A Character Recognition Algorithm for Unhealthy-text Embedded in Web Images

Sun Yan, Zhou Xue-guang

College of Electronic Engineering, Naval University of Engineering, Wuhan China

yzisun@126.com

Abstract: The Web with unhealthy-text embedded in images could not be filtered with the method of existing Web filtering technique. Based on the technique of Optical Character Recognition and unhealthy-text filtering, a character recognition algorithm for unhealthy-text embedded in Web images (CRAU) is proposed, in order to prevent the spread of unhealthy-text images in Internet which escape from the Web filtering system. The experiments show that recognition ratio of CRAU for small style characters (from 16-point to 9-point) is more than 98%. Therefore, CRAU is feasible in the engineering project.

Key words: Web filtering, character recognition, embedded in Web image, unhealthy-text

嵌入网页图片中的不良文本文字识别算法

孙艳　周学广

海军工程大学电子工程学院，湖北省 武汉市 430033

yzisun@126.com

【摘要】现有的信息过滤技术不能过滤将不良文本嵌入在图片中的网页。针对互联网中的不法分子为了逃避当前网页过滤系统将不良文本变形为图片的手段，利用现有的 OCR 技术和不良文本过滤技术，提出了一种嵌入网页图片中的不良文本文字识别算法（CRAU）。实验表明，CRAU 对于从三号到小五号汉字识别率达 98%以上，具有较好的应用价值。

【关键词】网页过滤；文字识别；嵌入网页图片；不良文本

1、引言

随着网络的飞速发展和上网人数的大量增加，网络作为一种新的媒体传播介质，具有高匿名性、高开放性和无地域性，已成为垃圾信息制造者的主要场所。目前网页内容过滤技术主要包括以下三种[1]：基于 URL 的过滤，基于关键词的过滤和基于图像内容的过滤。

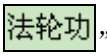
基于 URL 的过滤是将已知有害页面和网站收集到 URL 禁止列表库，将有益或需要的网页和网站收集到 URL 允许列表库，即设置网页黑白名单。但是由于这种过滤方式主要依靠人工，随着网页的动态变化和网页数量的递增，采用 URL 过滤会给管理人员带来很大的难度。基于关键词的过滤是对网页中的文本进行关键词匹配，通过设定阈值来判断是否过滤该网页，近两年用拼音、繁体字、特殊符号等手段代替或夹杂敏感或不良信息的情况频频出现，国内有部分学者已研究这类情况[2][3]。基于图像内容的过滤主要通过肤色检测过滤色情网站。对于宣传暴力、反动、色情等不良文本图片的研究尚未见诸报道，目前不少网页及邮件应用此手段来逃避现有的不良网页过滤系统。

基于上面的分析，本文利用现有的 OCR 技术和不良文本过滤技术，提出了一种嵌入在网页图片中的不良文本文字识别算法，以期解决将不良文本嵌入在图片中的网页过滤难题。

2、原理与设计

文字识别技术研究已久，现有的 OCR 软件，如汉王 OCR[4]，识别清晰、规范的印刷体汉字的速度和精度都很高，但没有公布主要技术及代码，使得这种技术难以嵌入其他软件中。本文提出的 CRAU 在打开网页时，可直接获取网页中的图片并识别和保存文本图片中的文字。该算法现已成功运用于基于关键词的网页过滤软件中，实验表明，该算法针对将不良文本嵌入在图片中的网页过滤具有可行性和有效性。

2.1 相关问题描述

中文主动干扰[5]是指网络攻击者了解中文特点，能够知己知彼，利用中文分词技术的困难性，采用在中文连续文本中随机夹杂符号（如宣扬邪教的信息“法?//*轮*　！功”)和/或用繁体字/同音字/英文代替（如用“法轮攻”代替“法轮功”）某个中文关键词，或将含中文关键词的文本嵌入在图片中的方法（如用图片“  ”代替“法轮功”），欺骗并绕开各种过滤器，造成网络内容安全处理效果大幅下降。

文献 3 介绍了一种柔性字符串匹配算法，该算法能够过滤采用中文主动干扰技术处理过的夹杂符号、繁体字、同音字的关键词，文献 6 研究了夹杂英文的关键词过滤情况。夹杂符号、繁体字、同音字和英文 4 种情况的关键词过滤系统已实现，并在第一届全国信息安全大赛中获奖。

978-1-935068-01-3 © 2009 SciRes.

