片拼接结果序号表

序号	8	14	12	15	3	10	2	16	1	4	5	9	13	18	11	7	17	0	6
----	---	----	----	----	---	----	---	----	---	---	---	---	----	----	----	---	----	---	---

(2)对附件 2 英文规则碎片图像边界灰度数据采用欧式距离离差平方和法聚类结果：

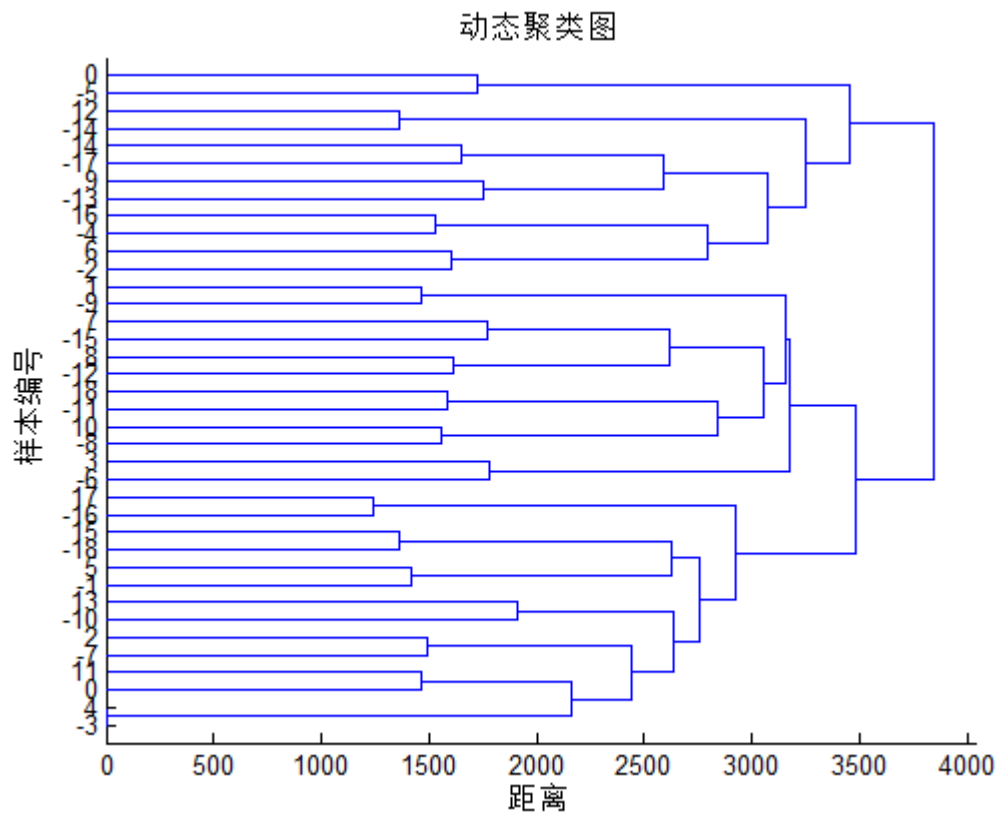


图 5 英文规则碎片边界灰度动态聚类图

同样设计程序时将聚类结果分为 19 类，结合图 5 可知分类结果如表 3 所示：

表 3 英文规则碎片边界灰度聚类分类结果

类别	1	2	3	4	5	6	7	8	9	10
右边界	4	3	6	2	7	15	18	11	0	5
左边界	-3	-6	-2	-7	-15	-18	-11	0	-5	-1
类别	11	12	13	14	15	16	17	18	19	
右边界	1	9	13	10	8	12	14	17	16	
左边界	-9	-13	-10	-8	-12	-14	-17	-16	-4	

分析聚类结果知，与对附件 1 聚类结果类似。-3 与 4（即 003.bmp 图像左边界与 004.bmp 图像右边界）相似度极高。所以 003.bmp 碎纸图片应该位于该页纸的最左边，004.bmp 碎纸图片应该位于该页纸的最右边。将同一类灰度数据向量样本自动匹配在一起，输出碎纸图片完全图。（见附录 1.2）附件 2 英文碎片的拼接结果序号如表 4 所示。

表 4 附件 1 英文碎片拼接结果序号表

序号	3	6	2	7	15	18	11	0	5	1	9	13	10	8	12	14	17	16	4
----	---	---	---	---	----	----	----	---	---	---	---	----	----	---	----	----	----	----	---

5.2 问题二模型的建立与求解

5.2.1 横纵切印刷文件规则碎纸片的拼接复原模型的建立

由对问题的分析知，首先将碎纸片中出现字体笔画的行完全涂黑，形成一条黑带，使用 Fuzzy C-Means 聚类法将这些黑带碎纸片集划分为 11 个子集。然后使用模型一中的系统聚类法将属于各行的碎纸片左右匹配，复原完整的行纸带，得到 11 条行纸带。将 11 条行纸带使用系统聚类的方法拼接复原。此处有一个问题，即印刷文字行与行之间存在宽度基本一致的空白间隙，在横切的时候，很有可能切到很多空白间隙，此时仍使用模型一的算法，即纸带边界灰度向量的系统聚类法，将只能实现少数未被切到空白间隙的行纸带之间的匹配，对于切到空白间隙的行纸带则不能正确匹配。但是文字的 row 与 row 之间的宽度基本一致，可以由此设计算法，给文字的 row 与 row 之间的空白间隙宽度设置一个上限值与一个下限值，使得相匹配的行纸带的上边界空白间隙宽度与下边界空白间隙的宽度之和在两个数值之间。由此可以寻找到正确的匹配，将碎纸片拼接复原。

(1) 碎纸片文件的行的涂黑

将印刷文件碎纸片中出现字体笔画的行完全涂黑，即将碎纸片图像导入 Matlab 工作空间后，每一张图像由一个灰度数据矩阵表示。灰度数据矩阵的任意行向量中若有灰度值不为 255，就将该行向量全部变为 0 灰度值涂黑。以附件 3 中的图片为例，如图 6 所示，以 95 号、96 号及 190 号图片涂黑为例。

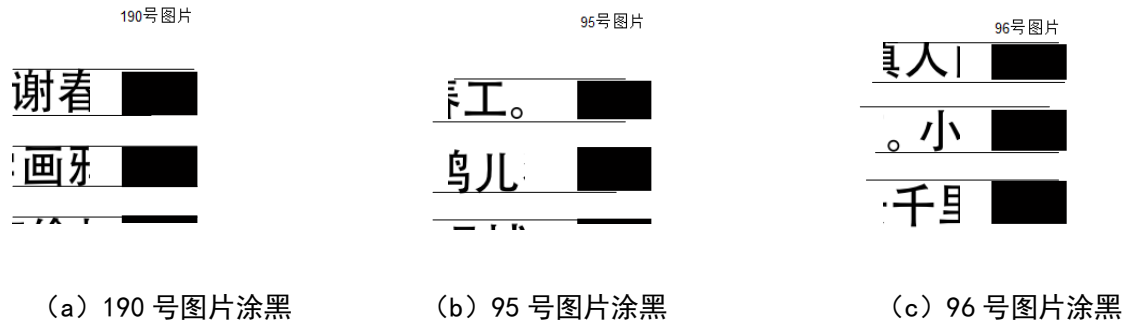


图 6 碎片图像的涂黑

(2) Fuzzy C-Means 聚类法 原理

给定样本观测数据矩阵

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}$$

其中， X 的每一行为一个样品，每一列为一个变量的 n 个观测值。模糊聚类就是将 n 个样品划分为 c 类 ($2 \leq c \leq n$)，记 $V = \{v_1, v_2, \dots, v_c\}$ 为 c 个类的聚类中心，其中

