

# 碎纸片的拼接复原

## 摘要

对破碎文件的拼接复原在现代社会中的作用越来越重要。其涉及领域十分广泛，涵盖司法物证复原，军事情报复原以及历史文物复原等。本文主要通过对碎纸片边缘对应像素点匹配后，用边值对比，聚类分析的方法判断出相邻的纸片。对于较复杂的情形进行计算机搜索以及部分人工干预，先确定部分特殊行和列，再用穷举搜索拼接余下碎纸块，最终达到拼接复原成原页面的目的。

### 针对问题一：

我们要对 19 条纵切出的碎纸片进行横向顺序的拼接。因为文字可视作点阵，因此将每条碎纸片以  $72 \times 1980$  的像素读入 matlab，每个格点的灰度值范围为 0-255。又易知相邻纸片左右边缘自上而下对应格点的灰度值应有极大的相似性。

扫描并记录每条碎纸片左右竖边格点灰度值矩阵，首先确定左侧竖边灰度值全为 255 的碎纸片为首张碎纸片，将其与余下碎纸片进行灰度值的 0/1 边值对比判断出与其相邻的纸片，接下来用同样的方法依次确定余下碎纸片横向拼接顺序。

### 针对问题二：

我们要对 209 块纵横相切出点碎纸块进行横向和竖向的拼接。由于碎片数量的增多，单纯使用 0/1 边值比较会导致误差偏大。运用其他文字特征来实现多重聚类分析达到顺序编排目的。

对于中文页面，研究印刷体文字特性，可得同一横排文字 baseline 是相应对齐的。记录碎纸片上下横边与所截字 baseline 之间的距离值，将值比较进行聚类分析，确定为同一横排的碎纸片；接着优化问题一中的相似度比较方法，构造离差和函数来确定同一横排的左右顺序。又由印刷体文字高度与行距值为固定值，将不同横排进行再一次的聚类分析确定竖向编排顺序。

对于英文页面，英文字母高度不一，因此要用异于中文页面的方法。而我们通过寻找到顶端及左边的边界碎片及搜寻碎片顺序的特殊化使得边值比较的精确度大大增加，有效地减少了人工干预的次数。使用计算机搜索和少量人工干预将特殊的首行和首列确定，然后从首行和首列的右上夹角开始运用离差和穷举搜索。

### 针对问题三：

我们要对正反两面共  $2 \times 209$  块的碎纸块进行拼接。由于英文字母四格三线的特点，无法像研究中文文字一样通过寻找特征行的方式拼接。我们采用类似于问题二附件四采用的穷举法，与贪心算法结合，边值对比找到匹配度最高的碎纸片并依照一定顺序进行排序。使得每张碎片都与相邻的正反两面一共四条边边值比对，同样大大增强了其准确性。

**关键字：**碎片复原，0/1 边值对比，像素点匹配，多重聚类分析，穷举搜索匹配

## 1. 问题重述

本文是基于在计算机技术迅速发展的今天，用自动拼接技术达到对破碎文件的拼接复原。破碎文件的拼接十分重要，其涉及的领域有历史文献修复，军事情报获取及司法物证复原等。传统的纯人工拼接虽然准确率较高但效率很低，而用合理算法实现的自动拼接技术则能在短时间内解决数量巨大的碎纸片拼接问题。

1. 题目给定来自同一页印刷文字文件的碎纸机破损纸片（仅纵切）。要求建立模型和算法对附件中的中、英文各一页文件的碎片数据进行拼接复原。若过程中需要人工干预，则写出干预方式及干预的时间节点。最后以图片形式及表格形式来表达复原结果。
2. 对于碎纸片既纵切又横切的情形，设计复原模型和算法，针对附件 3、4 里中、英文各一页文件的碎片数据进行拼接复原。同时若进行了人工干预，需写出干预方式及干预的时间节点。复原结果表达方式同上。
3. 从现实情形出发，解决双面打印文件的碎纸片拼接复原问题。设计相应的复原模型与算法，针对附件 5 的碎片数据给出拼接复原结果。复原结果表达方式同上。

## 2. 问题分析

### 2.1. 概论

这是一个聚类问题，根据碎纸片之间的联系，确定上下左右的拼接顺序实现页面复原。问题的特点：页面上中文印刷体形式统一；切出的碎纸片大小、形状相等且规则；纵横切出的碎纸片数量大，双面打印文件的碎纸片正反面不确定。难点在于：英文印刷体四线三格不便研究统一特征达到聚类目的，双面打印文件碎纸片正反面不易确定。

### 2.2. 问题一

问题一给出的是一单页页面仅纵切出的 19 条碎纸片，只需通过算法来确定碎纸片横向拼接的顺序。又因为碎纸片的形状规则且大小相等，因此不能够通过碎纸片边缘形状来确定，只能通过分析边缘文字特征进行拼接。易知首张纸片的左侧边缘为空白，末张纸片的右侧边缘为空白；两张相邻纸片的相邻边缘有很高的相似度，而文字可视为像素格点组成，因此就能通过碎纸片边缘对应格点的灰度值来进行相似度的比较，从而确定横向拼接顺序实现拼接复原。

### 2.3. 问题二

问题二给出的是一单页页面纵横相切出的  $11 \times 19$  的碎纸片。与第一问相比，碎纸片的数量大大增加，纸片竖直长度缩短；除了需要碎纸片横向顺序编排外还需确定竖向的拼接顺序；中、英文字体在横向印刷的特征上出现差异，需分开讨论。

对于中文页面，进行横向编排时，在问题一的基础上考虑到纸片竖直长度缩短，能配对的格点数减少会导致误差的加大，因此还需用其他文字的特征来进行聚类分析来确定横排顺序。进行竖向编排时，考虑中文印刷体文字高度与行距为固定值，通过所截文字顶部线和底部线与上下横边距离值的三重聚类分析来完成竖向的编排。

对于英文页面，英文字母的特征并不统一，在对碎纸片边缘格点灰度值的边值对比时误差较大。因此我们需要采取人工干预，先编排出较易确定的自上第一横排及自左第一竖排的碎纸片顺序。这样一来，在接下来对余下碎纸片进行格点灰度值边值对比时，能够与两条确定边双重的边值对比排除，使得能更快更准确的找到余下碎纸片的拼接顺序。

## 2.4. 问题三

我们要对正反两面共  $2 \times 209$  块的碎纸块进行拼接。在问题二英文算法的基础上，增加了正反面的干扰，但是只不过是大大的增加了计算量。同时，也是由于问题三有正反两面，所以恰巧能将另一面的两四条边来验证参与边值匹配的是否正确的对比。实际上这样的方法可以省去问题二中大量的人工干预并使结果更加精准。问题的难点在于确定碎纸片的哪些面在原页面的同一面。我们采用类似于问题二附件四采用的穷举法，与贪心算法结合，边值对比找到匹配度最高的碎纸片并依照一定顺序进行排序。使得每张碎片都与相邻的正反两面一共四条边边值比对，大大增强了其准确性。

## 3. 模型假设

- 1) 所有碎片上的文字字体统一、字号大小统一；
- 2) 被切割的碎片形状规则，大小一致；
- 3) 原文章行距统一，且与页首、页尾空白间距均不相同；
- 4) 复原后的文章是有意义的，即能依据我们有的知识经验进行人工干预。

## 4. 符号说明

$A_i$	问题一中第 $i$ 条纸片左侧竖边格点灰度值矩阵
$B_i$	问题一中第 $i$ 条纸片右侧竖边格点灰度值矩阵
$A_x$	问题一中求出的首条碎片左侧竖边格点灰度值矩阵
$A_y$	问题一中求出的首条碎片左侧竖边格点灰度值矩阵
$B_x$	问题一中求出的末条碎片右侧格竖边点灰度值矩阵
$B_y$	问题一中求出的末条碎片右侧格竖边点灰度值矩阵
$A_i(j)$	问题一中第 $i$ 条碎片左侧竖边格点灰度值矩阵的第 $j$ 个数据
$B_i(j)$	问题一中第 $i$ 条碎片右侧竖边格点灰度值矩阵的第 $j$ 个数据
$M$	全为 255 的 $1 \times 180$ 的矩阵
$N$	全为 255 的 $180 \times 1$ 的矩阵
$E_i$	问题二中四种分类各占的碎片个数 ( $i=1,2,3,4$ )
$L_i$	问题二中第 $i$ 块碎纸片左横边格点灰度值矩阵
$U_i$	问题二中第 $i$ 块碎纸片上横边格点灰度值矩阵
$D_i$	问题二中第 $i$ 块碎纸片下横边格点灰度值矩阵
$P$	碎片从上往下汉字截止的 <b>baseline</b> 到顶端的距离
$Q$	碎片从下往上汉字截止的 <b>baseline</b> 到底端的距离