不良文本是指宣传反动，包括反党、反政府、破坏国家统一等文本（如“法轮功、藏独”），宣传暴力、恐怖、色情等（如“性交易”）。

假定网页 $web=\{txt, pic\}$，$jpic \in pic$，其中 txt 表示网页中的文本集合，pic 表示网页中的图片集合，$jpic$ 表示夹杂敏感或不良信息的文本图片集合。$jpic$ 具有以下几个特点：

（1） 是屏幕显示汉字，不是由扫描仪扫描进去的印刷体，未发生相对变形；

（2） 字体一般为网络通用字体，与 txt 中字体相似或相同，通常为宋体三号到小五号；

（3） 背景与 txt 背景相同，比较单一；

（4） 可能包含多个敏感或不良关键词。

2.2 算法设计

由于中文主动干扰手段的多样化，基于关键词的网页过滤方法需不断完善，图 1 是一个完整的基于关键词的网页过滤软件的流程，其中变形关键词的挖掘包括夹杂字符、拼音、英文[6]、同音字、繁体字等关键词挖掘[7]，虚框内是本文研究的 CRAU 模块。

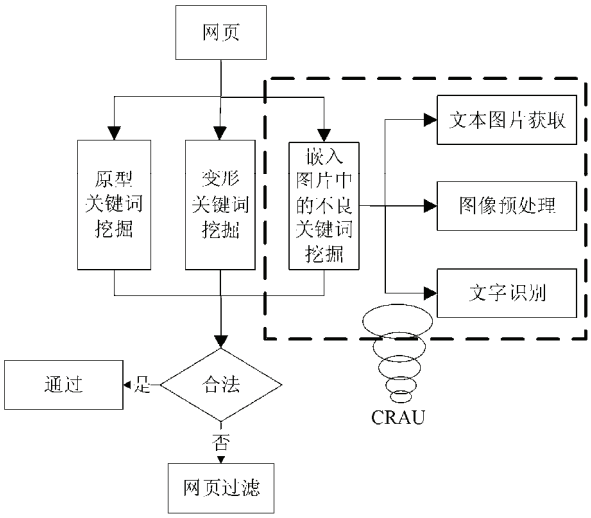


图 1　基于关键词的网页过滤流程图

本文算法设计步骤如下：

Step1：建立不良文本词汇库；

Step2：打开网页，获取网页中所有文本图片；

Step3：对文本图片进行图像预处理，包括灰度化、二值化、行分割、字分割、汉字归一化和细化；

Step4：建立汉字特征库，提取每个文字图片的特征，识别文字并保存结果。

3、算法实现

3.1 图像获取

图像获取分两步。第一步，图标过滤。图片长宽尺寸小于给定的阈值 $w \le w_t, h \le h_t$ 的图片认为是图标图片，可直接过滤掉。第二步，文本图片获取。根据文本图片固有的特点，即图片的颜色数不多、文字边缘具有丰富的信息[8]获取得到文本图片。图像获取后对图像进行格式转换，一般是将动态的图像格式转换为静态的图像格式，通常为.bmp 格式以便计算机处理。

3.2 图像预处理

图像预处理包括图像的灰度转化及二值化、图像分割（行分割、字符分割）以及汉字归一化和细化等三部分组成。

3.2.1 图像的灰度转化及二值化

本文采用的灰度转化是给 RGB 系数加权再求和[9]，即：

$$Y = 0.3R + 0.59G + 0.11B$$

灰度图像涉及到像素的多级值，不利于对图像进行进一步的处理，所以还需要对图像进行二值化。图像的二值化是将图像上的点的灰度值置成 0 或 1，也就是整个图像呈现出明显的黑白效果。为了得到理想的二值图像，一般采用阈值分割技术，即

$$g(i,j)=\begin{cases}1; & if \quad f(i,j) \ge Lt;\\ 0; & if \quad f(i,j)<Lt.\end{cases}$$

其中，$f(i,j)$ 为原灰度图像，$g(i,j)$ 为二值化后的图像，L_t 为阈值，L_t 的选择决定了二值化的效果，根据本文所研究图像的文字和背景比较单一，采用全局阈值法，取 $L_t=128$。

