

一种云端信息安全字形的生成模型

张 莉¹ 栗青生² 刘 泉³

(湖北经济学院工商管理学院 武汉 430205)¹ (浙江传媒学院电子信息学院 杭州 310018)²

(武汉理工大学信息工程学院 武汉 430070)³

摘 要 提出了一种云端信息安全字形的生成模型。该模型将汉字的字形抽象为汉字结构模式和汉字的风格模式,然后通过定义有效的汉字结构输出和汉字笔画生成方案,动态地生成了可用于信息安全保护的信息安全字形。该模型实现了汉字字形的 Web 存储和在客户端的特征字形的输出的监控,克服了现代汉字由于整字编码而导致的汉字信息在云端安全服务方面的不足,为汉字信息的云端存储和云端数据安全服务保护提供了一种有效的策略和方法;同时,也为设计更深层次的云端汉字信息服务系统奠定了基础。

关键词 汉字,笔画,字形描述,字形生成,信息安全

中图分类号 TP391 文献标识码 A

Chinese Character Generation Model for Cloud Information Security

ZHANG Li¹ LI Qing-sheng² LIU Quan³

(School of Administration, Hubei University of Economics, Wuhan 430205, China)¹

(School of Electronics and Information, Zhejiang University of Media and Communications, Hangzhou 310018, China)²

(School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China)³

Abstract This paper presented a Chinese generation model for cloud information security. By defining the effective Chinese character strokes output method and the Chinese character structure dynamic generation scheme, the model, which includes the structure and the style of Chinese character, can be used for the information security and protection. Compared with the Chinese character coding system, the description system makes it easier to store the Chinese characters into the Web and monitor output in the client. It overcomes the shortcomings that caused by lacking of service information security in entire encoding of modern Chinese characters. It provides an effective strategy and method for cloud storage and cloud data security services, and at same time, it also provides a deeper cloud Chinese character information service basis for the system of information in the cloud.

Keywords Chinese character, Strokes, Font description, Font generation, Information security

1 引言

汉字信息处理技术研究经过几十年的发展,已经开始从宏观走向微观、从通用走向个性化、从规模处理走向精细处理阶段,在这个变化过程中,越来越多的用户对安全、无泄漏、无风险的汉字信息的需求日益增强。研究更加安全、方便的信息保护策略越来越受到人们的重视。

对电子文档信息的防盗窃保护是信息安全领域研究的一项重要内容。传统的信息存储方式是以结构化文档的方式(如数据库)进行存储,与此相关的信息保护策略是基于结构化数据(数据库)的认证或准入机制的信息保护策略,这一策略的最大缺点是一旦认证机制失效,数据库中的所有信息都会出现最大的安全隐患。近年来,研究学者一方面从攻、防、守的信息保护的角度进行研究分析,另一方面也加强了对于基于信息文档内容的信息保护技术的研究。

纵观目前的数字文档保护技术,基于文档内容的保护技

术概括起来主要分为两类,一类是基于静态可识别标志的保护,另一类是基于动态可识别标志的保护。

1.1 基于静态可识别标志的文档保护

静态可识别标志是固定不变的标记。基于静态可识别标志的文档保护技术主要通过文档内部表示层或外部表示层增加可以识别的固定标志来完成。目前主要有数字签名技术(Digital Signature Technology, DST)和数字水印技术(Digital Watermarking Technology, DWT)^[1-4]。DST 是一种经典的数字文档保护技术,它是通过在文档的表示层中增加可以识别的数字签名来实现的;DWT 是一种基于版权的文档保护技术,由于数字水印技术是在文档内容的表示层增加了可以分离和识别的标志,因而也称为基于内容的版权保护。由于是固定不变的静态标记,不论是 DST 还是 DWT,仅适用于通用的静态文档。这些保护技术保护的静态文本对于具有动态结构变换和风格变换的云端数字文档来说无能为力。因此,面对快速发展的互联网络,迫切需要研究一种具有多级、安全、

本文受国家自然科学基金资助项目(60973051),河南省科技攻关资助项目(132102210115),河南省科技基础前沿研究资助项目(152300410089)资助。