## 5. 模型的建立与求解

### 5.1. 问题一模型的建立与求解

将碎纸片的拼接问题落实到纸片的左右竖边。首先确定出首张纸片，再对纸片左右两竖边格点灰度值进行边值对比模型，依次确定相邻纸片，达到复原的目标。

#### 5.1.1. 问题分析

对于问题 1，现在要复原碎纸机纵切出的 19 条碎纸片。因为只存在纵向的切割，因此在拼接时只要考虑 19 列纸片左右横向顺序。考虑碎纸片上文字都为点阵（如图 1），则前一张纸片的右竖边与后一张纸片的左竖边在相同的格点处应存在相近的灰度值。并且题目一中所给中、英文纸片都存在这个特点。

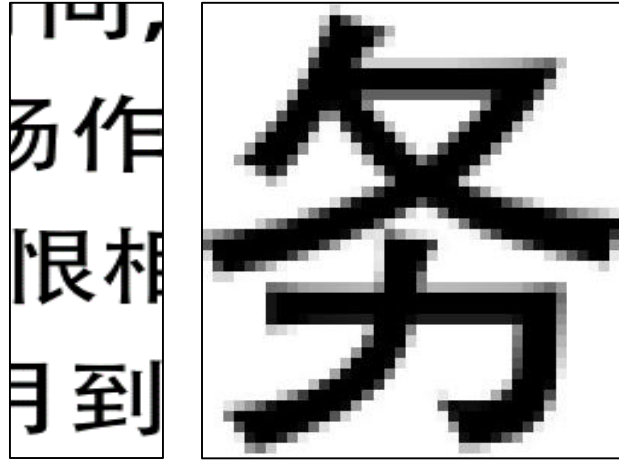


图 1 碎纸片格点灰度值示意图

### 5.1.2. 模型的建立

我们只需具体分析碎纸片的左右竖边。19 条纸片总共有 38 条边。每条竖边上有 1980 个格点。每个格点的灰度值范围为 0-255（由黑到白）。

具体算法步骤如下：

- 1) 将每条碎纸片以 72\*1980 的像素读入 MATLAB, 依次取出第  $i$  条碎片的最左和最右的两列灰度值放入矩阵  $A_i$ 、 $B_i$  ( $i=1, 2, \dots, 19$ )。

$$A_i = \begin{pmatrix} a_{1i} \\ a_{2i} \\ \vdots \\ a_{ji} \\ \vdots \\ a_{1979i} \\ a_{1980i} \end{pmatrix} \quad B_i = \begin{pmatrix} b_{1i} \\ b_{2i} \\ \vdots \\ b_{ji} \\ \vdots \\ b_{1979i} \\ b_{1980i} \end{pmatrix}$$

- 2) 用计算机搜索出两个矩阵  $A_x = B_y = \begin{pmatrix} 255 \\ 255 \\ \vdots \\ 255 \\ 255 \end{pmatrix}$ , 即第  $x$  条碎片的左列与第  $y$

条碎片的右列为空白。则可以判断出第  $x$  条为首条纸片, 第  $y$  条为末尾纸片。

- 3) 从第 2 张开始遍历所有其余的碎纸片, 对于第  $i$  张碎纸片, 若其左边沿  $A_i$  与  $i-1$  张右边沿  $B_{i-1}$  “最接近”, 拼合此两张。

**判断“最接近”两张的边值比较模型:**

采取 0/1 比较法, 即对于对应的格点, 若灰度值不同, 该点计 1, 否则计 0, 最后计算总和, 显然  $F$  越小, 两条碎片的边缘像素点灰度值相似度越高。

表达式形式为:

$$F_{i,i-1} = \sum_{j=1}^{1980} (A_i(j) - B_{i-1}(j)) \wedge 1$$

将两两间的  $F$  算出来排序,  $F_{min} = \inf(F)$  对应的  $i$  值即为所谓的“最接近”的碎纸片。

4) 若  $i$  为最后一张碎纸片, 退出, 完成复原, 否则重复 (3)。

### 5.1.3. 模型的求解

按照 0/1 比较法我们对模型进行求解, 较为顺利地拼接出复原图(见附件一、二)及复原顺序。复原顺序如下图所示:

表一: 附件一(中文)复原顺序

位置	1	2	3	4	5	6	7	8	9	10
编号	008	014	012	015	003	010	002	016	001	004
位置	11	12	13	14	15	16	17	18	19	
编号	005	009	013	018	011	007	017	000	006	

表二: 附件二(英文)复原顺序

位置	1	2	3	4	5	6	7	8	9	10
编号	003	006	002	007	015	018	011	000	005	001
位置	11	12	13	14	15	16	17	18	19	
编号	009	013	010	008	012	014	017	016	004	

### 5.1.4. 模型的精确度分析

为了判断此边值对比模型得出结果的精确度, 我们定义  $E$  为精确度函数:

$$E = \frac{1}{19} \sum \left| \frac{F' - F_{min}}{F' + F_{min}} \right|$$

其中  $F'$  为倒数第二小的  $F$  值且  $0 \leq E \leq 1$ 。易知  $F'$  越大  $E$  越大,  $F$  越精确, 表示如此拼接的可信度越大。分别算出中、英文的精确度函数取值为: 0.1684 和 0.0343。

精确度函数英文明显比中文低的原因是英文字母较为单一, 不及中文多变, 笔画数也较中文少, 这就造成了英文的精确度函数降低, 误差增大。

## 5.2. 问题二模型的建立与求解

问题二与问题一相比多了一个横向切割的干扰, 并且由于中英文字型特点的差异, 采取的处理方式也应不同。

对于中文, 我们考虑先将所有碎片按照上有切字下有切字、上有切字下无切

字、上无切字下有切字、上无切字下无切字分成四类，根据这四类碎片的有无切字的距离特征将其更加细化的分类，能将所有碎片所在行数确定，即将所有碎片分成 11 行，每行 19 块碎片，利用问题 1 中的边值对比辅以人工干预确定各行顺序及每行的排列。

对于英文，我们依据上方和右侧碎片的特征，利用计算机搜索加上小部分人工干预将纸张的首行和首列确定，然后用剩下的碎片的右侧和上方的格点灰度值特征进行相似度比较，用穷举搜索的方法将剩下的碎片依次从左到右、从上到下补上。

### 5.2.1. 附件三（中文）的模型建立与求解

#### 问题分析：

对于问题二，得到的是由纵横切出的  $11 \times 19$  个碎片。由于碎片个数较多，我们需要利用其特性。并且按先横向编排再竖向编排的方式进行复原。

#### 模型的建立：

基本算法分为 5 步，步骤为：

##### 1) 将碎片按照切字特征分类，共聚四类，第一次聚类分析。

按照上有切字下有切字、上有切字下无切字、上无切字下有切字、上无切字下无切字分成四类，其个数如下表所示：

将每块碎纸片以  $72 \times 180$  的像素读入 matlab。

定义矩阵  $M = (255 \ 255 \ \dots \ 255 \ 255)$ ，变量  $E_1, E_2, E_3, E_4$ ，其初始值都为 0。 $U_i, D_i$  为第  $i$  块碎纸片上横边与上横边格点灰度值组成的矩阵。

$$\begin{cases} E_1 + 1 & U_i \neq M \text{ 且 } D_i \neq M \\ E_2 + 1 & U_i \neq M \text{ 且 } D_i = M \\ E_3 + 1 & U_i = M \text{ 且 } D_i \neq M \\ E_4 + 1 & U_i = M \text{ 且 } D_i = M \end{cases}$$

则上有下有、上有下无、上无下有、上无下无的碎片个数分别为  $E_1, E_2, E_3, E_4$ 。

##### 2) 按照第 1 步的聚类分别统计不同 P、Q 值出现频数规律，将所有碎片分成 11 类，每一类里的 19 块碎片在同一行，进行第二次聚类分析。

为了更细致的分类，我们引入 P, Q,  $T_0$ ,

我们定义：P 为碎片从上往下汉字截止的 baseline 到顶端的距离；（如图 2 左所示）Q 为碎片从下往上汉字截止的 baseline 到顶端的距离；（如图 2 右所示） $T_0$  为已确定的汉字文字高度的常值。