$v_i = (v_{i1}, v_{i2}, \dots, v_{ip}) (i=1, 2, \dots, c)$ 。在模糊划分中，每一个样品不是严格地划分为某一类，而是以一定的隶属度属于某一类。

令 u_{ik} 表示第 k 个样品 x_k 属于第 i 类的隶属度，其中 $0 \leq u_{ik} \leq 1, \sum_{i=1}^n u_{ik} = 1$ 。定义目标函

数

$$J(U, V) = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^m d_{ik}^2$$

其中, $U = (u_{ik})_{c \times k}$ 为隶属度矩阵, $d_{ik} = \|x_k - v_i\|$ 。 $J(U, V)$ 表示各类中样品到聚类中心的加权平方距离之和, 权重是样本 x_k 属于第 i 类的隶属度 m 次方。Fuzzy C-Means 聚类法的聚类准则是求 U, V , 使得 $J(U, V)$ 取得最小值。Fuzzy C-Means 聚类法的具体步骤如下:

- 1) 确定类的个数 c , 幂指数 $m > 1$ 和初始隶属度矩阵 $U^{(0)} = (u_{ik}^{(0)})$, 令 $l = 1$ 表示第一步迭代。
- 2) 通过下式计算第 l 步的聚类中心 $V^{(l)}$:

$$v_i^l = \frac{\sum_{k=1}^n (u_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^n (u_{ik}^{(l-1)})^m}, i = 1, 2, \dots, c$$

- 3) 修正隶属度矩阵 $U^{(l)}$, 计算目标函数值 $J^{(l)}$ 。

$$u_{ik}^{(l)} = 1 / \sum_{j=1}^c (d_{ik}^{(l)} / d_{jk}^{(l)})^{\frac{2}{m-1}}, i = 1, 2, \dots, c; k = 1, 2, \dots, n$$

$$J^{(l)}(U^{(l)}, V^{(l)}) = \sum_{k=1}^n \sum_{i=1}^c (u_{ik}^{(l)})^m (d_{ik}^{(l)})^2$$

其中, $d_{ik}^{(l)} = \|x_k - v_i^{(l)}\|$ 。

- 4) 对给定的隶属度终止容限 $\varepsilon_u > 0$, 当 $\max \{ |u_{ik}^{(l)} - u_{ik}^{(l-1)}| \} < \varepsilon_u$ 时, 停止迭代, 否则 $l = l + 1$, 然后转第二步。

经过以上步骤的迭代之后, 可以求得最终的隶属度矩阵 U 和聚类中心 V , 使得目标函数 $J(U, V)$ 的值达到最小。根据最终的隶属度矩阵 U 中元素的取值可以确定所有样品的归属, 当 $u_{jk} = \max_{1 \leq i \leq c} \{u_{ik}\}$ 时, 可将样品 x_k 归为第 j 类。

(3) 行纸带空白间隙拼接算法设计

假设有已经拼接好的两条行纸带, 并且这两条行纸带相互匹配, 如图 7 所示。在行

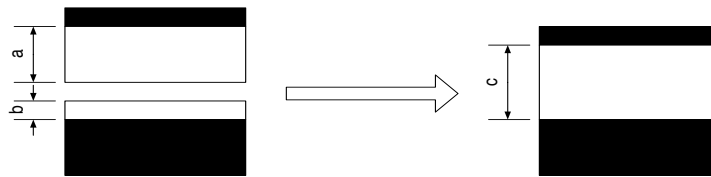


图 7 两条上下相互匹配的行纸带的匹配图

纸带的系统聚类分析中, 这样的纸带是不能被正确匹配的, 将这样的纸带提取出来。根据印刷文字的 row 与 row 之间的空白间隙宽度基本一致, 可以设计算法: 令位于上边的行纸带的空白间隙宽度为 a , 与其相匹配的位于下边的行纸带的空白间隙宽度为 b , 将这两条纸带相匹配后, 文字间的空白间隙为 $c=a+b$ 。此处的宽度用像素表示, 由统计分析知, c 的像素在 26~30 之间。可设计程序算法, 若 $a+b$ 的值在 26~33 之间, 就认为这两条纸带是相互匹配的, 将两两相互匹配的图像拼接在一起。

(4) 模型算法流程:

- 1) 涂黑碎纸片图像中有文字的部分;
- 2) 利用 Fuzzy C-Means 聚类法对涂黑图像聚类;
- 3) 利用系统聚类法把由 Fuzzy C-Means 聚类算法聚成一类的图像进行左右边界灰度向量比对, 类似于第一问的算法, 保存两两匹配的图像;
- 4) 将两两聚成一类的图片拼接在一起后使用系统聚类法对两两拼接的图像上下边界进行聚类分析, 得到可靠的两两聚成一类的结果并保存得到的纸条行与 row 之间的位置信息;
- 5) 将系统聚类分别不出来的图像, 根据行纸带空白间隙拼接算法, 对行间隔进行比对匹配, 同时得到纸条行与 row 之间的位置信息并保存。
- 6) 最后由所有已经拼接好的图像和计算机给出的参考信息进行整合, 得到附件 3、4 中横纵切印刷文件规则碎纸片拼接复原的完全图, 及完全图中图片序列号。

5.2.2 横纵切印刷文件规则碎纸片的拼接复原模型的求解

(1) 附件 3 中横纵切中文印刷文件规则碎纸片拼接复原步骤如下:

- 1) 把碎纸片图像中有文字的部分涂黑;
- 2) 利用 Fuzzy C-Means 聚类法对涂黑图像聚类, 分成 11 类。聚为一类的图像在同一行的可能性很大, 这 11 类代表 11 行, 但各行是完全图从上到下的第几行还不能确定。需要说明的是, Fuzzy C-Means 聚类算法不是很稳定, 依赖于参数 m 的选取。但是在此处对图像聚类多次运行后, 发现聚类结果变化不大。其中一种聚类结果见表 4, 其中 $di(i=1,2,\dots,11)$ 代表第 i 类。

表 5 Fuzzy C-Means 对中文碎纸片的聚类分析结果

d1 类	d2 类	d3 类	d4 类	d5 类	d6 类	d7 类	d8 类	d9 类	d10 类	d11 类
16	7	8	3	1	6	13	5	2	4	34
21	32	9	12	18	19	15	10	11	40	42
66	45	24	14	23	20	17	29	22	60	43
106	53	25	31	26	36	27	37	28	89	47
109	56	35	39	30	52	33	44	49	101	58
110	68	38	51	41	61	71	48	54	102	77

139	70	46	73	50	63	80	55	57	108	84
145	93	74	82	62	67	83	59	65	113	90
150	126	81	107	76	69	85	64	91	114	94
157	137	88	115	86	72	132	75	95	117	97
173	138	103	128	87	78	133	92	118	119	112
181	153	105	134	100	79	152	98	129	123	121
184	158	122	135	120	96	156	104	141	125	124
187	166	130	159	142	99	165	111	143	140	127
197	174	148	160	147	116	170	171	178	146	136
204	175	161	169	168	131	182	172	186	151	144
	196	167	176	179	162	198	180	188	154	149
	208	189	199	191	163	200	201	190	155	164
	0	193	203	195	177	202	206	192	185	183
						205			194	
									207	

在使用 Fuzzy C-Means 聚类法对涂黑图像聚类之前的预测效果是将涂黑的图像分为 11 类，每一类刚好包含 19 个元素，即 19 张图像。但分析表中的数据可知，聚类结果并非如预期一样完美，有三类出现异常：一类中只有 16 个元素，一类中有 20 个元素，最后一类有 21 个元素。虽然如此，其余 8 类都和预期一致，说明 Fuzzy C-Means 聚类算法对涂黑图像的聚类效果较好。

3) 利用系统聚类法把由 Fuzzy C-Means 聚类算法聚成一类的图像进行左右边界灰度向量比对，类似于第一问。将同一行的图像尽量两两聚类，即一张图像的左边界（-）与另一张图像的右边界（+）聚成一类。若聚成的一类中元素数目不为 2，则放弃结果，以保证聚类尽可能准确。并将两两聚成一类的图片拼接在一起。

4) 将所有行中两两拼接的图像以元胞数组数据结构的方式保留，图像数据结构保存如图 8 所示。

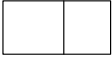

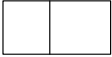
 i ng	 i ng	 i ng	...
[10, 104]	[64, 111]	[27, 64]	...
1	1	1	...