3.2.2 图像分割

图像分割的目的是将待识别的文字分割出来，包括行分割和字符分割。文字在图片中是以像素点的形式呈现出来的，错误的像素点的分割导致单个文字的不完整，从而影响文字的识别，因而图像分割是文字识别的关键部分。

(1) 行分割

行分割是指将两行或两行以上的文字行分割出来，通常采用投影法。具体操作：对图像进行水平投影，根据文字行与行之间的空白域、文字行的高度范围以及判断某一投影点上的黑点数可以判断文字行的上边界和下边界。如果某一扫描线上的黑点数小于一个很小的阈值，可认为该线为空白线，从空白线开始逐线向下扫描，扫描到一条线的黑点个数大于某一个

阈值，就找到一个字符行的上界；继续扫描，如果扫描到一条线的黑点个数小于某一个阈值，就找到了一个字符行的下界。

(2)字符分割

相对于行分割，字符分割要复杂的多。汉字是方块字，高宽比近似为 1，利用这个特性可以采用改进的垂直投影方法进行字符分割，字符分割算法如图 2 所示：

```
input: a text image
output: each character's segmentation mark
process:
1)  vertical project each line, put the trough into the array
    P[1,2,…,n];
2)  for i=1 to n
3)    if (i mod 2=0)
4)      if ((P[i]-P[i-1])<T1)
5)        if ((P[i]-P[i-3])<T2)
6)          if ((P[i-2]-P[i-3])<T2)
7)            if ((P[i+2]-P[i-1])<T1)
8)              if ((P[i+4]-P[i-1])<T2)
9)                delete P[i],P[i+1], P[i+2],P[i+3];
10)             else delete P[i],P[i+1];
11)           else if ((P[i+2]-P[i-1])<T2)
12)             delete P[i],P[i+1];
13)           else delete P[i-1],P[i];
14)         else delete P[i-2],P[i-1];
15)       else if ((P[i+2]-P[i-1])<T2)
16)         delete P[i],P[i+1];
17)       else if ((P[i+4]-P[i-1])<T2)
18)         delete P[i],P[i+1], P[i+2],P[i+3];
19)       else delete P[i],P[i-1];
```

图 2 字符分割算法

3.2.3 汉字归一化和细化

汉字归一化包括位置归一化和尺寸归一化。位置归一化即将汉字点阵图形移到规范的位置上来，可根据文字的中心或重心来进行；尺寸归一化就是将分割好的每个文字根据上下左右边界按比例线性放大或缩小成同一尺寸的文字。

细化又称为骨架化，即将图像中线条宽度大于 1 个像素点的线条细化成 1 个像素点，同时要求保存模式的基本结构和特征。细化算法按照处理过程中的迭代方式又分为串行和并行。典型的细化算法有 Hildich 算法[10]、SPTA 算法[11]和 ZS 算法[12]。通过对比各细化算法的效果及消耗的时间，本文对 ZS 算法[12]进行了改进。改进的 ZS 算法采用串行删除的方法，将扫描到的每一个值为 1 且同时满足下列情况的像素点置 0（假定背景是白像素点为 0，汉字是黑像素点为 1）：

(1).$2 \le N(P) \le 6$;

(2).$S(P) = 1$;

(3).$P1 * P3 * P5 = 0$;

(4).$P1 * P5 * P7 = 0$.

其中 $N(P)$ 表示 $P1...P8$ 为非零的个数， $S(P)$ 表示 $P1...P8$ 序列值跳变的次数，当前处理点 P0 与它的 8 邻点表示为如表 1 所示：

表 1 当前点 P0 与它的 8 邻点

P4	P3	P2
P5	P0	P1
P6	P7	P8

为减少毛刺，上述细化后的图片再与图 3 中的消除模板比较，若满足图 3 中的(a)、(b)、(c)和(d)中任何一个，则将当前像素点置 0：

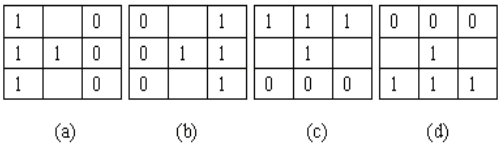


图 3 消除模板

3.3 汉字识别

3.3.1 特征提取

特征提取的目的是从图片中提取出有关文字的本质信息，滤掉不必要的信息。特征可以分为统计特征和结构特征，其中统计特征又包括局部特征和全局特征。结构特征是指汉字的组成结构特征，包括特征点(端点，拐点，三叉点，交叉点)和笔画序列等。统计特征有变换特征、矩特征、笔画密度特征、外围特征、网格特征等。