并且有， $0 \leq P \leq T_0, 0 \leq Q \leq 0$ 。

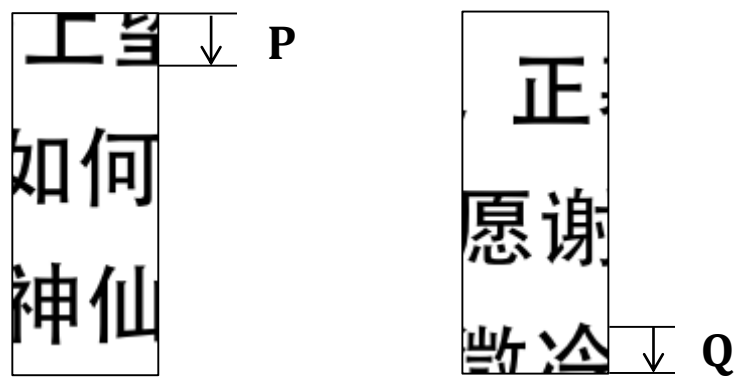


图 2 P、Q 值的确定

我们先从上端有字的 1、2 组入手，我们统计出不同的P值各自出现的频数。分别得到如下频数图：

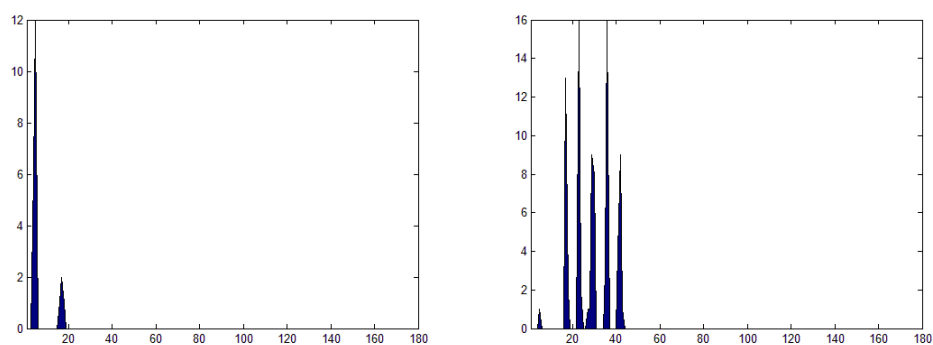


图 3 P-频数图

其中横坐标是P值，纵坐标为频数。

我们再分析下端有字的情况，分别从 1、3 组入手，同理统计出不同的Q值出现的频数，频数图如下：

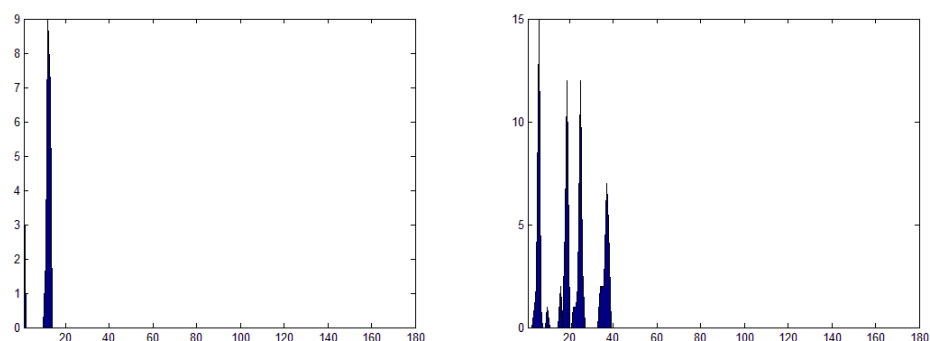


图 4 Q-频数图

其中横坐标为 Q 值，纵坐标为统计的不同 Q 值频数。

用 MATLAB 自动寻峰 findpeaks 函数搜寻整数峰值，然后能得到整数峰值点



（即正是 P、Q 值）和每种峰值出现的频数统计表。因此，我们能依据统计表将所有碎片分成 11 排。

**3) 将第二次聚类结果采用问题（一）中的求解方式优化后求解，少许人工干预即可得出每行的拼合情况。**

实际操作时发现，问题一中的函数误差较大，精确度函数较低。碎片之间会出现灰度值相似度的区分度不明显，容易造成排列错误，因此将函数进一步修正为离差和，函数如下：

$$F_{i,i-1} = \sum_{j=1}^{1980} |A_i(j) - B_{i-1}(j)|$$

进一步计算离差平方和：

$$F_{i,i-1} = \sum_{j=1}^{1980} (A_i(j) - B_{i-1}(j))^2$$

此外为全面比较模型优劣，重新设计改良相关系数，函数如下所示：

$$F_{i,i-1} = 1 - \frac{\sum(A_i(j) - \overline{A_i(j)})(B_{i-1}(j) - \overline{B_{i-1}(j)})}{\sqrt{\sum(A_i(j) - \overline{A_i(j)})^2 \cdot \sum(B_{i-1}(j) - \overline{B_{i-1}(j)})^2}}$$

改良相关系数  $F_{i,i-1}$  越小，相关性越大。

**4) 根据字体高度为常数，即截断汉字的两端碎片特征高度 P、Q 之和应为常数，可推断出每组之间的上下关系，第三次聚类分析。**

易知当一个字横穿隔断线的时候，会同时在上端与下端留下线索，故上端有字与上端无字的个数大致相等，且对应相连的两行碎片的峰值之和应为  $T_0$

即存在

$$P + Q = T_0$$

则可以确定通过比对，{①}={⑩}、{②}={⑨}即此两个编号内的碎片完全相同。

**5) 依次连接每行文字，利用像素统计法可得最后的拼合图像。**

由第四步的固定字高和行间距可以得到三组段落组块，不能再之前的离差和函数排列顺序。但是我们知道印刷体的文章行距都是固定的，而且与页首和页尾的空白间距不同。

所以我们编写程序，从上往下扫描并依次将每行像素的灰度值求和得到  $S(h)$ ，于是得到每行的像素总和  $S(h)$  随着  $h$ （扫描的格点位置与页首的距离）增加而变化。

**模型的求解：**

## 1) 第一次聚类分析

由第一步的函数可以归类得到四组碎片分类具体情况：

表三：聚类分类

分类	上有下有	上有下无	上无下有	上无下无
数目	25	76	86	22

## 2) 第二次聚类分析

第二步中，通过 MATLAB 峰值读取函数得到峰值的具体信息，并整理得到有切字的碎片分布表：

表四：切字碎片分布表

上端有字						下端有字					
上有下有			上有下无			上有下有			上无下有		
组号	P	频数	组号	P	频数	组号	Q	频数	组号	Q	频数
①	5	18	③	5	1	⑨	1	4	⑪	6	19
②	17	4	④	17	15	⑩	12	18	⑫	10	1
			⑤	23	18				⑬	18	19
			⑥	29	19				⑭	24	19
			⑦	36	19				⑮	36	18
			⑧	42	14						

根据每行近似等于 19 个元素，我们将有字部分进行二次聚类分析，将上表分成的 15 组确定为 11 排：

首先可以确定频数为 19 的组号——⑥、⑦、⑪、⑬、⑭各自分为一横排；

频数为 18 的组号——①、⑤、⑩、⑮ 进行人工干预补充碎片后也为自为一横排；

剩下的频数为 15 和 4 的②、④为一横排；

最后频数为 14 的⑧与剩下的上下没有切字的碎片凑成一组。

至此，我们得到了 11 组碎片分类，每组碎片都是 19 块碎片。将各行按照问题一的方式求解横向拼接方式。

### 3) 运用优化后的离差和函数将分好的 11 排碎片进行内部的左右排序

优化后的函数精确度显著提高，但是因为碎片变短，数据量变小，在每行内部排序时，可能出现错排情况，如下图所示：

便邮。温香熟美。醉慢云鬟垂两耳。多谢春工。红是玉红。一颗 不是花  
樱桃樊素口。不爱黄金，只爱人长久。学画鸦儿 眉尖已作伤春犹未就。皱。

图 5 错排情况

很明显这一排的右端排错了，需要人工干预修改顺序方才正确：

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。

图 6 人工干预后正确情况

计算所有函数模型在拼接结果正确前提下的精确度函数，得到下表

表五：精确度函数

		0/1 比较	离差和	离差平方和	相关系数
精确度 函数	中文	0.1684	0.4819	0.6088	0.0122
	英文	0.0343	0.1164	0.1983	无法拼出

从表中发现离差平方和精确度远高于其它方法，故我们在以后的计算比对中优先考虑采用离差平方和模型。

### 4) 第三次聚类分析

从第二步得到的表格可以看出 ⑦ 与 ⑪ 中  $36+6=42$ ，⑥ 与 ⑩ 中  $29+12=41$ ，⑤ 与 ⑬ 中  $23+18=41$ ，④ 与 ⑭ 中  $17+24=41$ ，③ 与 ⑮ 中  $5+36=41$ 。所以不难推断字的宽度约为 41 个像素点。并由此得出第二次聚类出的小组上下关系，即第三次聚类分析（前组在后组上方）：

第一类：⑪；⑦

第二类：⑩；⑥

第三类：⑬；⑤

第四类：⑭；④（②/⑨、④）

第五类：⑮；③（①/⑩、③）

如此以来，可以确定个别两行之间的前后顺序。

随后从上端无字的后两组入手，分别计算其空行从上往下延伸的像素点个数分布。利用类似的方式重新聚类分析，得出行间距约为 39 个像素点，同样通过聚类分析，确定了剩余五行的分组及各组之间的关系。