图 8 图像数据结构

其中第一行为图像矩阵，第二行为两个相互匹配的图片编号向量，一列表示一个元胞。

5) 在上述图像数据结构的基础上对图像上下边界聚类，随着上下边界聚类结果的确定，不同行之间的连接关系也会呈现出来。聚类方法如下：

[1] 使用系统聚类法对两两拼接的图像上下边界进行聚类分析，聚类图如图 9 所示。

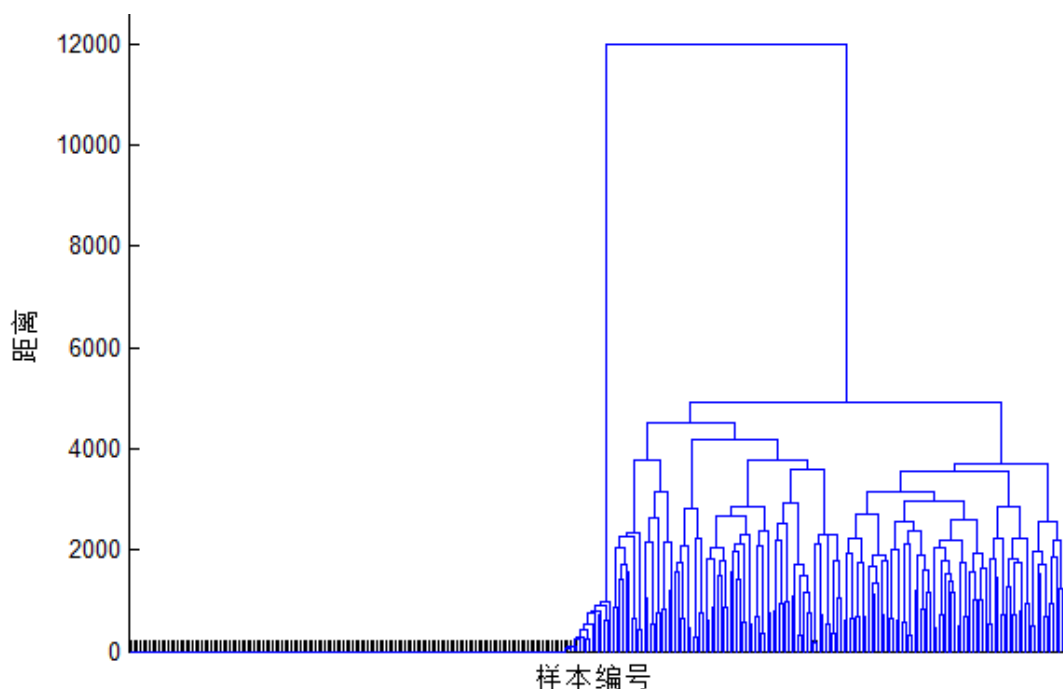


图 9 两两拼接图像上下边界灰度聚类图

分析聚类图可知，大量的边界在文字行与行之间的空白间隙被切开，边界上无任何信息可以利用。系统聚类只保留可靠的两两聚成一类的结果，同时保存无法识别其边界的图像。分析两两聚成一类的分类结果，可得到哪两类纸带相邻的信息。

[2] 将系统聚类分别不出来的图像，根据行纸带空白间隙拼接算法，对行间隔进行比对，即 $26 < a + b < 30$ 时，认为两图像相互匹配，将两两匹配的图像拼接在一起。同时可以得到哪两类纸带相邻的信息。

[3] 最后给出两次聚类结果供最后的人工干预参考。

6) 由所有已经拼接好的图像和计算机给出的参考信息进行整合。得到附件 3 中横纵切中文印刷文件规则碎纸片拼接复原的完全图（见附录 1.3），图片序列号见表 6, 其中

$Di(i=1,2,\dots,11)$ 代表第 i 行，完整图中碎纸片的图片按表中位置排列。

表 6 附件 3 横纵切中文碎片拼接结果序号表

D1	49	54	65	143	186	2	57	192	178	118	190	95	11	22	129	28	91	188	141
D2	61	19	78	67	69	99	162	96	131	79	63	116	163	72	6	177	20	52	36
D3	168	100	76	62	142	30	41	23	147	191	50	179	120	86	195	26	1	87	18
D4	38	148	46	161	24	35	81	189	122	103	130	193	88	167	25	8	9	105	74
D5	71	156	83	132	200	17	80	33	202	198	15	133	170	205	85	152	165	27	60
D6	14	128	3	159	82	199	135	12	73	160	203	169	134	39	31	51	107	115	176
D7	94	34	84	183	90	47	121	42	124	144	77	112	149	97	136	164	127	58	43
D8	125	13	182	109	197	16	184	110	187	66	106	150	21	173	157	181	204	139	145
D9	29	64	111	201	5	92	180	48	37	75	55	44	206	10	104	98	172	171	59
D10	7	208	138	158	126	68	175	45	174	0	137	53	56	93	153	70	166	32	196
D11	89	146	102	154	114	40	151	207	155	140	185	108	117	4	101	113	194	119	123

(2) 附件 4 中横纵切英文印刷文件规则碎纸片拼接复原步骤与附件 3 碎纸片的拼接复原步骤一致，最后由所有已经拼接好的图像和计算机给出的参考信息进行整合，得到附件 3 中横纵切英文印刷文件规则碎纸片拼接复原的完全图和图片序列号。

在第二步中利用 Fuzzy C-Means 聚类法对涂黑图像聚类，其聚类结果如表 7 所示。

表 7 Fuzzy C-Means 对英文碎纸片的聚类分析结果

d1 类	d2 类	d3 类	d4 类	d5 类	d6 类	d7 类	d8 类	d9 类	d10 类	d11 类
5	4	5	12	1	9	3	23	2	10	15
8	32	13	48	18	19	25	33	7	16	20
14	39	24	52	31	44	27	47	11	22	36
17	64	30	72	38	56	29	86	21	35	41
26	65	37	77	50	66	34	90	49	42	43
28	67	40	81	53	82	46	96	54	55	45
60	75	51	87	63	83	92	99	61	57	73
68	104	58	89	74	93	95	109	62	71	76
70	106	59	115	85	121	98	122	112	88	79
78	147	69	124	97	126	117	156	118	105	102
80	149	94	125	120	134	127	172	119	110	108
84	154	107	128	123	141	130	185	133	114	116
91	184	111	131	129	151	163	208	142	145	135
100	190	132	177	138	152	166		162	155	136
101	204	144	193	139	157	181		168	165	140
103		150	200	153	171	186		169	183	143
113		158	0	159	176	188		179	202	161
137		167		160	182			189		173
146		178		175	194			191		180
148		201		187	205			192		199
164		206		203				197		207
170										
174										
195										
196										
198										