张 莉(1973—),女,博士,副教授,主要研究方向为信息安全服务管理;栗青生(1966—),男,博士,教授,主要研究方向为智能信息处理、汉字计算;刘 泉(1963—),女,教授,博士生导师,主要研究方向为数字信号处理。

高效且灵活的信息保护机制以实现基于内容的结构化文档的安全保障机制。

1.2 基于动态可识别标志的文档信息保护

在云环境下,信息文档不再是一种静态内容和单一版本,而是具有动态性、时空立体性、多用户性、多安全等级、多媒体性与多版本性的“活”性文档。如何保护这种“活”结构化文档的机密性和完整性,实现对其安全访问是云计算服务质量控制的关键。

为了更好地体现云端数字化文档的安全服务性能,参考文献[5]提出了一种基于云端信息保护的汉字计算模型,从信息描述的角度,提出了一种基于内容的文档信息保护方法。该方法通过定义比汉字笔画更小的描述单位——笔元,来描述汉字,将编码汉字库抽象转换为非编码汉字信息描述库,克服了目前汉字字库只能在客户端以字库文件存储和服务的不足,实现了基于笔元描述的多级别汉字存储和云端存储功能。用户只需要将现有的编码文档进行一次编码转换,转换后的文档就具有了安全保护功能,既可以进行在线阅读,也可以进行离线存储,一旦检测出异常操作,或者用户的许可超过了阅读期限和使用次数,文档便会像纸片粉碎机一样将汉字信息进行粉碎,从而起到信息保护的目的。

2 编码汉字信息文档和非编码汉字信息文档

编码汉字信息文档是标准的文本文档,它是一种静态的文本,不论是在客户端还是在服务器端,这一文档都具有相同的编码属性,对这类文档的保护只能通过增加或隐藏一些可分离和识别的标签,使用静态可识别标志来实现。非编码汉字文档是一种非编码的动态文本,与标准的基于编码汉字的文档相比,非编码汉字文档更适合网络存储和网络保护,因此是一种更安全的文档。这两种文档的主要区别在于,前者是基于汉字的编码机制,后者是基于汉字的描述机制。

2.1 汉字的编码机制和汉字的描述机制

与编码文档的汉字编码机制相比,非编码汉字文档的字形描述机制的字形安全服务性能明显优越,主要表现在以下几个方面:

(1)存储方式。汉字描述机制使用汉字动态描述库来描述汉字信息,动态汉字字形描述库中汉字是非确定编码汉字,可以使用分布式存储和计算,并且可以广泛地存储在云端服务器中,为用户提供安全的个性化的字形服务。而汉字字库中的汉字是编码汉字,只能存储在本地客户端,仅适合字库文件计算。

(2)使用方式。非确定编码汉字可以在客户端根据用户的需求动态显示信息结果,因此本身就具有信息安全的保护功能。而字库在客户端显示信息的前提是客户端必须安装相应的字库文件系统,由于编码字库是开放的,因此汉字编码机制本身没有信息安全保护功能。

(3)信息保护方式。目前通用的网络信息保护方式采用的是认证和准入机制,这种信息保护方式的缺点是一旦这一机制失效,信息保护手段就面临失败的危险。而使用动态汉字字形描述库可以实现在认证和准入机制失效的情况下,利用云服务端的多级汉字特征描述库,通过服务器端的动态控制,实现基于内容的信息保护。

(4)信息保护的级别。目前,通用的文档信息安全保障机制是在标准编码下的多级机制,级数由算法来确定,而基于内容的信息保障机制可以根据在描述层的分级来增加保护的级

别。例如,可以将汉字信息的描述在汉字字形描述库的基础上再分别进行词或句层的描述,这样数据信息的安全保障则可由 n 级数据安全保障机制来完成, n 的数量由描述层的深度确定,即被描述文字在笔画层、字层、词层或句子层的结构数量。假设我们目前使用的信息加密算法对数据进行了 k 级数据保护,则使用基于内容的数据信息保护就可以达到 $k \times n$ 级。