5) 像素统计法拼合图像。

依次连接每行文字，利用像素统计法可得最后的拼合图像。通过求解模型，我们得出了三组段落组块，不易分出先后顺序：

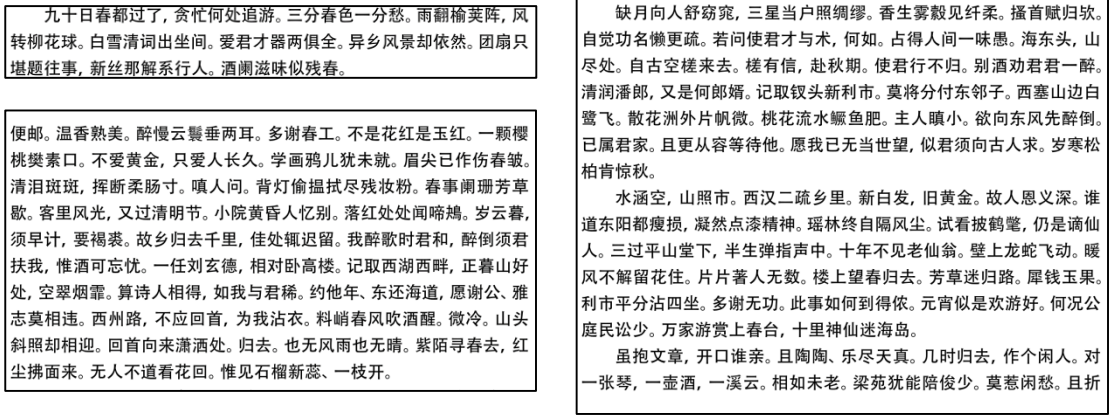


图 7 段落分块

由此我们能够拼出两种图像：



图 8 两种拼合结果

按列分别统计各行像素分布：



定义矩阵  $M = (255 \ 255 \ \dots \ 255 \ 255)$ ，矩阵  $N = \begin{pmatrix} 255 \\ 255 \\ \vdots \\ 255 \\ 255 \end{pmatrix}$ 。

$$\begin{cases} L_i = N & \text{第 } i \text{ 块碎纸片位于原图左竖边第一列} \\ U_i = M & \text{第 } i \text{ 块碎纸片位于原图上顶部第一行} \end{cases}$$

- 2) 用问题（一）中的方法，辅以少量人工干预，拼合上边界与左边界。
- 3) 从第二行第二列开始，从顶端与左端同时分别采用边值对比的方式，综合两种比较结果，确定拼接的碎片。期间需要逐步执行程序与人工干预相结合。

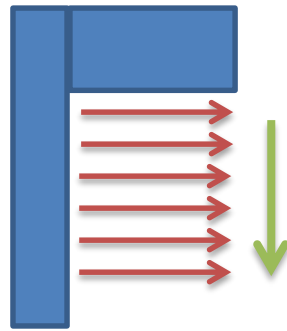


图 12 穷举顺序图示

我们记  $R_{i,j}$ 、 $L_{i,j}$ 、 $U_{i,j}$ 、 $D_{i,j}$  分别为位于第  $i$  行第  $j$  列的碎片右侧、左侧、上侧、下侧边缘灰度值矩阵。 $R(2, 1)$  和  $D(1, 2)$  已知。由于用单侧边缘灰度值的离差和来匹配误差比较大，所以考虑用上侧和左侧的边缘灰度值离差和的共同匹配，定义新离差和函数如下：

$$H(i, j) = \sum_{k=1}^{1980} (R_{2,1}(j) - L_{i,j}(k))^2 + \sum_{k=1}^{1980} (D_{1,2}(j) - U_{i,j}(k))^2$$

确定  $\inf[H(i, j)]$  所对应的第  $i$  行第  $j$  列的碎片应放在第 2 行第 2 列。如上图所示，依次从左往右，从上往下逐步确定碎片顺序，注意每次都更新一遍  $H(i, j)$ 。

- 4) 直到 11 行 19 列，即完成对碎片的拼接。得出拼接顺序：

表六：附件四拼接顺序

排	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	191	75	11	154	190	184	2	104	180	64	106	4	149	32	204	65	39	67	147
2	201	148	170	196	198	94	113	164	78	103	91	80	101	26	100	6	17	28	146
3	86	51	107	29	40	158	186	98	24	117	150	5	59	58	92	30	37	46	127

4	19	194	93	141	88	121	126	105	155	114	176	182	151	22	57	202	71	165	82
5	159	139	1	129	63	138	153	53	38	123	120	175	85	50	160	187	97	203	31
6	20	41	108	116	136	73	36	207	135	15	76	43	199	45	173	79	161	179	143
7	208	21	7	49	61	119	33	142	168	62	169	54	192	133	118	189	162	197	112
8	70	84	60	14	68	174	137	195	8	47	172	156	96	23	99	122	90	185	109
9	132	181	95	69	167	163	166	188	111	144	206	3	130	34	13	110	25	27	178
10	171	42	66	205	10	157	74	145	83	134	55	18	56	35	16	9	183	152	44
11	81	77	128	200	131	52	125	140	193	87	89	48	72	12	177	124	0	102	115

5.3. 问题三模型的建立与求解

5.3.1. 问题分析

对于问题三，我们采取类似于解决问题二英文的算法。与问题二中的英文不同的是，问题二只能考虑两条边值的对比问题，虽然已经尽最大可能减小了误差，但是还是难免人工干预，而问题三可以使得与四条边(正反两面)参与边值对比，虽然计算量增加了一倍，但是结果的可靠性也大大地增强了。

人工干预剔除了一些误差碎片。找出后采用边值对比的方式对顶端与右端碎片进行拼接，结果如下图所示：

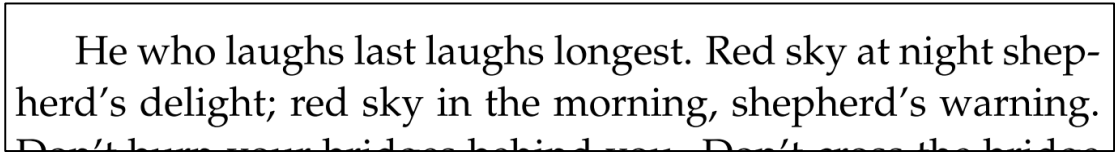


图 13 顶端横排结果

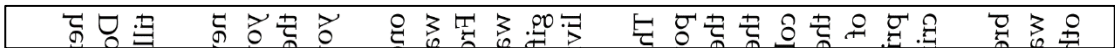


图 14 左端纵排（经转置后）结果

5.3.2. 模型的建立与求解

基本算法也分为 4 步，具体步骤如下：

- 1) 采用搜寻上边值白条与右边值白条的方法找到并分类原图顶端与左端一共  $11+19-1=29$  个碎纸片。
- 2) 用问题（一）中的方法，采取双边值比较，辅以少量人工干预，拼合出上边界与左边界。
- 3) 从第二行第二列开始，从顶端与左端同时分别均采用双边值对比的方式，综合两种比较结果，确定拼接的碎片。期间需要逐步执行程序与人工干预相结合。



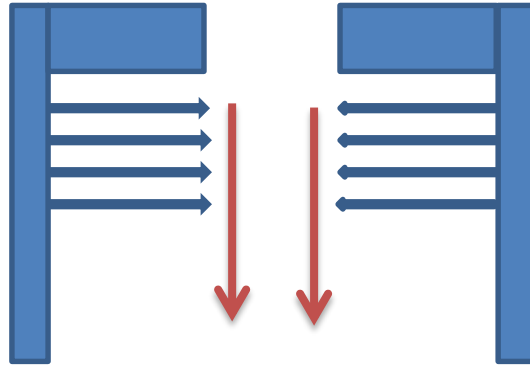


图 15 穷举顺序图示

重新定义新离差和函数如下：

$$H'(i,j) = \sum_{k=1}^{1980} (L'_{2,1}(j) - R'_{i,j}(k))^2 + \sum_{k=1}^{1980} (D'_{1,2}(j) - U'_{i,j}(k))^2$$

4) 直到 11 行 19 列，即完成对碎片的拼接。得出拼接顺序：

表七：附件五拼接顺序

排	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	136a	047b	020b	164a	081a	189a	029b	018a	108b	066b	110b	174a	183a	150b	155b	140b	125b	111a	078a
2	005b	152b	147b	060a	059b	014b	079b	144b	120a	022b	124a	192b	025a	044b	178b	076a	036b	010a	089b
3	143a	200a	086a	187a	131a	056a	138b	045b	137a	061a	094a	098b	121b	038b	030b	042a	084a	153b	186a
4	083b	039a	097b	175b	072a	093b	132a	087b	198a	181a	034b	156b	206a	173a	194a	169a	161b	011a	199a
5	090b	203a	162a	002b	139a	070a	041b	170a	151a	001a	166a	115a	065a	191b	037a	180b	149a	107b	088a
6	013b	024b	057b	142b	208b	064a	102a	017a	012b	028a	154a	197b	158b	058b	207b	116a	179a	184a	114b
7	035b	159b	073a	193a	163b	130b	021a	202b	053a	177a	016a	019a	092a	190a	050b	201b	031b	171a	146b
8	172b	122b	182a	040b	127b	188b	068a	008a	117a	167b	075a	063a	067b	046b	168b	157b	128b	195b	165a
9	105b	204a	141b	135a	027b	080a	000a	185b	176b	126a	074a	032b	069b	004b	077b	148a	085a	007a	003a
10	009a	145b	082a	205b	015a	101b	118a	129a	062b	052b	071a	033a	119b	160a	095b	051a	048b	133b	023a
11	054a	196a	112b	103b	055a	100a	106a	091b	049a	026a	113b	134b	104b	006b	123b	109b	096a	043b	099b

