二值图像中的数据隐藏算法

郭 萌, 张鸿宾, 魏 磊
(北京工业大学计算机学院, 北京 100124)

摘 要: 本文证明了在一个 m@n 的二值图像块中至多改变 1 个像素时所能隐藏的比特数的上界为 õ(log2(mn+ 1)8, 并实际构造了达到此上界的信息隐藏算法. 文中归纳了一些选择修改像素位置的规则, 较好地保持了数据嵌入后图像的质量. 对算法的鲁棒性和安全性进行了分析. 用 3 个例子) 数字手写签名中认证信息的嵌入、二值电子文档的窜改检测和卡通图像中注释信息的嵌入说明了本文算法可能的应用.

关键词: 数据隐藏; 数字水印; 图像认证; 二值图像
中图分类号: TP391 文献标识码: A 文章编号: 03722112 (2009) 1122409207

Data Hiding in Binary Images

GUO Meng, ZHANG Hong2bin, WEI Lei
(Computer College, Beijing University of Technology, Beijing 100124)

Abstract: This paper proposes a method to embed data in binary images with high embedding capacity. It is proved that, given an m @n image block of the host image, the upper bound of the amount of bits that can be embedded in that block is õ(log2(mn + 1) 8by changing at most 1 pixel in the block. An algorithm whose embedding capacity can reach the bound is then constructed. We also summarize some rules for selecting pixels to be changed in order to maintain higher quality of the host image after data hiding. The robustness and security of the proposed algorithm is analyzed. Three examples, data hiding in digital handwritten signature, temper detection of digital binary documents, and invisible annotation for cartoon image are tested to illustrate the potential applications.

Key words: data hiding; digital watermarking; image authentication; binary image

1 引言

近年来, 随着数字媒体应用的普及, 数字媒体内容的保密性和安全性, 它的知识产权保护、真实性和完整性的认证等问题已成为人们关注的焦点, 推动了信息隐藏和数字水印技术的发展.

目前图像中信息隐藏的研究大多是针对彩色和灰度图像的. 由于灰度和彩色图像的像素值通常较大, 小幅度地改变少量像素的值不会产生引人注意的变化[1~6]. 然而, 在二值图像中嵌入信息则要困难得多. 把黑(白)像素改为白(黑)像素(特别是非边界上的像素)很容易引起人们的注意. 目前已经有大量的金融票据、保险、专利、法庭和社会公安等文档(包括文本、图形和图片等)进行了数字化, 以二值电子文档的形式保存. 另外, 在电子政务和电子商务中, 还经常遇到数字化的手写签名. 如何认证这些重要的电子文档和手写签名, 检测和定位可能的窜改就成为一个十分迫切的问题. 数字水印技术(单独或与密码学相结合)则为二值图像的认证提供了一种有效的手段.

在二值图像水印的研究中, 文献[7~9]的工作都是针对特殊二值图像的. 对于一般的二值图像, Koch 和 Zhao 提出了一种改变图像块中黑白像素的比例(大于或小于 1)来嵌入水印的方法[10]. 这种算法很难处理那些黑(或白)像素有很高或很低比例的情况, 安全性和容量都不够好. Wu[11] 和 Tseng[12] 提出的方法是目前一般二值图像中数据隐藏的两种有代表性的方法. 根据图像连通性和平滑性的要求, Wu 定义了像素的可修改分值, 根据分值的大小, 优先修改那些不影响连通性和平滑性的像素. Wu 的方法在一个图像块中只嵌入一位信息, 因而隐藏容量有限而且和具体的图像有关. Tseng 的方法与 Wu 不同, 它允许在二值图像的任何位置上进行黑白像素的修改. 该方法的优点是嵌入容量大, 在一个 m@n 的图像块中, 若允许至多可以改变两个像素, 则最多可以嵌入õlog2(mn+ 1) 8位水印. 然而, 正如下面要

收稿日期: 2008210213; 修回日期: 2009209201
基金项目: 国家自然科学基金(No. 60775011); 北京市教育委员会科技发展计划项目

讨论的, Tseng 的方法并没有达到嵌入容量的上界.