分析表 7 可知，采用此模型算法拼接复原英文碎纸片不是非常理想，但是图像分为 11 类，每一类都还是拥有足够的信息的，即足够的图像元素。因此仍可以使用此算法对其继续拼接，只是在将同一类图像进行拼行的时候也需要加入人工干预。人工干预完成后继续运行程序算法，到最后一步，再进行人工干预。由所有已经拼接好的图像和计算机给出的参考信息进行人为整合，得到附件 4 中横纵切英文印刷文件规则碎纸片拼接复原的完全图（见附录 1.4）和图片序列号，如表 8 所示。 $Di(i=1,2,\dots,11)$ 代表第 i 行，完整图中碎纸片的图片按表中位置排列。

表 8 附件 4 横纵切英文碎片拼接结果序号表

D1	191	75	11	154	190	184	2	104	180	64	106	4	149	32	204	65	39	67	147
D2	201	148	170	196	198	94	113	164	78	103	91	80	101	26	100	6	17	28	146
D3	86	51	107	29	40	158	186	98	24	117	150	5	59	58	92	30	37	46	127
D4	19	194	93	141	88	121	126	105	155	114	176	182	151	22	57	202	71	165	82
D5	159	139	1	129	63	138	153	53	38	123	120	175	85	50	160	187	97	203	31
D6	20	41	108	116	136	73	36	207	135	15	76	43	199	45	173	79	161	179	143
D7	208	21	7	49	61	119	33	142	168	62	169	54	192	133	118	189	162	197	112
D8	70	84	60	14	68	174	137	195	8	47	172	156	96	23	99	122	90	185	109
D9	132	181	95	69	167	163	166	188	111	144	206	3	130	34	13	110	25	27	178
D10	171	42	66	205	10	157	74	145	83	134	55	18	56	35	16	9	183	152	44
D11	81	77	128	200	131	52	125	140	193	87	89	48	72	12	177	124	209	102	115

5.3 问题三模型的建立与求解

5.3.1 英文印刷文字双面打印文件规则碎片的拼接与复原模型的建立

英文印刷文字双面打印文件规则碎片的拼接与复原模型的建立是基于问题一与问题二的模型算法的改进。加入正反辨识程序算法，由设计的程序辨识出纸张文件的正反两面后，然后按照模型二的算法对纸张文件的正反两面分别拼接复原。

加入正反辨识程序算法的程序流程：

[1] 将 xxxa 文件与 xxxb 文件分别导入 Matlab 工作，存储数据问 mat 格式；

[2] 首先使用模型二的算法，经过人工干预，辨识出 xxxa 文件的图像是否是纸张文件的同一面。

[3] 如果是，则保存 xxa 图像的完全图，并使用模型二的算法拼接复原 xxxb 文件中的图像。

[4] 如果不是，就进行人工干预，分离出不是同一面的图像，人工辨识其属于哪一面。将同一面的图像保存，运用问题二的模型进行拼接复原

5.3.1 英文印刷文字双面打印文件规则碎片的拼接与复原模型的求解

对附件 5 的英文印刷文字双面打印文件规则碎片进行拼接复原步骤如下：

(1) 将 xxxa 文件与 xxxb 文件分别导入 Matlab 工作，存储数据问 mat 格式；

(2) 使用模型二的算法，利用 Fuzzy C-Means 聚类法对涂黑图像聚类，其聚类结果如表 10 所示。

表 10 Fuzzy C-Means 对英文碎纸片的聚类分析结果

第 1 类的有	43	49	54	55	91	99	100	103	104	106	109	112	113	215	252	258	263	305	312	313	315	321
第 2 类的有	16	50	73	171	201	202	208	228	262	339	372	380	386	399	411							
第 3 类的有	17	28	64	102	116	154	179	267	292	351	367	406	416									
第 4 类的有	3	7	32	42	61	69	74	77	80	94	126	135	137	143	216	241	247	278	283	286	298	347
第 5 类的有	10	25	46	59	68	75	76	122	157	165	168	182	192	219	223	249	255	268	272	277	284	325
第 6 类的有	18	20	29	35	58	110	111	140	159	163	183	189	227	229	238	244	256	275	282	287	317	320
	334	345	349	359	398																	

6.2 模型的缺点

问题二的横纵切印刷文件规则碎纸片的拼接复原模型对是半自动的拼接复原模型，需要人为的干预，但是干预的方便程度不够，希望能够设计一个很好的人机交互界面。

参考文献

- [1] 谢中华 . MATLAB 统计分析与应用：40 个案例分析[M]. 北京：北京航空航天大学出版社，2010. 290-298, 318.
- [2] 杨丹，赵海滨，龙哲等. MATLAB 图像处理实例详解[M]. 北京：清华大学出版社，2013. 207-208.
- [3] 司守奎，孙玺菁. 数学建模算法与应用[M]. 北京：国防工业出版社，2013. 193-206
- [4] 周凯，宋军全，邬学军. 数学建模入门与提高[M]. 杭州：浙江大学出版社，2011. 12

附录

1.1 附件 1 中文拼接复原结果与序号

城上层楼叠巘。城下清淮古汴。举手揖吴云，人与暮天俱远。魂断。魂断。后夜松江月满。簌簌衣巾莎枣花。村里村北响燥车。牛衣古柳卖黄瓜。海棠珠缀一重重。清晓近帘栊。胭脂谁与匀淡，偏向脸边浓。小郑非常强记，二南依旧能诗。更有鲈鱼堪切脍，儿辈莫教知。自古相从休务日，何妨低唱微吟。天垂云重作春阴。坐中人半醉，帘外雪将深。双鬟绿坠。娇眼横波眉黛翠。妙舞蹁跹。掌上身轻意态妍。碧雾轻笼两凤，寒烟淡拂双鸦。为谁流睇不归家。错认门前过马。

我劝髯张归去好，从来自己忘情。尘心消尽道心平。江南与塞北，何处不堪行。闲离阻。谁念萦损襄王，何曾梦云雨。旧恨前欢，心事两无据。要知欲见无由，痴心犹自，倩人道、一声传语。风卷珠帘自上钩。萧萧乱叶报新秋。独携纤手上高楼。临水纵横回晚鞚。归来转觉情怀动。梅笛烟中闻几弄。秋阴重。西山雪淡云凝冻。凭高眺远，见长空万里，云无留迹。桂魄飞来光射处，冷浸一天秋碧。玉宇琼楼，乘鸾来去，人在清凉国。江山如画，望中烟树历历。省可清言挥玉尘，真须保器全真。风流何似道家纯。不应同蜀客，惟爱卓文君。自惜风流云雨散。关山有限情无限。待君重见寻芳伴。为说相思，目断西楼燕。莫恨黄花未吐。且教红粉相扶。酒阑不必看茱萸。俯仰人间今古。玉骨那愁瘴雾，冰姿自有仙风。海仙时遣探芳丛。倒挂绿毛么凤。

俎豆庚桑真过矣，凭君说与南荣。愿闻吴越报丰登。君王如有问，结袜赖王生。师唱谁家曲，宗风嗣阿谁。借君拍板与门槌。我也逢场作戏、莫相疑。晕腮嫌枕印。印枕嫌腮晕。闲照晚妆残。残妆晚照闲。可恨相逢能几日，不知重会是何年。茱萸仔细更重看。午夜风翻幔，三更月到床。簟纹如水玉肌凉。何物与侬归去、有残妆。金炉犹暖麝煤残。惜香更把宝钗翻。重闻处，余熏在，这一番、气味胜从前。菊暗荷枯一夜霜。新苞绿叶照林光。竹篱茅舍出青黄。霜降水痕收。浅碧鳞鳞露远洲。酒力渐消风力软，飕飕。破帽多情却恋头。烛影摇风，一枕伤春绪。归不去。凤楼何处。芳草迷归路。汤发云腴酳白，盏浮花乳轻圆。人间谁敢更争妍。斗取红窗粉面。炙手无人傍屋头。萧萧晚雨脱梧楸。谁怜季子敝貂裘。

序号	8	14	12	15	3	10	2	16	1	4	5	9	13	18	11	7	17	0	6
----	---	----	----	----	---	----	---	----	---	---	---	---	----	----	----	---	----	---	---

1.2 附件 2 英文拼接复原结果与序号

fair of face.