排	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	078b	111b	125a	140a	155a	150a	183b	174b	110a	066a	108a	018b	029a	189b	081b	164b	020a	047a	136b
2	089a	010b	036a	076b	178a	044a	025b	192a	124b	022a	120b	144a	079a	014a	059a	060b	147a	152a	005a
3	186b	153a	084b	042b	030a	038a	121a	098a	094b	061b	137b	045a	138a	056b	131b	187b	086b	200b	143b
4	199b	011b	161a	169b	194b	173b	206b	156a	034a	181b	198b	087a	132b	093a	072b	175a	097a	039b	083a
5	088b	107a	149b	180a	037b	191a	065b	115b	166b	001b	151b	170b	041a	070b	139b	002a	162b	203b	090a
6	114a	184b	179b	116b	207a	058a	158a	197a	154b	028b	012a	017b	102b	064b	208a	142a	057a	024a	013a
7	146a	171b	031a	201a	050a	190b	092b	019b	016b	177b	053b	202a	021b	130a	163a	193b	073b	159a	035a
8	165b	195a	128a	157a	168a	046a	067a	063b	075b	167a	117b	008b	068b	188a	127a	040a	182b	122a	172a



9	003b	007b	085b	148b	077a	004a	069a	032a	074b	126b	176a	185a	000b	080b	027a	135b	141a	204b	105a
10	023b	133a	048a	051b	095a	160b	119a	033b	071b	052a	062a	129b	118b	101a	015b	205a	082b	145a	009b
11	099a	043a	096b	109a	123a	006a	104a	134a	113a	026b	049b	091a	106b	100b	055b	103a	112a	196b	054b

## 6. 模型评价与分析

### 6.1. 模型的优点

- 1) 在问题二对中文碎纸片的横向拼接时，对边值对比的模型进行了改进，并且对碎纸片上下横边与文字 **baseline** 的距离值进行初步聚类分析确定横向编排顺序，大大增加了结果的准确性。
- 2) 在得到 11 行编排好的横排纸片后，对碎纸片横边与文字 **baseline** 的距离值进行配对，再次聚类分析，确定竖向拼接顺序。之后通过人工干预排除错误的段落拼接，得到准确的复原拼接结果。
- 3) 采用多种函数模型交叉验证，并定义精确性函数 **E** 进行比较。
- 4) 第三问通过巧妙地搜索顺序，使得边值利用率最大化

### 6.2. 模型的缺点

在问题二英文碎纸片拼接时，为使结果准确性加强进行的人工干预较多，使操作变得复杂。

## 7. 模型的改进

### 基于小波分析的字母特征搜索法

已有模型的分析：在对英文的碎片进行依照行距进行分类时会遇到 **bdfhkl**、**gpq** 以及 **j** 这 12 个字母的干扰，具体如下图所示：

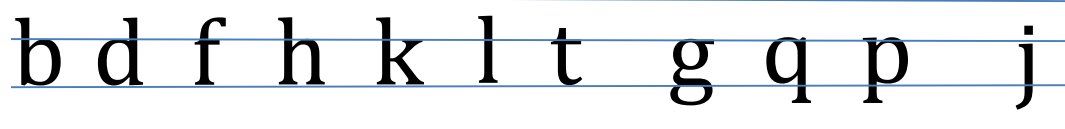


图 16 干扰字母

这三类字母在小碎片的边缘出现会干扰程序对行距、字长的判断，而造成计算的误差，所以得加入很多人工干预。如何减少人工干预的次数是模型优化的关键。

我们知道英文字母一共只有 26 个，大小写一共 52 种字形。从碎片中观察发现，至少能单独得到 26 个小写英文字母的完整网格图，如果能找到合适的统计函数得到 26 个各具特征的波形图，并且这些波形图能唯一对应一个字形，我们

就能十分准确得到每张碎片的边缘特征。最后依照这些碎片携带的特征准确匹配得到碎片的排列顺序。

下面以字母a和一个简单的统计函数 $H(x)$ 为例说明上述方法：

从碎片中截取出一个  $22*24$  的网格格点图，如图左侧 a 所示，依次将网格图中的灰度值存入矩阵 $W_k(k = 1, 2 \dots 26)$ 中，然后从左向右搜索，同时用函数 $H(x)$ 得到一个特征波形图，定义函数如下：

$$H(x) = \sum_i W_k(i, x)$$

其中 $W_k(i, x)$ 表示矩阵 $W_k$ 的 i 行 x 列的灰度值。利用该函数统计得到的散点形成波形图来作为字母 a 的特征波形。

当某一碎片刚好切到了字母 a，如图右侧 a 所示，则可以同样统计得到波形图的一部分，然后与 26 种波形图搜索匹配确定碎片的该位置上应接字母 a，从而确定碎片部分特征。



图 17 部分特征

## 8. 参考文献

- [1] Mark M. Meerschaer 著 刘来福 杨淳 黄海洋译 数学建模方法与分析
- [2] 马莉著 MATLAB 数学实验与建模 清华大学出版社
- [3] 唐远炎 王玲 小波分析与文本文字识别 科学出版社

9. 附录

9.1. 附件一和附件二的复原结果图及表格

城上层楼叠嶂。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。簌簌衣巾落枣花。村北村南响犂车。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。别离阻。谁念萦损襄王，何曾梦云雨。旧恨前欢，心事两无凭。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。临水纵横回晚靉。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。

组豆庚桑真过矣，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，飐飐，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。汤发云腴酽白，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。

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-

附件一

位置	1	2	3	4	5	6	7	8	9	10
编号	008	014	012	015	003	010	002	016	001	004
位置	11	12	13	14	15	16	17	18	19	
编号	005	009	013	018	011	007	017	000	006	

附件二

位置	1	2	3	4	5	6	7	8	9	10
编号	003	006	002	007	015	018	011	000	005	001
位置	11	12	13	14	15	16	17	18	19	
编号	009	013	010	008	012	014	017	016	004	

9.2. 附件三结果（中文）

便郎。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。落红处处闻啼鸟。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。回首向来萧瑟处。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。桃花流水鳜鱼肥。主人嗔小。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。此事如何到得依。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

附件三

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

### 9.3. 附件四结果（英文）

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 is 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

附件四

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

## 9.4. 附件五结果（正反）

He who laughs last laughs longest. Red sky at night shepherd's delight; red sky in the morning, shepherd's warning. Don't burn your bridges behind you. Don't cross the bridge till you come to it. Hindsight is always twenty-twenty.

Never go to bed on an argument. The course of true love never did run smooth. When the oak is before the ash, then you will only get a splash; when the ash is before the oak, then you may expect a soak. What you lose on the swings you gain on the roundabouts.

Love thy neighbour as thyself. Worrying never did anyone any good. There's nowt so queer as folk. Don't try to walk before you can crawl. Tell the truth and shame the Devil. From the sublime to the ridiculous is only one step. Don't wash your dirty linen in public. Beware of Greeks bearing gifts. Horses for courses. Saturday's child works hard for its living.

Life begins at forty. An apple a day keeps the doctor away. Thursday's child has far to go. Take care of the pence and the pounds will take care of themselves. The husband is always the last to know. It's all grist to the mill. Let the dead bury the dead. Count your blessings. Revenge is a dish best served cold. All's for the best in the best of all possible worlds. It's the empty can that makes the most noise. Never tell tales out of school. Little pitchers have big ears. Love is blind. The price of liberty is eternal vigilance. Let the punishment fit the crime.

The more things change, the more they stay the same. The bread always falls buttered side down. Blood is thicker than water. He who fights and runs away, may live to fight another day. Eat, drink and be merry, for tomorrow we die.

What can't be cured must be endured. Bad money drives out good. Hard cases make bad law. Talk is cheap. See a pin and pick it up, all the day you'll have good luck; see a pin and let it lie, bad luck you'll have all day. If you pay peanuts, you get monkeys. If you can't be good, be careful. Share and share alike. All's well that ends well. Better late than never. Fish always stink from the head down. A new broom sweeps clean. April showers bring forth May flowers. It never rains but it pours. Never let the sun go down on your anger.

Pearls of wisdom. The proof of the pudding is in the eating. Parsley seed goes nine times to the Devil. Judge not, that ye be not judged. The longest journey starts with a single step. Big fish eat little fish. Great minds think alike. The end justifies the means. Cowards may die many times before their death. You can't win them all. Do as I say, not as I do. Don't upset the apple-cart. Behind every great man there's a great woman. Pride goes before a fall.

You can lead a horse to water, but you can't make it drink. Two heads are better than one. March winds and April showers bring forth May flowers. A swarm in May is worth a load of hay; a swarm in June is worth a silver spoon; but a swarm in July is not worth a fly. Might is right. Let bygones be bygones. It takes all sorts to make a world. A change is as good as a rest. Into every life a little rain must fall. A chain is only as strong as its weakest link.