本文提出一种二值图像中(包括扫描后的文本、图形、手写签名, 地图和卡通画等) 大容量的数据隐藏算法. 该算法把一幅图像分成若干块, 然后在每一块中通过至多改变一个像素来嵌入数据, 数据的提取不需要原图像. 要讨论的问题是, 当至多改变一个像素时能嵌入的数据的上限是多少, 如何设计这样的算法, 以及如何选择修改像素的位置, 保证嵌入信息后的图像质量.

2 最多改变一个像素时的隐藏容量分析

图像中的信息隐藏是通过改变原图像一些位置上的像素值来实现的. 数据隐藏容量是指在一幅图像中能够隐藏多少比特的数据. 希望隐藏容量越大越好. 从本文后面的分析可以看出, 如果原二值图像的像素无限多, 那么只改变一个像素可以嵌入任意多的数据. 但是, 实际图像的像素数是有限的, 而为了不降低嵌入后图像的质量, 原图像中能够改变的像素也有一定的限制, 因此只能嵌入一定量的数据. 在目前发表的文献中, 大部分的算法都是在一个图像块中改变一个或几个像素去嵌入一位二进制数据, 这远没有达到图像块的最大嵌入容量. 文献[12] 提出了一种增大数据隐藏容量的算法, 在一个 m @n 的图像块中至多改变 2 位像素时所能嵌入的比特数 r 最多为õlog$_2$(mn+ 1)8. 但是, 文[12] 的算法是否已经达到容量的上限了呢? 在一个给定大小的图像块中, 至多改变一个像素时最多可以嵌入多少位的数据? 换个角度提出这个问题就是, 当要求至多改变 1 位像素而嵌入 r 位数据时, 图像块的最小尺寸是多少? 这个问题解决了, 我们就能清楚一幅图像的数据隐藏容量有多大并加以充分利用. 这一节我们先讨论这个问题.

不失一般性, 假设一幅二值图像分成了若干个 m @n 的子块, 在每个子块中至多修改一个像素来嵌入二进制数据. 下面的定理给出了数据嵌入容量的上界.

定理 1 在一个 m@n 的二值图像块中, 若至多改变 1 个像素去嵌入 r 位二进制数据, 则应满足: m @n \ 2^r- 1.

证明 图像块 m @n 个像素的取值状态可以记为 $s_1 s_2$, s_{mn}, s_i I {0, 1}, i= 1, 2, , , mn, 要隐藏的二进制数据串是 $d_1 d_2$, d_r, d_i I {0, 1}, i= 1, 2, , , r. 改变 1 个像素后图像块的状态集合记为 P = { $s_1 s_2$, sc_k, $s_{mn-1} s_{mn}$| k= 1, 2, , mn}, 其中 s^c_k 表示将第 k 个像素 s_k 翻转后的取值, 显然| P | = mn. 因此, 至多改变 1 个像素时图像块共有 m @n+ 1 种取值状态(包括不改变任何像素) . 隐藏的信息是通过图像块的取值状态来确定的. 由于任意一个 r 位二进制数至少应对应一个图像块的取值状态, 因此, 至多改变 1 个像素去嵌入 r 位二进制数据时, 应该满足 m @n+ 1\ 2^r.

定理 1 给出了至多改变一个像素嵌入 r 位二进制数据时图像块尺寸的下界, 它同时也是在一个 m @n 大小的图像块中至多改变一个像素时能够嵌入的二进制数据位数r 的上界. 例如, 一个 4 @4 的图像块, 根据定理, 至多改变一个像素时最多可以嵌入不超过 4 位的二进制数据.

3 最多改变一个像素时的数据隐藏算法

本节讨论能够实现定理 1 的最大隐藏容量的算法.

311 算法的思想

目前大多数的二值图像数据隐藏算法是在一个图像块中改变一个或几个像素来嵌入 1 位二进制数据. 为了能尽量多地嵌入数据, 我们可以将图像块中的像素分组, 让块中每一组像素表示要嵌入的一位二进制数.