The customer is always right. East, west, home's best. Life's not all beer and skittles. The devil looks after his own. Manners maketh man. Many a mickle makes a muckle. A man who is his own lawyer has a fool for his client.

You can't make a silk purse from a sow's ear. As thick as thieves. Clothes make the man. All that glisters is not gold. The pen is mightier than sword. Is fair and wise and good and gay. Make love not war. Devil take the hindmost. The female of the species is more deadly than the male. A place for everything and everything in its place. Hell hath no fury like a woman scorned. When in Rome, do as the Romans do. To err is human; to forgive divine. Enough is as good as a feast. People who live in glass houses shouldn't throw stones. Nature abhors a vacuum. Moderation in all things.

Everything comes to him who waits. Tomorrow is another day. Better to light a candle than to curse the darkness.

Two is company, but three's a crowd. It's the squeaky wheel that gets the grease. Please enjoy the pain which is unable to avoid. Don't teach your Grandma to suck eggs. He who lives by the sword shall die by the sword. Don't meet troubles half-way. Oil and water don't mix. All work and no play makes Jack a dull boy.

The best things in life are free. Finders keepers, losers weepers. There's no place like home. Speak softly and carry a big stick. Music has charms to soothe the savage breast. Ne'er cast a clout till May be out. There's no such thing as a free lunch. Nothing venture, nothing gain. He who can does, he who cannot, teaches. A stitch in time saves nine. The child is the father of the man. And a child that's born on the Sab-

序号	3	6	2	7	15	18	11	0	5	1	9	13	10	8	12	14	17	16	4
----	---	---	---	---	----	----	----	---	---	---	---	----	----	---	----	----	----	----	---

1.3 附件 3 横纵切中文碎纸片拼接复原结果与序号

便邮。温香熟美。醉慢云鬟垂两耳。多谢春工。不是花红是玉红。一颗樱桃樊素口。不爱黄金，只爱人长久。学画鸦儿犹未就。眉尖已作伤春皱。清泪斑斑，挥断柔肠寸。嗔人问。背灯偷搵拭尽残妆粉。春事阑珊芳草歇。客里风光，又过清明节。小院黄昏人忆别。落红处处闻啼鵲。岁云暮，须早计，要褐裘。故乡归去千里，佳处辄迟留。我醉歌时君和，醉倒须君扶我，惟酒可忘忧。一任刘玄德，相对卧高楼。记取西湖西畔，正暮山好处，空翠烟霏。算诗人相得，如我与君稀。约他年、东还海道，愿谢公、雅志莫相违。西州路，不应回首，为我沾衣。料峭春风吹酒醒。微冷。山头斜照却相迎。回首向来潇洒处。归去。也无风雨也无晴。紫陌寻春去，红尘拂面来。无人不道看花回。惟见石榴新蕊、一枝开。

九十日春都过了，贪忙何处追游。三分春色一分愁。雨翻榆荚阵，风转柳花球。白雪清词出坐间。爱君才器两俱全。异乡风景却依然。团扇只堪题往事，新丝那解系行人。酒阑滋味似残春。

缺月向人舒窈窕，三星当户照绸缪。香生雾縠见纤柔。搔首赋归欤。自觉功名懒更疏。若问使君才与术，何如。占得人间一味愚。海东头，山尽处。自古空槎来去。槎有信，赴秋期。使君行不归。别酒劝君君一醉。清润潘郎，又是何郎婿。记取钗头新利市。莫将分付东邻子。西塞山边白鹭飞。散花洲外片帆微。桃花流水鳊鱼肥。主人瞋小。欲向东风先醉倒。已属君家。且更从容等待他。愿我已无当世望，似君须向古人求。岁寒松柏肯惊秋。

水涵空，山照市。西汉二疏乡里。新白发，旧黄金。故人恩义深。谁道东阳都瘦损，凝然点漆精神。瑶林终自隔风尘。试看披鹤氅，仍是谪仙人。三过平山堂下，半生弹指声中。十年不见老仙翁。壁上龙蛇飞动。暖风不解留花住。片片著人无数。楼上望春归去。芳草迷归路。犀钱玉果。利市平分沾四坐。多谢无功。此事如何到得侬。元宵似是欢游好。何况公庭民讼少。万家游赏上春台，十里神仙迷海岛。

虽抱文章，开口谁亲。且陶陶、乐尽天真。几时归去，作个闲人。对一张琴，一壶酒，一溪云。相如未老。梁苑犹能陪俊少。莫惹闲愁。且折

序号表：

D1	49	54	65	143	186	2	57	192	178	118	190	95	11	22	129	28	91	188	141
D2	61	19	78	67	69	99	162	96	131	79	63	116	163	72	6	177	20	52	36
D3	168	100	76	62	142	30	41	23	147	191	50	179	120	86	195	26	1	87	18
D4	38	148	46	161	24	35	81	189	122	103	130	193	88	167	25	8	9	105	74
D5	71	156	83	132	200	17	80	33	202	198	15	133	170	205	85	152	165	27	60
D6	14	128	3	159	82	199	135	12	73	160	203	169	134	39	31	51	107	115	176
D7	94	34	84	183	90	47	121	42	124	144	77	112	149	97	136	164	127	58	43
D8	125	13	182	109	197	16	184	110	187	66	106	150	21	173	157	181	204	139	145
D9	29	64	111	201	5	92	180	48	37	75	55	44	206	10	104	98	172	171	59
D10	7	208	138	158	126	68	175	45	174	0	137	53	56	93	153	70	166	32	196
D11	89	146	102	154	114	40	151	207	155	140	185	108	117	4	101	113	194	119	123

bath day. No news is good news.

Procrastination is the thief of time. Genius is an infinite capacity for taking pains. Nothing succeeds like success. If you can't beat em, join em. After a storm comes a calm. A good beginning makes a good ending.

One hand washes the other. Talk of the Devil, and he is bound to appear. Tuesday's child is full of grace. You can't judge a book by its cover. Now drips the saliva, will become tomorrow the tear. All that glitters is not gold. Discretion is the better part of valour. Little things please little minds. Time flies. Practice what you preach. Cheats never prosper.

The early bird catches the worm. It's the early bird that catches the worm. Don't count your chickens before they are hatched. One swallow does not make a summer. Every picture tells a story. Softly, softly, catchee monkey. Thought is already late, exactly is the earliest time. Less is more.

A picture paints a thousand words. There's a time and a place for everything. History repeats itself. The more the merrier. Fair exchange is no robbery. A woman's work is never done. Time is money.