2.2 汉字信息的笔元描述

笔元描述是我们在甲骨文数字化过程中提出的一种有效的多字种描述方案^[6-10]。简单地讲,笔元就是有方向的笔段,它保留了笔画和笔段描述汉字的特点,加入了汉字书写过程中行笔方向性属性,并且将笔画或笔段之间的连接点以界点、驻点或势点进行描述,使描述过程更加简洁,更适合汉字(包括各类字体)的字形计算。十几年前,互联网的出现给自然语言信息处理技术提供了增加标注的空间,让人们不用修改原有的编码,仅通过添加标注就可以得到汉字的部分语义,给互联网上汉语的适度理解创造了条件。今天,面对即将到来的下一代互联网,网络存储、云端存储、大数据服务等等一系列新技术的应用,使汉字的信息表示拥有了多元描述的条件,具备了多元描述的空间,因此,汉字信息的笔元描述可以变得更加适用和方便。

(1) 笔元

定义 1 笔元是汉字笔画的特征点之间、特征点与驻点之间或驻点与驻点之间的一个有方向的线段。

设 $T(v_i, v_e)$ 是汉字笔画的两个特征点, $v_{z1}, v_{z2}, \dots, v_{zn}$ 是其中 n 个驻点,则汉字的笔元可以表示为:

$$Y(v_i, v_e)$$

如果用笔元 Y 来表示汉字的笔画,则有:

$$T(v_i, v_e) = Y(v_{z1}, v_{z2}) + Y(v_{z2}, v_{z3}) + \dots + Y(v_{zn}, v_{zj})$$

其中, $i, j \in \{1, 2, 3, \dots, n\}$ 。

由以上可以看出,汉字的笔元集合是特征点集合的一个子集。这个集合在汉字字形的空间变换中起着重要的作用。

事实上,每一个汉字笔画中的驻点是随机的或不确定的,我们称之为动态的。因此,用笔元描述的汉字是动态汉字。

(2) 基本笔元和组合笔元

汉字的基本笔元是和组成汉字的基本笔画相对应的笔元,其只有两个特征点,如表 1 所列,它的表达式可以表示为:

$$S_{12} = (x_1, y_1, x_2, y_2), n = 2$$

除了基本笔元以外,其它的笔画或部件的组合称为组合笔元,如表 2 所列,其表达式可以表示为:

$$CPie = Pie_1 + Pie_2 + \dots + Pie_i$$

表 1 特征点和基本笔元

No	Stroke Elements	Feature Point	Graph	Vector Graph
1	Hen(H)	$v_1 v_2$		
2	Shu(V)	$v_1 v_2$		
3	Pie(R_L)	$v_1 v_2$		
4	Na(L_R)	$v_1 v_2$		
5	Dian(H)	$v_1 v_2$		

表 2 组合笔元

No	Stroke Elements	Feature Point	Feature Expression
1	CHen(H)	$v_1 v_2 \dots v_i$	$CHen = Hen_1 + Hen_2 + \dots + Hen_i$
2	CShu(V)	$v_1 v_2 \dots v_i$	$CShu = Shu_1 + Shu_2 + \dots + Shu_i$
3	CPie(R _L)	$v_1 v_2 \dots v_i$	$CPie = Pie_1 + Pie_2 + \dots + Pie_i$
4	CNa(L _R)	$v_1 v_2 \dots v_i$	$CNa = Na_1 + Na_2 + \dots + Na_i$
5	Cdian	$v_1 v_2 \dots v_i$	$CDian = Dian_1 + Dian_2 + \dots + Dian_i$

(3) 扩展笔元

扩展笔元是由一个或多个基本笔元和组合笔元组成的笔元,它是更接近汉字部件的笔元。例如,汉字中的提笔画可以由撇笔画的表达式扩展表示为:

$$Ti = -Pie$$

同样,汉字的左钩和右钩可以表示为:

$$RGou = -Pie$$

$$LGou = -Na$$

因此,扩展笔元的表达式可以表示为如表 3 所列的特征表达式。

表 3 扩展笔元

No	Stroke Elements	Feature Point	Feature Expression
1	Ti (-Pie)	$v_1 v_2$	$Ti = -Pie$
2	RGou (-Na)	$v_1 v_2$	$Gou = -Pie$
3	LGou (-Na)	$v_1 v_2$	$Gou = -Na$