假设要在一个 3 @3 的图像块中嵌入 3 比特数据. 我们可以把 9 个像素分成 3 组, 每个像素可以同时属于不同的组. 令每一组像素代表要嵌入的一位二进制数. 如果一组中黑像素的个数是奇数, 这一组就表示隐藏了一个/ 10; 如果是偶数, 就表示隐藏了一个/ 00. 问题是如何分组, 使得无论要嵌入任何 3 位二进制数, 都能保证至多只改变一个像素即可.

假设已经将这 9 个像素分成了 3 组, 初始状态下这 3 组像素所表示的二进制数分别是/ 00、/ 00、/ 00. 如果要嵌入的数据恰好是/ 0000, 那么不需要改变任何像素. 如果要嵌入的数据是/ 0010, 这表明第 1 组和第 2 组像素表示的二进制数与要嵌入的一致, 而第 3 组像素表示的数据/ 00 与要嵌入的/ 10不一致. 因此需要把第 3 组像素所表示的二进制数改为/ 10. 这可以通过把第 3 组中一个像素的值取非来实现. 这意味着第 3 组中必须至少有 1 个像素只属于第 3 组, 而不在其他的两组中. 这样, 将该像素值取非后, 就能保证第 3 组表示的值变为/ 10, 而不影响第 1, 2 组. 假如要嵌入的数据是/ 1100, 这时第 1 和第 2 组表示的二进制数与要嵌入的都不同, 只有第 3 组是一致的. 因此需要同时改变第 1、2 组表示的二进制数. 由于只能修改一个像素, 这时就需要有一个像素, 既属于第 1 组, 也属于第 2 组, 但不属于第 3 组. 把这个像素的值取非后, 第 1、2 组代表的二进制数就会同时改变, 又不会影响第 3 组.

图 1 显示了当要嵌入 3 位二进制数据时所有需要改变像素值的 7 种情况, {#}内的数字表示要修改的像素所在的组号. 当把所有可能的情况都考虑后, 就能得到满足至多修改一个像素嵌入 3 位二进制数据时总共需要的最少像素数. 假设有一个 3 @3 的图像块 F, 我们用下面一个分组矩阵 I 来表示这个图像块中像素的分

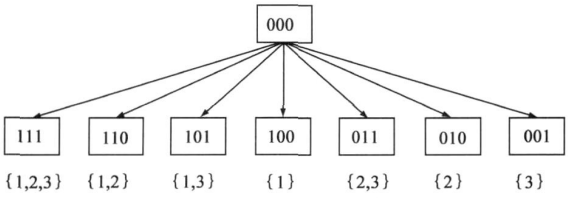


图1　嵌入3位数据时需要改变像素值的7种情况

组信息:

$$I=\begin{pmatrix}\{1\} & \{2\} & \{3\}\\ \{1,2\} & \{1,3\} & \{2,3\}\\ \{1,2,3\} & \{1\} & \{2\}\end{pmatrix}$$

其中, 矩阵 I(1, 1) 位置的元素/ {1}0表示它对应的图像块 F(1, 1) 位置的像素属于第 1 组. I 中位于(3, 1) 的元素/ {1, 2, 3}0表示矩阵 F(3, 1) 位置上的像素同时属于第 1、2、3 组. 由图 1 不难看出, 若要满足至多改变1个像素嵌入3 位二进制数的条件, 则分组矩阵最少需要 7 个元素. 即, 分别需要1 个像素仅属于第1 组、第 2 组、和第 3 组; 分别需要 1 个像素同时属于第 1、2 组、第 2、3 组和第 1、3 组; 还要有 1 个像素同时属于第 1、2、3 组. 所有的分组信息可以用集合 A_3= {{1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}}来表示. 这样, 在一个至少有 7 个像素的图像块中, 就可以实现至多改变一个像素而嵌入 3 位二进制数据.