Nobody can casually succeed, it comes from the thorough self-control and the will. Not matter of the today will drag tomorrow. They that sow the wind, shall reap the whirlwind. Rob Peter to pay Paul. Every little helps. In for a penny, in for a pound. Never put off until tomorrow what you can do today. There's many a slip twixt cup and lip. The law is an ass. If you can't stand the heat get out of the kitchen. The boy is father to the man. A nod's as good as a wink to a blind horse. Practice makes perfect. Hard work never did anyone any harm. Only has compared to the others early, diligently

序号

D1	191	75	11	154	190	184	2	104	180	64	106	4	149	32	204	65	39	67	147
D2	201	148	170	196	198	94	113	164	78	103	91	80	101	26	100	6	17	28	146
D3	86	51	107	29	40	158	186	98	24	117	150	5	59	58	92	30	37	46	127
D4	19	194	93	141	88	121	126	105	155	114	176	182	151	22	57	202	71	165	82
D5	159	139	1	129	63	138	153	53	38	123	120	175	85	50	160	187	97	203	31
D6	20	41	108	116	136	73	36	207	135	15	76	43	199	45	173	79	161	179	143
D7	208	21	7	49	61	119	33	142	168	62	169	54	192	133	118	189	162	197	112
D8	70	84	60	14	68	174	137	195	8	47	172	156	96	23	99	122	90	185	109
D9	132	181	95	69	167	163	166	188	111	144	206	3	130	34	13	110	25	27	178
D10	171	42	66	205	10	157	74	145	83	134	55	18	56	35	16	9	183	152	44
D11	81	77	128	200	131	52	125	140	193	87	89	48	72	12	177	124	209	102	115

1.5 问题一的源程序

1.5.1 图像边界灰度值比较程序

程序名	imgcompare.m	软件名	Matlab
<pre> %% 图像边界灰度值比较 %% 将图像的边界单独提取出来保存 load data1; boundarycellmatrix=cell(2,19); imgcellmatrix=cell(1,19); %% 图像导入元胞数组 imgcellmatrix{1}=img0; imgcellmatrix{2}=img1; imgcellmatrix{3}=img2; imgcellmatrix{4}=img3; imgcellmatrix{5}=img4; imgcellmatrix{6}=img5; imgcellmatrix{7}=img6; imgcellmatrix{8}=img7; imgcellmatrix{9}=img8; imgcellmatrix{10}=img9; imgcellmatrix{11}=img10; imgcellmatrix{12}=img11; imgcellmatrix{13}=img12; imgcellmatrix{14}=img13; imgcellmatrix{15}=img14; imgcellmatrix{16}=img15; </pre>			

```

imgcellmatrix{17}=img16;
imgcellmatrix{18}=img17;
imgcellmatrix{19}=img18;
%% 清除中间变量
clear img0 img1 img2 img3 img4 img5 img6 img7 img8 img9 img10 img11...
    img12 img13 img14 img15 img16 img17 img18;
%% 图像元胞数组转换
Imgcelldata=imgcellmatrix;
for j=1:19
    Imgcelldata{j}=double(Imgcelldata{j});
    Imgcelldata{j}=255-Imgcelldata{j};
end
%% 图像预处理
%中值滤波处理
for j=1:19
    I=Imgcelldata{j};
    PSF=fspecial('average',3);
    L=imfilter(I,PSF);
    Imgcelldata{j}=L;
end
clear I M N u v U V D H J K L;
%% 图像的边界左右个一条边界
for i=1:2
    for j=1:19
        if i==1
            boundarycellmatrix{i,j}=Imgcelldata{j}(:,1);
        end
        if i==2
            boundarycellmatrix{i,j}=Imgcelldata{j}(:,72);
        end
    end
end
%% 作图比较
x=Imgcelldata{1}(100:250,72);
y=Imgcelldata{7}(100:250,1);
figure('Color',[1 1 1]);
plot(100:250,x,'r-','linewidth',2)
hold on;
plot(100:250,y,'b--','linewidth',2);
axis([100 260 0 180]);
title('000 号图片与 006 号图片边界像素比较图')
legend('000 号图片','006 号图片');
hold off;

```

备注	
----	--

1.5.2 纵切规则碎纸片的边界灰度聚类及图像拼接复原程序

程序名	dataread.m	软件名	Matlab
<pre> mg0=imread('000.bmp'); img1=imread('001.bmp'); img2=imread('002.bmp'); img3=imread('003.bmp'); img4=imread('004.bmp'); img5=imread('005.bmp'); img6=imread('006.bmp'); img7=imread('007.bmp'); img8=imread('008.bmp'); img9=imread('009.bmp'); img10=imread('010.bmp'); img11=imread('011.bmp'); img12=imread('012.bmp'); img13=imread('013.bmp'); img14=imread('014.bmp'); img15=imread('015.bmp'); img16=imread('016.bmp'); img17=imread('017.bmp'); img18=imread('018.bmp');</pre>			
备注	将 bmp 图片数据读入 Matlab 工作空间，每张图片由一个灰度数据矩阵表示。		

程序名	datasort1.m	软件名	Matlab
<pre> %% 本程序针对第一问纵切的情况，将图像聚类并且给出拼接出来的图像 %% 数据导入 load data1; %%中文图象数据 %load data2; %%英文图象数据 boundarycellmatrix=cell(2,19); imgcellmatrix=cell(1,19); %% 图像导入元胞数组 imgcellmatrix{1}=img0; imgcellmatrix{2}=img1; imgcellmatrix{3}=img2; imgcellmatrix{4}=img3; imgcellmatrix{5}=img4; imgcellmatrix{6}=img5; imgcellmatrix{7}=img6; imgcellmatrix{8}=img7; imgcellmatrix{9}=img8; imgcellmatrix{10}=img9;</pre>			

```

imgcellmatrix{11}=img10;
imgcellmatrix{12}=img11;
imgcellmatrix{13}=img12;
imgcellmatrix{14}=img13;
imgcellmatrix{15}=img14;
imgcellmatrix{16}=img15;
imgcellmatrix{17}=img16;
imgcellmatrix{18}=img17;
imgcellmatrix{19}=img18;
%% 清除中间变量
clear img0 img1 img2 img3 img4 img5 img6 img7 img8 img9 img10 img11...
    img12 img13 img14 img15 img16 img17 img18;
%% 图像元胞数组转换
Imgcelldata=imgcellmatrix;
for j=1:19
    Imgcelldata{j}=double(Imgcelldata{j});
    Imgcelldata{j}=255-Imgcelldata{j};
end
%% 图像预处理
%中值滤波处理
for j=1:19
    I=Imgcelldata{j};
    PSF=fspecial('average',3);
    L=imfilter(I,PSF);
    Imgcelldata{j}=L;
end
clear I M N u v U V D H J K L;
%% 图像的边界左右各一条边界
for i=1:2
    for j=1:19
        if i==1
            boundarycellmatrix{i,j}=Imgcelldata{j}(:,1);
        end
        if i==2
            boundarycellmatrix{i,j}=Imgcelldata{j}(:,72);
        end
    end
end
%% 创建数据矩阵
M=zeros(19*2,1980);
for i=1:2
    if i==1
        for j=1:19