从字形描述的角度来分析汉字特征,汉字笔画特征可以分为字形特征和语义特征,字形特征是汉字外观表征,语义特征是汉字内容的表征。汉字特征点的描述中,在汉字笔画的特征点确定的情况下,增加或减少汉字的描述中的驻点数量,或者增加或改变汉字的笔元数量,并不改变汉字或笔画所包含的语义信息。

3 汉字字形的动态输出

3.1 汉字结构的输出

汉字是一种比较特殊的字符,一般是由基本笔画按照一定的空间关系和构字规范构成偏旁部首,再进一步由偏旁部首根据一定的间架结构布局构成汉字字符,因此,汉字是一种具有层次结构的字符,层结构是汉字的一个基本特征,每一个汉字字符都可以表示成为树状结构。叶节点表示构成字符的每一个笔画(Stroke);叶节点的上一层非叶节点表示构成该字符的偏旁部首(Radical),也就是笔画与笔画之间的空间关系;再往上层的非叶节点表示的是字符部件之间的间架结构关系(Component Layout),这部分可能有多个层次来表示部件之间的嵌套关系,而根节点所表示的结构就可以代表这个字符本身。

比较简单的字符的结构化层次表示如图 1 所示,相对复杂的字符的结构化表示如图 2 所示。

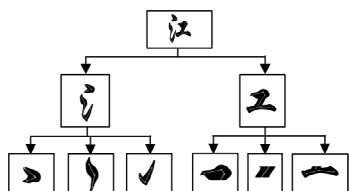


图 1 简单字符的结构化层次表示

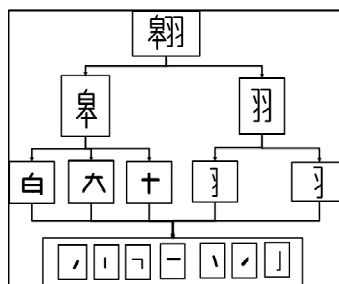


图 2 复杂字符的结构化层次表示

从图中可以看出,无论是简单汉字还是复杂汉字,汉字均可以表示成结构形式,并且将结构分解之后,所有的汉字都源于有限的几个笔画结构,这是本文进行汉字结构描述的基础,也是我们引入笔元的重要依据。

汉字的风格是汉字形成过程中表现出来的个体特色和特征的概括,结构相同的汉字可以有不同的风格。汉字风格主要通过笔画的风格、偏旁和部首的风格来体现。

即使是同一字体的标准字形,同一类型的笔画在不同的汉字中可以有不同的风格表现,特别是对于标准的印刷字体,这一现象非常普遍。如图 3 所示,同属于一种笔画“点”在不同的汉字中或者不同的偏旁部首,都存在一些明显或细微的变化,这些变化正是用来进行安全识别的一个重要手段。



图 3 点笔画在不同汉字中的不同风格

传统的文档中,汉字的风格源于不同的字体设计,但在需要数字化保护的文档中,不同的风格还可以用于汉字字形的安全识别。文献[6]已经研究了基于笔画的汉字计算机自动生成技术。信息安全字形的生成就是在基于汉字字形结构的输出过程中,通过不同的笔画生成指令来完成笔画的计算机自动生成,从而形成新的汉字字形。

3.2 编码转换算法

编码转换是将需要保护的标准编码的文档转换为非编码的保护文档,转换后的文档存储在云端服务器上,不同用户调用和访问该文档的过程就是不同的文字结构和风格动态输出的过程,编码转换算法如下。

步骤 1 变量的初始化。

这一步骤主要对文档名称 doc_name、内容 doc_string、类型 D、原始数据变量 F、原始数据文档汉字数量 char_num1、目标数据变量 M、目标数据字形数量 char_num2、字形描述库 char_discription_data(i)和标准字进库 sdander_data 进行初始化操作。

Open doc_name//打开文档

String doc_string;

String D;

String F,M;

int char_num1,char_num2;

int char_num,stroke_num;

步骤 2 判断文档 doc_string 编码类型。若文档编码类型为“编码汉字”,则转步骤 2.1;若文档编码类型为“笔画汉字”,则转步骤 2.2;若文档编码类型为单个笔画和部件,则转步骤 2.3;否则则转向步骤 3。