3.3.2 识别方法

识别汉字的方法根据特征的提取可分为结构模式识别、统计模式识别及两者的结合。本文采用三级分类器集成识别汉字，第一级为粗分类，采用 8 维四周特征（分类器 1.1）和封闭区域个数特征（分类器 1.2），第二级为细分类，采用横竖笔画骨架特征（分类器 2），第三级分类区分相似字，采用结构点特征（分类器 3），如图 4 所示：

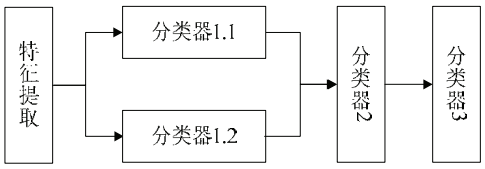


图4 三级分类器集成结构图

978-1-935068-01-3 © 2009 SciRes.

(1)分类器1

特征选取：8 维四周特征和封闭区域个数特征。

由于汉字的边缘信息比较丰富，将汉字的四周均匀分成 8 份，提取每份的像素点数，得到 8 维四周特征依次为 $(m_1, m_2..m_8)$，如图 5 所示：

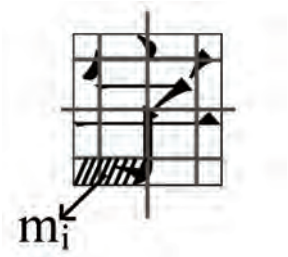


图 5 汉字的 8 维四周特征

四周特征和封闭区域个数特征的提取对象都是细化前文字。封闭区域个数提取比较简单，如“可”只有一个封闭区域，而“器”有四个封闭区域。

匹配过程：

建立 8 维四周特征库；求出待识别文字的 8 维四周特征 $(m_1, m_2..m_8)$；采用绝对值距离进行相似性比较 $S(m_i, n_i) = |m_i - n_i|$，求 $\sum_{i=1}^{8} S(m_i, n_i)$，按从小到大顺序得到前 $k1$ 个汉字，得到字符集 $C1.1 = \{C_1, C_2, ..., C_{k1}\}$。

建立封闭区域个数特征库；求出待识别文字封闭区域个数 k；在特征库中寻找封闭区域数为 $k \pm \Delta s$ 的前 $k2$ 个汉字，得到字符集 $C1.2 = \{C_1, C_2, ..., C_{k2}\}$，其中 Δs 为置信宽度；$C1 = C1.1 \wedge C1.2$。

(2)分类器2

特征选取：横竖笔画端点坐标值。

横竖笔画骨架特征提取的对象是细化后文字，因为细化对于横竖笔画比较稳定，而对于撇捺笔画可能产生斜率变化。当扫描到一条横线像素点个数大于一个阈值时即认为是横笔画，同理扫描到一条竖线像素点个数大于一个阈值时即认为是竖笔画，记下横笔画和竖笔画的端点坐标。

匹配过程：

建立横竖笔画骨架特征库；求出待识别文字横竖笔画端点坐标值；采用绝对值距离比较横笔画和竖笔画长度相似性，即

$$\sum_{i=1}^{l} \left| \sqrt{(x_{i1} - x_{i2})^2 + (y_{i1} - y_{i2})^2} - \sqrt{(x'_{i1} - x'_{i2})^2 + (y'_{i1} - y'_{i2})^2} \right| \le T_c$$

其中 $x_{i1}, x_{i2}, y_{i1}, y_{i2}$ 分别表示待识别汉字横笔画或竖笔画两端点的横坐标和纵坐标值，$x'_{i1}, x'_{i2}, y'_{i1}, y'_{i2}$ 分别表示特征库中横笔画或竖笔画两端点的横坐标和纵坐标值，l 表示横笔画和竖笔画的个数，T_c 为长度阈值。再采用欧式距离比较笔画端点，即

$$\sum_{i=1}^{l} \left| \sqrt{(x_{i1} - x'_{i1})^2 + (y_{i1} - y'_{i1})^2} \right| \le T_d \quad \text{and} \quad \sum_{i=1}^{l} \left| \sqrt{(x_{i2} - x'_{i2})^2 + (y_{i2} - y'_{i2})^2} \right| \le T_d$$