Don't look a gift horse in the mouth. Old soldiers never die, they just fade away. Seeing is believing. The opera ain't over till the fat lady sings. Silence is golden. Variety is the spice of life. Tomorrow never comes. If it ain't broke, don't fix it. Look before you leap. The road to hell is paved with good

### 附件五

排	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	136a	047b	020b	164a	081a	189a	029b	018a	108b	066b	110b	174a	183a	150b	155b	140b	125b	111a	078a
2	005b	152b	147b	060a	059b	014b	079b	144b	120a	022b	124a	192b	025a	044b	178b	076a	036b	010a	089b
3	143a	200a	086a	187a	131a	056a	138b	045b	137a	061a	094a	098b	121b	038b	030b	042a	084a	153b	186a
4	083b	039a	097b	175b	072a	093b	132a	087b	198a	181a	034b	156b	206a	173a	194a	169a	161b	011a	199a
5	090b	203a	162a	002b	139a	070a	041b	170a	151a	001a	166a	115a	065a	191b	037a	180b	149a	107b	088a
6	013b	024b	057b	142b	208b	064a	102a	017a	012b	028a	154a	197b	158b	058b	207b	116a	179a	184a	114b
7	035b	159b	073a	193a	163b	130b	021a	202b	053a	177a	016a	019a	092a	190a	050b	201b	031b	171a	146b
8	172b	122b	182a	040b	127b	188b	068a	008a	117a	167b	075a	063a	067b	046b	168b	157b	128b	195b	165a
9	105b	204a	141b	135a	027b	080a	000a	185b	176b	126a	074a	032b	069b	004b	077b	148a	085a	007a	003a
10	009a	145b	082a	205b	015a	101b	118a	129a	062b	052b	071a	033a	119b	160a	095b	051a	048b	133b	023a
11	054a	196a	112b	103b	055a	100a	106a	091b	049a	026a	113b	134b	104b	006b	123b	109b	096a	043b	099b

排	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	078b	111b	125a	140a	155a	150a	183b	174b	110a	066a	108a	018b	029a	189b	081b	164b	020a	047a	136b
2	089a	010b	036a	076b	178a	044a	025b	192a	124b	022a	120b	144a	079a	014a	059a	060b	147a	152a	005a
3	186b	153a	084b	042b	030a	038a	121a	098a	094b	061b	137b	045a	138a	056b	131b	187b	086b	200b	143b
4	199b	011b	161a	169b	194b	173b	206b	156a	034a	181b	198b	087a	132b	093a	072b	175a	097a	039b	083a

5	088b	107a	149b	180a	037b	191a	065b	115b	166b	001b	151b	170b	041a	070b	139b	002a	162b	203b	090a
6	114a	184b	179b	116b	207a	058a	158a	197a	154b	028b	012a	017b	102b	064b	208a	142a	057a	024a	013a
7	146a	171b	031a	201a	050a	190b	092b	019b	016b	177b	053b	202a	021b	130a	163a	193b	073b	159a	035a
8	165b	195a	128a	157a	168a	046a	067a	063b	075b	167a	117b	008b	068b	188a	127a	040a	182b	122a	172a
9	003b	007b	085b	148b	077a	004a	069a	032a	074b	126b	176a	185a	000b	080b	027a	135b	141a	204b	105a
10	023b	133a	048a	051b	095a	160b	119a	033b	071b	052a	062a	129b	118b	101a	015b	205a	082b	145a	009b
11	099a	043a	096b	109a	123a	006a	104a	134a	113a	026b	049b	091a	106b	100b	055b	103a	112a	196b	054b

## 9.5. MATLAB 程序

### 9.5.1. 附件一的复原:

此为 0/1 比较法的复原，若要改成离差和或者离差平方和，只需要将程序中

**no(i,j)=sum((B(2, :, i)-B(1, :, j))~=zeros(1,1980));**

一句替换为

**no(i,j)=sum(abs(B(2, :, i)-B(1, :, j)))或者 no(i,j)=sum((B(2, :, i)-B(1, :, j)).^2)即可。**

```
clear all; close all;clc;
```

```
A=zeros(1980,72,19);
```

```
bianhao=zeros(1,19);
```

```
for i=0:9
```

```
    A(:, :, i+1)=imread(strcat('附件 1\00',num2str(i),'.bmp'));
```

```
    A(:, :, i+1)=A(:, :, i+1)/255.0;
```

```
end
```

```
for i=0:8
```

```
    A(:, :, i+11)=imread(strcat('附件 1\01',num2str(i),'.bmp'));
```

```
    A(:, :, i+11)=A(:, :, i+11)/255.0;
```

```
end
```

%导入数据

```
B=zeros(2,1980,19);
```

```
for i=1:19
```

```
    B(1, :, i)=A(:, 1, i);
```

```
    B(2, :, i)=A(:, 72, i);
```

```
    if B(1, :, i)==zeros(1,1980)+1
```

```
        bianhao(i)=1;
```

```
    end
```

```
    if B(2, :, i)==zeros(1,1980)+1
```

```
        bianhao(i)=19;
```

```
    end
```

```
end
```

%判断首末张

```
no=zeros(19,19);
```

```

i=9;j=1;num=2;total=A(:, :, 9);
while i~=7
    for j=1:19
%         a1=0;a2=0;a3=0;
%         ave1=sum(B(2,:,i))/1980;
%         ave2=sum(B(1,:,j))/1980;
%         for tt=1:1980
%             a1=a1+(B(2,tt,i)-ave1)*(B(1,tt,j)-ave2);
%             a2=a2+(B(2,tt,i)-ave1)^2;
%             a3=a3+(B(1,tt,j)-ave2)^2;
%         end
%         no(i,j)=a1/sqrt(a2*a3);           %相关系数算法
%         no(i,j)=sum((B(2,:,i)-B(1,:,j))~=zeros(1,1980)); %0/1 比较法
    end
    min=inf;mini=i;
    for j=1:19
        if no(i,j)<min
            min=no(i,j);
            mini=j;
        end
    end
    i=mini;
    bianhao(mini)=num
    total=[total A(:, :, i)];           %拼接图像
    num=num+1;
end
for i=1:19
    no(i,:)=sort(no(i,:));
end
S=0;
for i=1:19
    if i~=7
        S=S+(no(i,2)-no(i,1))./(no(i,2)+no(i,1));
    end
end
end
S=S/19;           %精确度函数

imshow(total);

```



```
imwrite(total,'1.bmp','bmp');
```

### 9.5.2. 附件二的复原:

与附件一方法相同，程序类似：

```
clear all; close all; clc;
A=zeros(1980,72,19);
bianhao=zeros(1,19);
for i=0:9
    A(:,:,i+1)=imread(strcat('附件 2\00',num2str(i),'.bmp'));
    A(:,:,i+1)=A(:,:,i+1)/255.0;
end
for i=0:8
    A(:,:,i+11)=imread(strcat('附件 2\01',num2str(i),'.bmp'));
    A(:,:,i+11)=A(:,:,i+11)/255.0;
end %导入数据
B=zeros(2,1980,19);
for i=1:19
    B(1,:,i)=A(:,1,i);
    B(2,:,i)=A(:,72,i);
    if B(1,:,i)==zeros(1,1980)+1
        bianhao(i)=1;
    end
    if B(2,:,i)==zeros(1,1980)+1
        bianhao(i)=19;
    end
end %判断首末张
no=zeros(19,19);
i=4;j=1;num=2;total=A(:,:,4);
while i~=5
    for j=1:19
        % a1=0;a2=0;a3=0;
        % ave1=sum(B(2,:,i))/1980;
        % ave2=sum(B(1,:,j))/1980;
        % for tt=1:1980
        % a1=a1+(B(2,tt,i)-ave1)*(B(1,tt,j)-ave2);
        % a2=a2+(B(2,tt,i)-ave1)^2;
        % a3=a3+(B(1,tt,j)-ave2)^2;
        % end
```

```

%          no(i,j)=a1/sqrt(a2*a3);          %相关系数算法
          no(i,j)=sum((B(2,:,i)-B(1,:,j)).^2); %离差平方和
end
min=inf;mini=i;
for j=1:19
    if no(i,j)<min
        min=no(i,j);
        mini=j;
    end
end
i=mini;
bianhao(mini)=num
total=[total A(:,i)];          %拼接图像
num=num+1;
end
for i=1:19
    no(i,:)=sort(no(i,:));
end
S=0;
for i=1:19
    if i~=5
        S=S+(no(i,2)-no(i,1))./(no(i,2)+no(i,1));
    end
end
end
S=S/19;          %精确度函数

imshow(total);
imwrite(total,'2.bmp','bmp');

```

### 9.5.3. 附件三的复原（中文部分 1）：

此为为中文碎片分类

```

clear all; close all;clc;
A=zeros(180,72,209);
bianhao=zeros(11,19);
for i=0:9
    A(:, :, i+1)=imread(strcat('附件 3\00',num2str(i),'.bmp'));
    A(:, :, i+1)=A(:, :, i+1)/255.0;
end