312　至多改变一个像素时的数据隐藏算法

本小节讨论当至多改变一个像素时, 能够实现上述思想的数据隐藏算法. 首先定义一些符号:

F: 图像矩阵. 不失一般性, 假设 F 的尺寸是 m @n 的整数倍. 将 F 分成若干个 m @n 的图像子块, 其中第 i 块记作 F_i.

K: m @n 的随机矩阵, 作为密钥由嵌入方和检测方共享.

I: m @n 的秘密分组矩阵, 由嵌入方和接受方共享, 其元素要满足一定的条件(将在后面说明).

r: 在每个 m @n 子块中要嵌入的数据的位数, 满足 $2^r-1\leq m\times n$.

下面引入一些定义.

定义 1　集合{1, 2, , , r}的所有非空子集组成的集合称为完备集, 记为记为 A_r. 例如, 当 r= 3 时, A_3= {{1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}}就是一个完备集.

定义 2　一个 m @n 矩阵I 称为分组矩阵, 如果 A_r 中的任一元素都是 I 中的一个元素. I 中剩余的位置可以重复安排 A_r 的任何元素. 例如, 当 r= 3 时, 下面的矩阵就是一个满足要求的分组矩阵:

$$I=\begin{pmatrix}\{1\} & \{2\} & \{3\}\\ \{1,2\} & \{1,3\} & \{2,3\}\\ \{1,2,3\} & \{1\} & \{2\}\end{pmatrix}$$

下面先给出隐藏算法的具体步骤. 算法中的密钥矩阵 K 和分组矩阵I 由嵌入方和检测方共享.

至多改变一个像素时的数据隐藏算法:

(1) 计算 $F_i \copyright K$, © 是逐位异或运算.

(2) 利用分组矩阵 I, 对每个 i= 1, 2, , , r, 求出所有属于第 i 组的元素的集合:

$$S_i=\{(j,k)\mid i\in I_{j,k}\},\quad i=1,2,\ldots,r$$

集合 S_i 表示第 i 组中的所有像素在图像矩阵中的坐标.

(3) 将各组中所有像素的值求和, 然后模 2:

$$Group_i=\left(\sum_{(j,k)\in S_i}[F_i\copyright K]_{j,k}\right)\bmod 2$$

$Group_i$ 表示初始时第i 组像素所表示的二进制数.

(4) 设要嵌入的二进制数据为 $d_1, d_2, , , d_r$, $d_i\in\{0,1\}$, 计算:

$$Index_i=Group_i\copyright d_i,\quad i=1,2,\ldots,r$$

$Index_i$= 0 表示原图像块中的第 i 组像素所表示的二进制数与要嵌入的的第 i 位二进制数据一致, $Index_i$= 1 表示不一致.

(5) 计算与要嵌入的数据位不一致的像素组的集合: S= {k| $Index_k$= 1}, 此时有两种情况:

(a) 若 S= 5, 表明原图像块中各组所表示的二进制位都与要嵌入的数据一致, 不需要修改像素点.

(b) 若 S X 5, 例如 S= {1, 3, 5}, 这说明原图像块中第 1、3、5 组所表示的二进制位与要嵌入的不一致. 在分组矩阵 I 中找到含{1, 3, 5}元素的位置(i, j), 然后把它对应的矩阵中相应位置上$[F_i]_{j,k}$的像素值取非.

这样就完成了数据的嵌入过程. 当检测方收到嵌入数据后的图像时, 可以按以下步骤提取数据:

(　)将图像按与嵌入时相同的块大小把图像分为 m @n 的子块, 第 i 个图像块记为 F_i^c.

(　)对每个图像块, 计算 $F_i^c\copyright K$.

(　)利用分组矩阵 I, 求出每个数据位对应的像素的集合:

$$S_i=\{(j,k)\mid i\in I_{j,k}\},\quad i=1,2,\ldots,r$$

(　)计算各数据位:

$$Group_i=\left(\sum_{(j,k)\in S_i}[F_i^c\copyright K]_{j,k}\right)\bmod 2.$$

把各数据位级联起来, 可得 $Group_1$, $Group_r$= d_1d_2, d_r, 即得被嵌入的 r 位二进制数.