步骤 2.1 读取当前字符,并将其放入原始数据变量 F。同时将文

档汉字 char_num1 数值减 1。

步骤 2.1.1 在描述库中检索并找到对应的描述汉字(汉字结构描述),并放入目标数据变量 M 中;并将汉字计数变量 char_num2 置 1;文档位置 pos 自加 1。

步骤 2.1.2 判断 char_num1, 如果 char_num1 != 0, 则转步骤 2.1, 否则转步骤 3。

步骤 2.2 读取一个字形, 字形总数 char_num = char_num - 1; 在描述库中检索有没有该描述字形, 如果有, 则转步骤 3。

步骤 2.2.1 将 doc_string 加入描述库, 判断 char_num 是否为 0, 如果是转步骤 3, 否则转步骤 2.2。

步骤 2.3 读取一个笔画或部件的描述, 笔画总数 strock_num = stroc_num - 1; 在描述库中检索有没有该描述字形, 如果有, 则转步骤 3。

步骤 2.3.1 将 doc_string 加入描述库, 判断 strock_num 是否为 0, 如果是转步骤 3, 否则转步骤 2.3。

步骤 3 结束。

通过编码转换算法, 标准的编码文档生成了可以自动生成特殊字形的保护文档, 进而在客户端进行结构的转换和笔画的生成, 形成具有个性化的特殊字形文档, 这些特殊字形像数字水印一样在任何需要的时候提取出来, 如果试图销毁这些“水印”, 文档就会被“粉碎”。图 4 为文档输入和输出示结构意图。

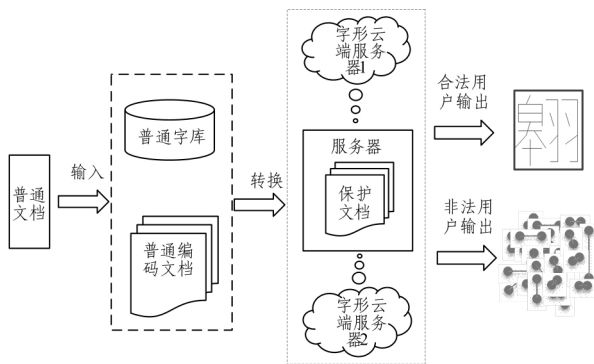


图 4 文档输入和输出示结构意图

编码转换后的文档可以直接进行汉字的结构输出, 如果在结构输出的同时使用文献[6]所述的笔画生成算法, 还可以对输出风格进行控制。详细内容请参考文献[6,7]。

4 云端信息保护

信息安全字形是通过在客户端动态生成不同保护级别的字形文档的方法来实现, 主要的实现方法有以下几种。

4.1 文字结构保护方法

(1) 结构水印保护

特征水印保护方法是在文档生成过程中对指定文档或用户使用不同的汉字字形来表示的方法。具体步骤如下:

步骤 1 将需要保护的标准文档进行编码转换。

步骤 2 将转换后的文档中的每一个汉字字形结构按保护需要进行调整和变形。

步骤 3 按照变形后的结构, 输出需要保护的文档。

在按照以上 3 个步骤输出的文档中, 文档中的每一个汉字的结构的变换都是一种可以检测的特殊水印。

以 Bezier 曲线变换为例来说明其工作原理。

在如下形式的 Bezier 曲线表达式中:

$$C(t) = \sum_{k=0}^n P_k B_{k,n}(t), t \in [0, 1], K=0, 1, \dots, n$$

$$B_{k,n}(t) = C_n^k t^k (1-t)^{n-k} = \frac{n!}{k!(n-k)!} t^k (1-t)^{n-k}$$

$$t \in [0, 1], k=0, 1, \dots, n$$

以 3 次贝塞尔曲线进行结构拟合。

设 3 次贝塞尔曲线的 4 个控制点为 $cp[i]$, 其中, $i=0, 1, 2, 3$; 取 $cp[0] = (x_1, y_1)$ 为笔元的始点, 其它 3 个点 $cp[1], cp[2], cp[3]$ 的选取方法为: 以选取向量 (v_1, v_2) 方向的正向为基线, 以点 v_2 为原点逆时针旋转 90° , 即 $cp[1] = a$ 处 (a 的值可以设计为整数值) 为第二个点, 其它两个控制点设定方法与此类似。在标准字形结构上通过在拐点处的拟合, 即可改变拐点形状。图 5(b) 所示是用标准的汉字部件(图 5(a))在驻点处用 Bezier 曲线拟合后的不同输出。