```

```

for i=10:99
    A(:,i,i+1)=imread(strcat('附件 3\0',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
for i=100:208
    A(:,i,i+1)=imread(strcat('附件 3\1',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
%读入数据
t1=0;t2=0;t3=0;t4=0;
for i=1:209
    if (sum(A(1,:,i))~=72)&(sum(A(180,:,i))~=72)
        t1=t1+1; %上有下有
    end
    if (sum(A(1,:,i))~=72)&(sum(A(180,:,i))==72)
        t2=t2+1; %上有下无
    end
    if (sum(A(1,:,i))==72)&(sum(A(180,:,i))~=72)
        t3=t3+1; %上无下有
    end
    if (sum(A(1,:,i))==72)&(sum(A(180,:,i))==72)
        t4=t4+1; %上无下无
    end
end
end

```

#### 9.5.4. 附件三的复原（中文部分 2）：

搜索峰值及统计峰值的程序，此为上无下有情况下，从底行到 **baseline** 的像素点统计。其余统计程序基本类似。

```

clear all; close all;clc;
A=zeros(180,72,209);
bianhao=zeros(11,19);
for i=0:9
    A(:,i,i+1)=imread(strcat('附件 3\00',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
for i=10:99
    A(:,i,i+1)=imread(strcat('附件 3\0',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
for i=100:208

```

```

A(:,i,i+1)=imread(strcat('附件 3\',num2str(i),'.bmp'));
A(:,i,i+1)=A(:,i,i+1)/255.0;
end %读入数据
t1=0;t2=0;t3=0;t4=0;
t=0;p=zeros(1,180);
for i=1:209

    if (sum(A(1,:,i))==72)&(sum(A(180,:,i))~=72)
        t3=t3+1; %上无下有
        t=1;
        while sum(A(180-t,:,i))~=72
            t=t+1; %计算字数向上延伸的像素点个数
        end
        plot(i,t,'*');hold on %绘制分布散点图
        p(t)=p(t)+1;
    end
end
grid on
figure
findpeaks(p) %自动寻峰
area(p); %绘制分布频数图
9.5.5. 附件三的复原（中文部分 3）：
此为每行内的单独求解（下面有字情况）
clear all; close all;clc;
A=zeros(180,72,209);
h=2; %h=2,3,4,5
for i=0:9
    A(:,i,i+1)=imread(strcat('附件 3\00',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
for i=10:99
    A(:,i,i+1)=imread(strcat('附件 3\0',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end
for i=100:208
    A(:,i,i+1)=imread(strcat('附件 3\',num2str(i),'.bmp'));
    A(:,i,i+1)=A(:,i,i+1)/255.0;
end

```

```

t=0;p=zeros(1,180);tt=0;
for i=1:209
    if sum(A(180,:,i))~=72
        t=1;tt=tt+1;
        while sum(A(180-t,:,i))~=72
            t=t+1;
        end
        p(t)=p(t)+1;
    end
end

```

```

P=[1 6 12 18 24 36]; %峰值
B=zeros(6,100);
st=zeros(1,6)+1;

```

```

for i=1:209
    if sum(A(180,:,i))~=72
        t=1;
        while sum(A(180-t,:,i))~=72
            t=t+1;
        end
        for k=1:6
            if abs(t-P(k))<=2
                B(k,st(k))=i;
                st(k)=st(k)+1;
                break
            end
        end
    end
end
end
end

```

%分组

```

bianhao=zeros(1,19);
C=zeros(2,180,19);
for i=1:19
    C(1,:,i)=(A(:,1,B(h,i))+A(:,2,B(h,i))+A(:,3,B(h,i))+A(:,4,B(h,i))+A(:,5,B(h,i)))./5;

    C(2,:,i)=(A(:,72,B(h,i))+A(:,71,B(h,i))+A(:,70,B(h,i))+A(:,69,B(h,i))+A(:,68,B(h,i)))./5;

```

```

        if C(1,:,i)==zeros(1,180)+1
            bianhao(i)=1;be=i;
        end
        if C(2,:,i)==zeros(1,180)+1
            bianhao(i)=19;en=i;
        end
    end
    %求解第一和最后一张碎纸片

    for i=1:19
        C(1,:,i)=A(:,1,B(h,i));
        C(2,:,i)=A(:,72,B(h,i));
    end

    no=zeros(19,19);
    i=be;j=1;num=2;total=A(:,B(h,be));

    while i~=en
        ii=i;
        for j=1:19
            no(i,j)=sum((C(2,:,i)-C(1,:,j)).^2);
            % 离差平方和法
            no(j,i)=no(i,j);
        end
        min=inf;mini=i;
        for j=1:19
            if (no(i,j)<min)&(bianhao(j)==0)
                min=no(i,j);
                mini=j;
            end
        end
        i=mini;
        bianhao(i)=num
        total=[total A(:,B(h,i))];
        num=num+1;
        if num==19
            break
        end
    end
    total=[total A(:,B(h,en))];

```

```

imshow(total);
%以下为人工干预
%%
%改一次
if h==2
total=[];
shunxu=[6,13,14,16,1,2,7,3,17,12,10,15,19,4,9,5,18,11,8];
BB=[];
for i=1:19
    for j=1:19
        if shunxu(j)==i
            total=[total A(:,B(h,j))];
            BB=[BB B(h,j)];
        end
    end
end
BB=BB-1;
imshow(total)
end
%%
%改一次
if h==4
    total=[];
shunxu=[3,8,1,15,14,16,9,5,17,18,2,13,7,4,10,12,19,6,11];
BB=[];
for i=1:19
    for j=1:19
        if shunxu(j)==i
            total=[total A(:,B(h,j))];
            BB=[BB B(h,j)];
        end
    end
end
BB=BB-1;
imshow(total)
end
%%
%改 0 次

```

```

if h==3
    total=[];
    shunxu=bianhao;
    BB=[];
    for i=1:19
        for j=1:19
            if shunxu(j)==i
                total=[total A(:,j),B(h,j)];
                BB=[BB B(h,j)];
            end
        end
    end
    BB=BB-1;
    imshow(total)
end
%%
%改 0 次
if h==5
    total=[];
    shunxu=bianhao;
    BB=[];
    for i=1:19
        for j=1:19
            if shunxu(j)==i
                total=[total A(:,j),B(h,j)];
                BB=[BB B(h,j)];
            end
        end
    end
    BB=BB-1;
    imshow(total)
end

```

#### 9.5.6. 附件三的复原（中文部分 4）：

此为每行内的单独求解（上面有字情况）

```
clear all; close all;clc;
```

```
h=2;%h=2,3,4,5
```

```
A=zeros(180,72,209);
```

```
bianhao=zeros(11,19);
```



```

for i=0:9
    A(:,:,i+1)=imread(strcat('附件 3\00',num2str(i),'.bmp'));
    A(:,:,i+1)=A(:,:,i+1)/255.0;
end
for i=10:99
    A(:,:,i+1)=imread(strcat('附件 3\0',num2str(i),'.bmp'));
    A(:,:,i+1)=A(:,:,i+1)/255.0;
end
for i=100:208
    A(:,:,i+1)=imread(strcat('附件 3\'',num2str(i),'.bmp'));
    A(:,:,i+1)=A(:,:,i+1)/255.0;
end

t=0;p=zeros(1,180);
for i=1:209
    if sum(A(1,:,i))~=72
        t=1;
        while sum(A(t,:,i))~=72
            t=t+1;
        end
        p(t)=p(t)+1;
    end
end

P=[5 17 23 29 36 42]; %峰值
B=zeros(6,100);
st=zeros(1,6)+1;

for i=1:209
    if sum(A(1,:,i))~=72
        t=1;
        while sum(A(t,:,i))~=72
            t=t+1;
        end
        for k=1:6
            if abs(t-P(k))<=2
                B(k,st(k))=i;
                st(k)=st(k)+1;
            end
        end
    end
end

```

```

        break
    end
end
end
end
end                                     %分组

bianhao=zeros(1,19);
C=zeros(2,180,19);
for i=1:19
    C(1,:,i)=(A(:,1,B(h,i))+A(:,2,B(h,i))+A(:,3,B(h,i))+A(:,4,B(h,i))+A(:,5,B(h,i)))./5;
    C(2,:,i)=(A(:,72,B(h,i))+A(:,71,B(h,i))+A(:,70,B(h,i))+A(:,69,B(h,i))+A(:,68,B(h,i)))./5;
    if C(1,:,i)==zeros(1,180)+1
        bianhao(i)=1;be=i;
    end
    if C(2,:,i)==zeros(1,180)+1
        bianhao(i)=19;en=i;
    end
end                                     %第一和最后

for i=1:19
    C(1,:,i)=A(:,1,B(h,i));
    C(2,:,i)=A(:,72,B(h,i));
end

no=zeros(19,19);
i=be;j=1;num=2;total=A(:,:,B(h,be));

while i~=en
    ii=i;
    for j=1:19
        no(i,j)=sum((C(2,:,i)-C(1,:,j)).^2);
        % no(j,i)=no(i,j);
    end
    min=inf;mini=i;
    for j=1:19
        if (no(i,j)<min)&(bianhao(j)==0)
            min=no(i,j);