其中 T_d 为距离阈值，得到符合条件的前 $k3$ 个汉字组成的字符集 $C2 = \{C_1, C_2, ..., C_{k3}\}$。

$$C1 \wedge C2 = \begin{cases} null, & \text{refuse to recognize;} \\ C_1, & \text{the result;} \\ C_1, ..., C_k (k > 1) & \text{go to the classifier 3。} \end{cases}$$

(3)分类器3

特征选取：网格结构点特征。

对于细化后的文字图像，首先求出端点、交叉点和三叉点坐标值[13]，具体操作：第一步，扫描当前点的8邻域，按图5所示序列，求出8邻域序列值跳变的次数 n，若

$$n = \begin{cases} 1, & \text{端点;} \\ 3, & \text{三叉点;} \\ 4, & \text{交叉点。} \end{cases}$$

第二步，把图像分成均匀 8×8 份，求出每个网格内的特征点。

匹配过程：

建立特征点特征库；求出待识别文字的网格结构点坐标值；求出所有这些结构点的欧氏距离和，欧氏距离最小的为识别结果 C，即

$$C = Min\left(C_j \left| C_j = \sum_{i=1}^{n} \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2},\ 1 \le j \le k \right. \right)$$

其中 n 为结构点个数，k 为 $C1 \wedge C2$ 中汉字个数。

4、实验结果与分析

实验环境：主频2.20GHz，内存1G，操作系统为Windows XP，开发语言为Delphi。采用GB2312字符集的6763个一二级汉字作为训练样本，随机对《红楼梦》截图120张，三号、小三号、四号、小四号、五号、小五号各20张，作为第一组测试样本。联机上网获取有文本图片的100个网页作为第二组测试样本。测试结果如表2所示：

表2 测试结果表

测试样本	字数	正识率	误识率	漏识率
第一组	3897	99.45%	0.03%	0.52%
第二组	4431	98.58%	1.01%	0.41%

第一组的漏识率略高的原因是图片里的有些汉字，如“卜”比较窄小，图像字符分割时直接将其作为标点删除而漏识。第二组网页中的误识率略高，主要是因为现在的一些网络用词非常生僻，这也表明特征库有待完善。测试识别的速度是3秒/百字。

与汉王OCR 6.0 的识别结果对比如图6所示，其中（a）为待识别图片（小四号宋体），（b）为CRAU的识别结果，（c）为汉王OCR 6.0 的识别结果，可见CRAU对于小字号汉字的识别率远远高于汉王OCR。

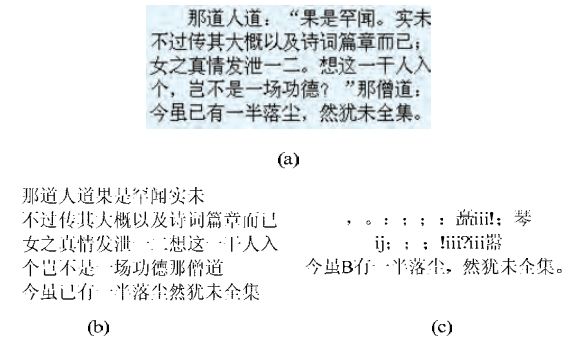


图6 识别结果对比图

（a）为待识别图片，（b）为CRAU的识别结果，（c）为汉王OCR 6.0 的识别结果

随机对《红楼梦》截图30张，其中三号、小三号、四号、小四号、五号、小五号各5张，使用CRAU与汉王OCR 6.0进行文字识别后得到的结果如图7所示：

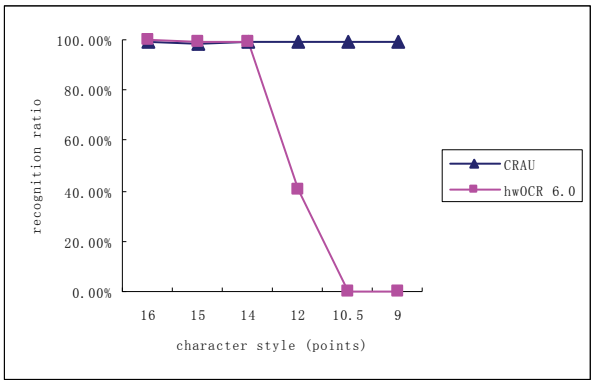


图7 识别率对比图

从图7可以看出，对于三号到四号汉字，汉王OCR 6.0 和CRAU的识别率都在99%以上，但是汉王OCR 6.0对于小四号汉字的识别率降至40%，五号和小五号汉字已不能识别，而CRAU对于三号到小五号汉字的识别率一直保持在98%以上，这也表明了CRAU更适合于识别嵌入在网页图片中的文字。