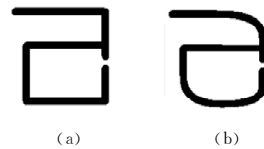


图 5 相同结构在驻点处的不同拟合效果

(2) 结构变形保护

结构变形保护是使用汉字结构的动态描述属性而进行的保护。虽然汉字的结构固定, 但一定规范内的结构改变不会影响汉字的使用, 正如日常每个人的书写笔迹不同, 但也能正常交流一样, 适当修改一些汉字的组成结构, 如笔画的相对位置, 部件的位置和大小比例等, 可以像指纹一样使数字文档得到保护。

图 6 所示是对标准字形中每个笔元尾结点垂直方向微调后的输出。

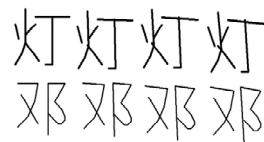


图 6 结构变形保护

4.2 笔画变形水印保护

文字结构水印保护方法仅做了结构上的变形, 如果保持汉字结构不变, 也可以通过修改笔画风格来实现笔画变形水印的保护。具体方法如下:

步骤 1 将需要保护的标准文档进行编码转换。

步骤 2 保持转换后的文档中的每一个汉字字形结构不变。

步骤 3 按照文献[6]中的笔画生成算法, 输出需要保护的文档。

例如: 可以使用有个性特征的笔画实现特殊的数字水印保护, 如图 7 所示。

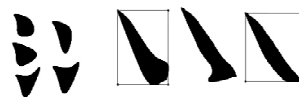


图 7 点笔画和捺笔画的变形

4.3 隐藏笔元保护法

由于描述笔画的笔元中间可以加入若干个驻点, 因此可以在各个笔元的驻点之间或始点与结点之间加入一些隐藏信

息,在必要时可以将隐藏的信息提取出来,具体方法是:

步骤1 按照本文第2节的方法,建立一个隐藏了笔元的汉字字形描述库。

步骤2 将需要保护的标准文档进行编码转换,即基于步骤1建立的字形描述库进行编码转换。

步骤2 保持转换后的文档中的每一个字形结构不变。

步骤3 按照本文4.1节和4.2节的笔画生成算法,输出需要保护的文档。

由于汉字结构复杂,隐藏笔元的形式和方式可以很多,不仅可以在笔元中隐藏公司的图标,还可以在字中藏字,一旦遭到攻击,隐藏的文本可以迅速走到前台,对文档内容进行保护。

图8所示为在笔元SE中隐蔽另一个笔元HSE的原理示意图。

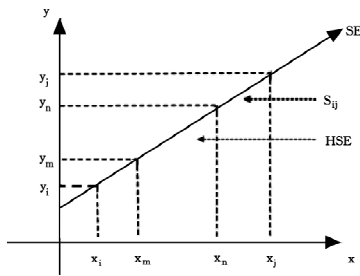


图8 笔元的隐藏

5 实验结果与分析

5.1 具有结构水印保护的字形输出

以任意汉字为例,结合文献[6]说明生成方法的有效性。

原始字形结构输出:使用已经建立的常用字形的描述库中的任意两个字形进行直接结构输出,可以得到如图9(a)所示的字形结构。

Bezier曲线进行拟合变换后输出:在字形描述库的基础上使用Bezier曲线进行拟合变换,可以得到对应的具有结构保护水印的字形输出,如图9(b)所示。更多的笔画和部件保护输出如图9(c)所示。从图中可以看出,不同部件或笔画结构上的变换都可以实现。