```

```

        M(j,:)=(boundarycellmatrix{i,j})';
    end
    else
        for j=1:19
            M(j+19,:)=(boundarycellmatrix{i,j})';
        end
    end
end
clear i j ;
%% 聚类分析
figure('Color',[1 1 1])
s=cell(1,38);
for i=1:38
    if i<=19
        s{i}=num2str(-mod(i-1,19));
    else
        s{i}=num2str(mod(i-1,19));
    end
end
y=pdist(M,'euclidean');           %计算欧式距离
z=linkage(y,'ward');               %离差平方和方法聚类
[H,T]=dendrogram(z,38,'orientation','right','labels',s); %绘制动态聚类树
ylabel('样本编号');
xlabel('距离');
title('动态聚类图');
Clss=cluster(z,'cutoff',0.1);
sortnumber=zeros(18,2);
for i=1:19
    tm=find(Clss==i);    %求第 i 类的对象
    for j=1:2
        if tm(j)<=19
            tm(j)=-tm(j);
        else
            tm(j)=mod(tm(j),20)+1;
        end
    end
    tm=reshape(tm,1,length(tm)); %变成行向量
    sortnumber(i,:)=tm;
end
clear H T M    PSF T i s y z Clss j    tm score;
%% 图象拼接处理
clear imgcellmatrix boundarycellmatrix
Resultshow(sortnumber);

```

备注

程序名	Resultshow	软件名	Matlab
<pre> %% 本函数根据边界连接序号做出拼接好的图象 function Resultshow(A) load data1 %%中文图象数据 %load data2 %%英文图象数据 %% 序号变换和图象拼接 a=zeros(19,1); x1=-A(:,1); x2=A(:,2); a(1)=x2(1); for i=2:19 a(i)=x1(x2==a(i-1)); end a=a-1; a=[a;a(1)]; a(1)=[]; imgg=[img0 img1 img2 img3 img4 img5 img6 img7 img8 img9 img10 img11 img12 img13 img14 img15 img16 img17 img18]; [m,n]=size(imgg); img=imgg; a' for i=1:19 img(:,(1+72*(i-1)):(72*(i-1)+72))=imgg(:,(1+72*a(i)):(72*a(i)+72)); end %% 图象增强并显示和保存 figure; img=imadjust(img); imshow(img) imwrite(img,'result.png'); </pre>			
备注			

1.6 问题二的源程序

程序名	colsort.m	软件名	Matlab
<pre> load data1; % shunxu=cell(11,23); shunxu=cell(1,11); rowcluster;%%%%%%%%%%%%%% %%%%%%%%%%%%%% disp('聚类均匀吗？ 1 表示可以接受， 0 表示不可以'); fff=input(""); while ~fff </pre>			


```

rowcluster;
disp('聚类均匀吗? 1 表示可以接受, 0 表示不可以');
fff=input("");
end
badsort=[];
for j=1:11
    [A,badsort{j}]=rowsort(index{j},0.72);%%%%%%%%%%
%    k=1;%k 表示 shunxu 元胞内存放正确数据的位置
    x1=-A(:,1);%除去不正常数据
    x2=A(:,2);
    x2(x1<0)=[];x1(x1<0)=[];
    x1(x2<0)=[];x2(x2<0)=[];
%    for i=1:length(x1)%计算一类的正确数据
%        imshow([I{x2(i)},I{x1(i)}]);
%        disp('正确吗? 1 正确, 0 不正确');
%        f=input("");
%        if f
%            shunxu{j,k}=[x2(i),x1(i)];
%            k=k+1;
%        end
%    end
    shunxu{j}=[x2,x1];
clear A x1 x2
end
pinjie%%%%%%%%%%
%%%%%%%%%%
%% 产生一个元胞第一行放两两拼接的图片, 第二行放对应编号, 第三行放所在的行
%%%%%%%%%% shunxu 元胞内部元素总长度测试
len=0;
for j=1:11
    len=len+length(shunxu{j});
end
%%%%%%%%%%
matchcell=cell(3,len);
kk=0;
for j=1:11
    leng=length(shunxu{j});
    for i=kk+1:kk+leng
        matchcell{2,i}=shunxu{j}(i-kk,:);
        matchcell{3,i}=j;
        matchcell{1,i}=[I{shunxu{j}(i-kk,1)},I{shunxu{j}(i-kk,2)}];
    end
    kk=leng+kk;
end
end

```

```

%% 产生上下边界元胞
ublbcell=cell(2,len);
for j=1:len
    ublbcell{1,j}=matchcell{1,j}(1,:);
    ublbcell{2,j}=matchcell{1,j}(180,:);
end

M=zeros(2*len,length(ublbcell{1,1}));
for i=1:2
    for j=1:len
        if i==1
            M(j,:)=ublbcell{1,j};
        else
            M(len+j,:)=ublbcell{2,j};
        end
    end
end

%% 聚类分析
%figure('Color',[1 1 1]);
%s=char(1,2*len);
%for i=1:2*len
%    s(i)=' ';
%end
y=pdist(M,'euclidean');           %计算欧式距离
z=linkage(y,'ward');              %离差平方和方法聚类
%[H,T]=dendrogram(z,2*len,'labels',s); %绘制动态聚类树
%xlabel('样本编号');
%ylabel('距离');
%title('动态聚类图');
Clss=cluster(z,'cutoff',0.727);
%Clss=cluster(z,'cutoff',cutoff);
sortnumber=cell(1,1);
tmm=cell(1,1);
for i=1:max(Clss)
    tm=find(Clss==i);    %求第 i 类的对象
    tmm{1}=tm;
    for j=1:length(tm)
        if tm(j)<=len
            tmm{1}(j)=tm(j);
        else
            tmm{1}(j)=-(mod(tm(j),len+1)+1);
        end
    end
end
sortnumber=[sortnumber,tmm];

```

```

end
clear H T i s y z Clss j tm tmm;
%% 输出
sortnumber=sortnumber(2:length(sortnumber));
badsort=[];
k=0;
for j=1:length(sortnumber)
    if length(sortnumber{j})==2
        k=k+1;
    end
end
output1=zeros(k,2);
ii=1;
jj=1;
for j=1:length(sortnumber)
    if length(sortnumber{j})==2
        output1(ii,:)=sortnumber{j};
        ii=ii+1;
    else
        % fprintf('不能区别\n');
        sortnumber{j};
        badsort{jj}=sortnumber{j};
        jj=jj+1;
    end
end
result1=zeros(size(output1));
for i=1:length(output1(:,1))
    for j=1:length(output1(1,:))
        if output1(i,j)>0
            result1(i,j)=matchcell{3,abs(output1(i,j))};
        else
            result1(i,j)=-matchcell{3,abs(output1(i,j))};
        end
    end
end
end

%% 边缘白条检测进一步 行匹配
%%初始化变量
unrecongize=[];
for i=1:length(badsort)
    unrecongize=[unrecongize,((badsort{i}{:}))];
end

```

```

unrecongize=unrecongize(2:length(unrecongize));
suanhangjianju  %%%%%%%%%%%
%%%%%%%%%%
lenu=length(find(unrecongize>0));
lenl=length(find(unrecongize<0));
m=find(unrecongize>0);
n=find(unrecongize<0);
for i=1:lenu
    boundaryu(i)=wide{unrecongize(m(i))}(1);
end
for i=1:lenl
    boundaryl(i)=wide{abs(unrecongize(n(i)))}(length(wide{abs(unrecongize(n(i)))}));
end
%%%%%%%%%%                                计 算 行 间 距 匹
配                                %%%%%%%%%%
%len2=length(boundaryu);

output2=[];
for i=1:lenu
    for j=[1:i,i:lenl]
        if boundaryu(i)+boundaryl(j)>=26 && boundaryu(i)+boundaryl(j)<=30

output2=[output2 ;unrecongize(m(i)),unrecongize(n(j))];        %%%%%%%%%%
%%%%%%%%%%问题
        end
    end
end
output2=output2(2:length(output2),:);
result2=zeros(size(output2));
for i=1:length(output2(:,1))
    for j=1:length(output2(1,:))
        if output2(i,j)>0
            result2(i,j)=matchcell{3,abs(output2(i,j))};
        else
            result2(i,j)=-matchcell{3,abs(output2(i,j))};
        end
    end
end
end

%% 聚类结果用户评价
result=result1;
output = output1;
disp(' number    UpRow    LowRow    ')
disp([(1:length(result))',result])