5、结论

CRAU能够在打开网页时直接获取网页中的文本图片，并识别出文本图片中的文字，对于三号到小五号汉字识别率达98%以上。通过与汉王OCR 6.0 对比，CRAU更适合于识别嵌入网页图片中的文字，表明了CRAU具有较高的实用价值。但是CRAU尚存在功能扩展，首先，需考虑颜色复杂的图片，识别藏在复杂颜色图片中的不良文本文字；其次，字库需扩大，考虑多种字体，这些都将成为我们下一步的研究目标。

致谢

感谢张焕国教授提供的基金支持，感谢亲人的鼓励和关心，感谢各位评审的专家与教授。

References (参考文献)

[1] Lee P Y, Hui S H, Fong A C M. An Intelligent Categorization Engine for Bilingual Web Content Filtering[J]. IEEE Transactions on Multimedia, 2005, 7(6): 1183-1190.

[2] Li Dun, Chao Yuanda, WAN Yueliang. Transf- ormed keywords identification in information security. Computer Engineering. 2007, 33(21): 155-159.李钝，曹元大，万月亮. 信息安全中的变形关键词的识别[J], 计算机工程,2007,33(21):155-159.

[3] Zhou Xueguang, Zhang Huanguo,. A Flexible pattern matching in Chinese strings. Proceedings of the 27th Chinese Control Conference, July. 2008, Vol.5: 610-614.周学广，张焕国，一种柔性中文字符串匹配算法[C]，第27届中国控制会议论文集，昆明，2008,5:610-614.

[4] http://www.hw99.com/tech/ocr-2.htm.

[5] Zhou Xueguang, Zhang Huanguo,. Flexible pattern matching algorithm for anti-active-jamming in Chinese string. Journal of Wuhan University (Natural Science Edition), 2009,55(1):101-104.周学广，张焕国. 抗中文主动干扰的柔性中文串匹配算法[J]，武汉大学学报（理学版），2009, 55(1):101-104.

[6] Cao Sui, Zhou Xue guang, Sun Yan, Zhang Huan guo, Research on Chinese key words mining with malicious jamming in English[C], IEEE Computer Society Press, Chengdu, China, 2009.

[7] Wu Di, Zhou Xue guang, Zhang Huan gou, The pattern matching algorithms formalized analyze in Chinese strings[C], IEEE Computer Society Press, Shanghai, China, 2008:403-407.

[8] Michael R. Lyu, Jiqiang Song, Min Cai. A Comprehensive Methd for Multilingual Video Text Detection, Localization, and Extraction[J]. IEEE Transactions on circuits and systems for video technology, 2005,15(2):243-255.

[9] JIANG Xiangang. Software design for digital imagies recognition. Beijing: China WaterPower Press, 2008. 蒋先刚. 数字图像模式识别工程软件设计[M]. 北京：中国水利水电出版社, 2008:19-20.

[10] Naccache N J, Shinghal R. An investigation into the skeletonization approach of Hilditch[J]. Pattern Recognition, 1984:17(3):274-284.

978-1-935068-01-3 © 2009 SciRes.

[11] Naccache N J, Shinghal R. SPTA: A Proposed Algorithm for Thinning Binary Patterns. IEEE Trans. SMC, 1984, 14(3): 409-418.

[12] Zhang Y Y, Sun C Y. A Fast Parallel Algorithm for Thinning Digital Patterns[J]. Communications of the ACM , 1984, 27(6): 236-539.

[13] Dawule Abuduhayer, Gulila Adonbiek, Hand-written Kazakh character recognition system using artificial network. Computer Engineering and Applications. 2008,44(1): 225-228.达吾勒·阿布都哈依尔，古丽拉·阿东别克. 基于ANN的哈萨克文手写文字识别系统的研究[J],计算机工程与应用，2008,44(1): 225-228.