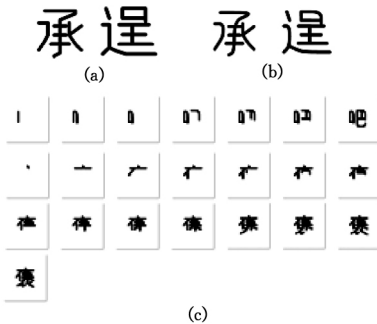


图9 结构笔画水印输出示意图

5.2 具有风格水印保护的字形输出

设两个特征点分别为 $v_1(153, 189)$ 和 $v_2(353, 389)$, 按照文献[6]构造 d_4 和 d_5 两个向量控制向量(或权向量), 其中 d_4 为 v_1v_2 正方向上的一控制向量, 其模长 $|d_4| = 30$, d_5 为 v_1v_2 负方向上的一个控制向量, 模长 $|d_5| = 20$, 保持其它权向量的默认值不变, 可以得到如图10所示的初始点图形。

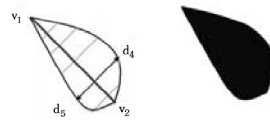


图10 初始点图形

从图11中可以看出,同一笔画经过不同的变换可以得到风格不同的笔画水印,对应到相应汉字就可以形成不同风格的汉字水印。

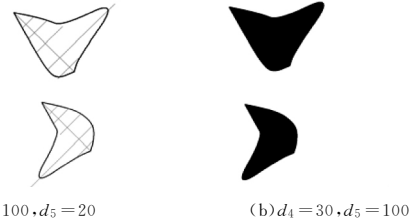


图11 风格水印输出

结束语 本文将汉字字形生成与信息安全字形生成相结合,提出了一种与信息安全相关的汉字字形生成模型,并对生成实验和效果进行分析,得到了具有指导意义的结论。与现有的数字水印和数字签名技术相比,本文提出的信息保护字形具有实时性和动态性,以及更好的网络存储和网络控制属性,达到了对云端信息进行加密和保护的目的。它为中文信息文档的安全保护和版权保护提供了一种新的思路。将描述文本和编码文本有效结合起来设计未来的汉字系统是一个值得深入研究的课题。下一步的研究目标将在此基础上继续深入实验,为未来云端汉字系统的研究奠定基础。

参考文献

- [1] Mandolkar V M. RSE for Electronic Text document Protection [C] // 2010 2nd International Conference on Computer Engineering and Technology, 2010;30-43
- [2] Meng Ying-jie, Si Lei, Shi Yao. A Chinese Texts Zero-Watermarking Algorithm Based on Vector Graphics [J]. Journal of Computer Research and Development, 2009, 46(supp): 22-26
- [3] Meng Ying-jie, Guo Tao, Guo Zhi-hua, et al. Chinese Text Zero-watermark Based on Sentence's Entropy [C] // Proceedings of the International Conference on Multimedia Technology (ICMT 2010). IEEE Computer Society, 2010; 864-867
- [4] Yamada T. Use of both invisible and emergable watermarks to deter illegal copying of images [C] // 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems KES 2013. 2013; 401-410
- [5] 栗青生, 张莉, 熊晶. 基于云端信息保护的汉字计算模型 [J]. 计算机科学, 2015, 42(11): 73-79
- [6] 栗青生, 熊晶, 吴琴霞, 等. 基于特征加权的汉字点笔画生成研究 [J]. 北京大学学报(自然科学版), 2014, 50(1): 153-160
- [7] 吴琴霞, 栗青生, 高峰. 基于语义构件的甲骨文字库自动生成技术研究 [J]. 北京大学学报(自然科学版), 2014, 50(1): 161-166
- [8] 吴琴霞, 栗青生. 基于动态描述库的汉字字形自动生成技术研究 [J]. 科学技术与工程, 2013, 13(15): 4425-4431
- [9] 栗青生, 吴琴霞, 杨玉星. 甲骨文字形动态描述库及其字形生成技术研究 [J]. 北京大学学报(自然科学版), 2013, 49(1): 61-67
- [10] 栗青生, 吴琴霞, 王蕾. 基于甲骨文字形动态描述库的甲骨文输入方法 [J]. 中文信息学报, 2012(4): 45-63