```

```

        mini=j;
    end
end
i=mini;
bianhao(i)=num
total=[total A(:,B(h,i))];
num=num+1;
if num==19
    break
end
end
total=[total A(:,B(h,en))];
imshow(total);
%一下为人工干预
%%
%改一处，此在下面有字的 h=2
if h==5
total=[];
shunxu=[15,2,17,19,18,1,11,4,5,14,3,10,8,6,12,9,7,13,16];
BB=[];
for i=1:19
    for j=1:19
        if shunxu(j)==i
            total=[total A(:,B(h,j))];
            BB=[BB B(h,j)];
        end
    end
end
end
BB=BB-1;
imshow(total)
end
%%
%改 2 处，此在下面有字的 h=5
if h==2
total=[];
shunxu=[16,17,5,15,6,1,3,19,7,13,10,18,9,11,2,4,14,8,12];
BB=[];%每行的编号
for i=1:19

```

```

        for j=1:19
            if shunxu(j)==i
                total=[total A(:,B(h,j))];
                BB=[BB B(h,j)];
            end
        end
    end
end
BB=BB-1;
imshow(total)
end
%%
%改 0 次
if h==4
    total=[];
    shunxu=bianhao;
    BB=[];
    for i=1:19
        for j=1:19
            if shunxu(j)==i
                total=[total A(:,B(h,j))];
                BB=[BB B(h,j)];
            end
        end
    end
end
BB=BB-1;
imshow(total)
end

```

**9.5.7. 附件三的复原（中文部分 5）：**  
按行统计像素分布情况  
A=imread('test1.bmp');  
p=zeros(1,180\*11);  
for i=1:11\*180  
 for j=1:19\*72  
 if A(i,j)~=0  
 p(i)=p(i)+1;  
 end  
 end  
end  
end

```
plot(p);
```

### 9.5.8. 附件四的复原（英文部分 1）：

对于给定的顺序，还原出图像

```
A=load('2.txt');
```

```
i=1;j=1;
```

```
T=[];TT=[];
```

```
B=zeros(72,180);
```

```
while j<=11
```

```
    if A(j,i)<10
```

```
        B=imread(strcat('附件 4\00',num2str(A(j,i)),'.bmp'));
```

```
    % B=B./255.0;
```

```
    else
```

```
        if A(j,i)<100
```

```
            B=imread(strcat('附件 4\0',num2str(A(j,i)),'.bmp'));
```

```
            % B=B./255.0;
```

```
        else
```

```
            B=imread(strcat('附件 4\1',num2str(A(j,i)),'.bmp'));
```

```
            %B=B./255.0;
```

```
        end
```

```
    end
```

```
    TT=[TT B];
```

```
    i=i+1;
```

```
    if i>19
```

```
        i=1;
```

```
        j=j+1;
```

```
        T=[T TT'];
```

```
        TT=[];
```

```
    end
```

```
end
```

```
T=T';
```

```
imshow(T);
```

```
imwrite(T,'5.bmp','bmp');
```

### 9.5.9. 附件四的复原（英文部分 2）：

此为求解程序，需要在末尾几行设置断点，必要时进行人工干预。

```
clear
```

```
A=[191 75 11 154 190 184 2 104 180 64 106 4 149 32 204 65 39 67
```

```
147];%已知的横排
```

```
B=[191 201 86 19 159 20 208 70 132 171
```

```

81];                                %已知的纵排
for i=2:11
    for j=2:19
        if i==2                    %判断若为第二行，读取顶行数据
            if A(j)<10
                TO=imread(strcat('附件 4\00',num2str(A(j)),'.bmp'));
            else
                if A(j)<100
                    TO=imread(strcat('附件 4\0',num2str(A(j)),'.bmp'));
                else
                    TO=imread(strcat('附件 4\',num2str(A(j)),'.bmp'));
                end
            end
        else                        %否则读取前一行数据
            if result(i-1,j)<10
                TO=imread(strcat('附件 4\00',num2str(result(i-1,j)),'.bmp'));
            else
                if result(i-1,j)<100
                    TO=imread(strcat('附件 4\0',num2str(result(i-1,j)),'.bmp'));
                else
                    TO=imread(strcat('附件 4\',num2str(result(i-1,j)),'.bmp'));
                end
            end
        end
    end

    if j==2                        %判断若为第二列，读取最左列数据
        if B(i)<10
            LE=imread(strcat('附件 4\00',num2str(B(i)),'.bmp'));
        else
            if B(i)<100
                LE=imread(strcat('附件 4\0',num2str(B(i)),'.bmp'));
            else
                LE=imread(strcat('附件 4\',num2str(B(i)),'.bmp'));
            end
        end
    else                            %否则读取前一列数据
        if last<10
            LE=imread(strcat('附件 4\00',num2str(last),'.bmp'));

```

```

        else
        if last<100
        LE=imread(strcat('附件 4\0',num2str(last),'.bmp'));
        else
        LE=imread(strcat('附件 4\',num2str(last),'.bmp'));
        end
        end
    end
    LE=LE(:,72);          %提取最左边数据
    TO=TO(180,:);        %提取最上边数据
    min=inf;
    for ii=0:208          %读取自身数据
        if ~any(A==ii)&~any(B==ii)
            if ii<10
                DO=imread(strcat('附件 4\00',num2str(ii),'.bmp'));
            else
                if ii<100
                    DO=imread(strcat('附件 4\0',num2str(ii),'.bmp'));
                else
                    DO=imread(strcat('附件 4\',num2str(ii),'.bmp'));
                end
            end
        end
        DOle=DO(:,1);      %提取最左边数据
        DOto=DO(1,:);      %提取最上边数据
        no1=sum((DOle-LE).^2); %离差平方和法
        no2=sum((DOto-TO).^2);
        no=no1+no2;        %综合比较法
        if no<min & ~any(A==ii) & ~any(B==ii)
            min=no;        %选择并记录最小值
            last=ii;
        end
    end
    result(i,j)=last      %存储数据，随时查看修正
end
end
end

```

#### 9.5.10. 附件五的复原（双面）：

此为求解程序，需要在末尾几行设置断点，必要时进行人工干预。

```

clear;
A={'136a' '047b' '020b' '164a' '081a' '189a' '029b' '018a' '108b'
  '066b' '110b' '174a' '183a' '150b' '155b' '140b' '125b' '111a'
  '078a'};%已知的横排
B={'136a' '005b' '143a' '083b' '090b' '013b' '035b' '172b' '105b' '009a'
  '054a'};%已知的纵排
for i=2:11
    for j=2:19
        if i==2 %判断若为第二行，读取顶行数据
            TO=imread(strcat('附件 5\',A{j},'.bmp'));
        else %否则读取前一行数据
            TO=imread(strcat('附件 5\',result{i-1,j},'.bmp'));
        end
        if i==2 %判断若为第二列，读取最左列数据
            LE=imread(strcat('附件 5\',B{i},'.bmp'));
        else %否则读取前一列数据
            LE=imread(strcat('附件 5\',last,'.bmp'));
        end

        LE=LE(:,72); %提取最左边数据
        TO=TO(180,:); %提取最上边数据
        min=inf;
        for ii=0:208 %读取自身数据
            % if ~any(A==ii)&~any(B==ii)
            if ii<10
                DO1=imread(strcat('附件 5\00',num2str(ii),'a.bmp'));
                DO2=imread(strcat('附件 5\00',num2str(ii),'b.bmp'));
            else
                if ii<100
                    DO1=imread(strcat('附件 5\0',num2str(ii),'a.bmp'));
                    DO2=imread(strcat('附件 5\0',num2str(ii),'b.bmp'));
                else
                    DO1=imread(strcat('附件 5\',num2str(ii),'a.bmp'));
                    DO2=imread(strcat('附件 5\',num2str(ii),'b.bmp'));
                end
            end

            DO1le=DO1(:,1); %提取 a 最左边数据
            DO1to=DO1(1,:); %提取 a 最上边数据
        end
    end
end

```



```

        DO2le=DO2(:,1);          %提取 b 最左边数据
        DO2to=DO2(1,:);         %提取 b 最上边数据
        noa=sum((DO1le-LE).^2)+sum((DO1to-TO).^2); %离差平方和法
        nob=sum((DO2le-LE).^2)+sum((DO2to-TO).^2); %离差平方和法
        if noa<nob
            f='a';no=noa;
        else
            f='b';no=nob;
        end
        if no<min% & ~any(A==ii) & ~any(B==ii)
            min=no;              %选择并记录最小值
            if ii<10
                last={strcat('00',num2str(ii),f)};
            else
                if ii<100
                    last={strcat('0',num2str(ii),f)};
                else
                    last={strcat(num2str(ii),f)};
                end
            end
        end
    end
end
result(i,j)=last              %存储数据，随时查看修正
end
end

```