```

<pre>temp=input('行聚类结果可以接受第几个?\n(向量)'); sqre=cell(2,length(temp)); for i=1:length(temp) for j=2:-1:1 sqre{j,i}=matchcell{2,abs(output(temp(i),j))}; end end</pre>	
备注	调用的数据被存储为 mat 格式，此处不便展示，见源程序的电子档

程序名	rowcluster.m	软件名	Matlab
<pre>%%%%%%%%%%%%%% 函数根据中文的手写特征识别 行 %%%%%%%%%%%%%%% %% 数据载入 load data1 hei=I; m=I{209}; [hhh,lll]=size(m); linshi=255*ones(1,lll); mudi=zeros(1,lll); for i=1:209 j=1; while j<hhh+1 if hei{i}(j,:)==linshi else hei{i}(j,:)=mudi; end j=j+1; end end end bound=zeros(hhh,209); for i=1:209 bound(:,i)=hei{i}(:,1); end boundary=bound'; clear i mudi linshi j hhh lll %% 图象匹配 %%%%%%%%%%%%%% 方 法 一 距 离 聚 类 %%%%%%%%%%%%%%% % s=char(1,209);</pre>			

```

% for i=1:209
%     s(i)=' ';
% end
% figure('Color',[1 1 1])
% y=pdist(boundary,'seuclidean');           %计算欧式距离
% z=linkage(y,'ward');                       %离差平方和方法聚类
% [H,T]=dendrogram(z,209,'labels',s);      %绘制动态聚类树
% xlabel('样本编号');
% ylabel('距离');
% title('动态聚类图');
% Clss=cluster(z,'maxclust',11);
% index=cell(1,11);
% for i=1:11
%     tm=find(Clss==i); %求第 i 类的对象
%     tm=reshape(tm,1,length(tm)); %变成行向量
%     index{i}=tm;
% end
% clear PSF H T i s y z Clss tm

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 方 法 二 fuzzy C-means 算
法 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5%%
C_data=boundary;
%options=[2,1000,1e-10,1]; %%%%%%%%%%%%%% 中文识别
options=[1.5,2000,1e-10,1]; %%%%%%%%%%%%%% 英文识别
[center,U,obj_fcn]=fcm(C_data,11,options);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% plot result and output result
%
maxU=max(U);
index=cell(1,11);
for i=1:11
    index{i}=find(U(i,:)==maxU);
end
%
for i=1:11
    fprintf('第%d 类的有%s\n',i,int2str(index{i}));
end
clear center U obj_fcn C_data maxU
%
```

备注	函数根据中文的手写特征识别行
----	----------------

程序名	rowsort.m	软件名	Matlab
%% 将 根 据 行 的 聚 类 结 果 拼 接 图 像 %%			

```

function [output,badsort]=rowsort(colnumvector,cutoff)
%% 数据导入
load data1;      %%中文图象数据
%load data2;     %%英文图象数据
len=length(I);
len1=length(colnumvector);
boundarycellmatrix=cell(2,len1);
%% 图像类型转化
for j=1:len
    I{j}=double(I{j});
    I{j}=255-I{j};
end
%% 选出根据行间距被聚为一类的图
Img=cell(1,len1);
for j=1:len1
    Img(j)=I(colnumvector(j));
end
%% 图像的边界左右各一条边界
for i=1:2
    for j=1:len1
        if i==1
            boundarycellmatrix{i,j}=Img{j}(:,1);
        end
        if i==2
            boundarycellmatrix{i,j}=Img{j}(:,72);
        end
    end
end
%% 创建数据矩阵
M=zeros(2*len1,length(Img{1}(:,1)));
for i=1:2
    if i==1
        for j=1:len1
            M(j,:)=(boundarycellmatrix{i,j})';
        end
    else
        for j=1:len1
            M(j+len1,:)=(boundarycellmatrix{i,j})';
        end
    end
end
clear i j;
%% 聚类分析
%figure('Color',[1 1 1]);

```

```

s=cell(1,2*len1);
for i=1:2*len1
    if i<=len1
        s{i}=num2str(-colnumvector(i));
    else
        s{i}=num2str(colnumvector(mod(i,len1+1)+1));
    end
end
y=pdist(M,'euclidean');           %计算欧式距离
z=linkage(y,'ward');               %离差平方和方法聚类
%[H,T]=dendrogram(z,len1,'labels',s); %绘制动态聚类树
%xlabel('样本编号');
%ylabel('距离');
%title('动态聚类图');
%Clss=cluster(z,'cutoff',0.7125);
Clss=cluster(z,'cutoff',cutoff);
sortnumber=cell(1,1);
tmm=cell(1,1);
for i=1:max(Clss)
    tm=find(Clss==i);    %求第 i 类的对象
    tmm{1}=tm;
    for j=1:length(tm)
        if tm(j)<=len1
            tmm{1}(j)=-colnumvector(tm(j));
        else
            tmm{1}(j)=colnumvector(mod(tm(j),len1+1)+1);
        end
    end
    sortnumber=[sortnumber,tmm];
end
clear H T i s y z Clss j tm tmm;
%% 输出
sortnumber=sortnumber(2:length(sortnumber));
badsort=[];
k=0;
for j=1:length(sortnumber)
    if length(sortnumber{j})==2
        k=k+1;
    end
end
output=zeros(k,2);
ii=1;
jj=1;
for j=1:length(sortnumber)

```


<pre> if length(sortnumber{j})==2 output(ii,:)=sortnumber{j}; ii=ii+1; else % fprintf('不能区别\n'); sortnumber{j}; badsort{jj}=sortnumber{j}; jj=jj+1; end end %Resultshow(output) </pre>	
备注	该程序将根据行的聚类结果拼接图像

程序名	pinjie.m	软件名	Matlab
<pre> shunxuu=cell(1,1); % xxxxx 是元胞数组，存放另外的连接关系 for i=1:11 xx=1;%xx 代表个数 A=shunxu{i}; x1=A(:,1); x2=A(:,2); while ~isempty(x1) %此处循环解决所有第一次未连接起来的图片 x0=[]; x0(1)=x1(1);x1(1)=[];%此处开始为上面的基本段 x0(2)=x2(1);x2(1)=[]; j=2; while findd(x1,x0(j))%该循环找右边图片 x0(j+1)=x2(x1==x0(j)); x2(x1==x0(j))=[];x1(x1==x0(j))=[]; j=j+1; end while findd(x2,x0(1))%该循环找左边图片 x0=[x1(x2==x0(1)),x0]; x1(x2==x0(2))=[];x2(x2==x0(2))=[];%注意!!! 此处 x0 已经被覆盖，上一句的 x0(1)为这一句的 x0(2)。这行错误找了一个小时!!! end shunxuu{i,xx}=x0; xx=xx+1; %xx 一直累加 end end </pre>			

<pre>clear A x1 x2 x end</pre>	
备注	该程序用于对相互匹配的图像的拼接

程序名	suanhangjianju.m	软件名	Matlab
<pre>load data1 hei=tuhei(I); wide=cell(1,209); for i=1:209 wide{i}=suanwide(hei{i}); end hangjianju=zeros(1,209); for i=1:209 hangjianju(i)=wide{i}(3); end</pre>			
备注	此程序用于计算文字行与行之间空白间隙的宽度（像素）		