从上面的算法可以看出, 上述算法已经达到了至多改变一个像素而嵌入 r 位数据的最大嵌入容量. 完备集 A_r 的基数为 2^r-1, 而分组矩阵的元素个数只要不少于它就可以. 实际上原图像分块的大小是由分组矩阵的大小决定的, 并且尺寸相同. 如果取 2^r-1 为图像分块矩阵的大小, 那么根据定理 1, 这种算法就达到了

最大的嵌入容量.

可以看出, 这种算法比起目前大多数的改变一个或几个像素而嵌入1 位数据的算法, 在数据嵌入容量上有了很大的提高.

4 图像的质量控制

上一节算法的优点是最大限度地利用了图像的嵌入容量. 但是, 一旦分组矩阵确定之后, 对于给定的嵌入数据, 要修改的像素位置也就确定了, 没有可选择的余地, 因此有可能会修改比较醒目位置上的像素而造成图像质量的下降. 一种可能的解决方法就是牺牲一些嵌入容量, 增大图像块的尺寸, 或者在每小块中少嵌入一些数据. 例如, 把图像分成 5 @5 的块, 上一节算法每块最多可以嵌入4 比特数据, 现在为了改善图像的质量, 只在每个图像块中嵌入 3 位数据. 这时完备集为 $A_3=\{\{1\},\{2\},\{3\},\{1,2\},\{1,3\},\{2,3\},\{1,2,3\}\}$. 一种可能的分组矩阵为:

$$I=\begin{bmatrix} \{1\} & \{2\} & \{3\} & \{1,2\} & \{1,3\} \\ \{2,3\} & \{1,2,3\} & \{1\} & \{2\} & \{3\} \\ \{1,2\} & \{1,3\} & \{2,3\} & \{1,2,3\} & \{1\} \\ \{2\} & \{3\} & \{1,2\} & \{1,3\} & \{2,3\} \\ \{1,2,3\} & \{1\} & \{2\} & \{3\} & \{1,2\} \end{bmatrix}$$

这样, 为了同一目的而可以修改的像素就有多个, 可以挑选对图像质量影响较小的一个像素进行修改. 例如, 假定我们要修改分组矩阵 I 中元素{1, 2}所对应的像素的值, 这时 I 中共有 4 个{1, 2}元素, 把这 4 个位置对应的任何一个像素取非后的效果都是一样的, 因此可以选择对图像质量影响最不显著的一个像素进行修改.

目前关于二值图像的视觉感知模型的研究还很少. 下面我们归纳一些选择修改像素的基本原则, 以减小图像质量的降低.

(1) 全黑和全白的图像块内的像素不能作修改.

(2) 要修改的像素应尽量在图像的边界上, 并尽量保持边界的平滑性.

(3) 一个像素的 8 连通区域内的像素值若都和该像素相同, 则尽量不修改该像素.

(4) 尽量不选修改后改变图像连通区域个数的像素.

(5) 若只能选择一个像素修改, 并且该像素修改后图像块变为全黑或全白, 则该图像块不嵌入数据, 并将该图像块变为全黑或全白.

从上面这些原则可以看出, 在选择要修改的像素时, 不仅和这个像素有关, 而且和该像素周围邻域像素的黑白模式有关. 在后面的实验中, 我们采用了这 5 种规则, 取得了较好的效果.

5 算法的鲁棒性和安全性

这一节, 分析算法的鲁棒性和安全性方面的问题.

511 算法的安全性

本文提出的算法具有很好的安全性. 设原图像分为 m @n 的图像块, 则密钥矩阵 K 和分组矩阵也均为 m @n. 假定采用至多改变 1 位像素嵌入 r 位数据的算法. 因为 K 是一个任意矩阵, 因此共有 2^{mn}种选择方法. 根据分组矩阵的定义, 满足条件的不同分组矩阵的选择方法共有:

$$\binom{mn}{2^r-1}\#\left(2^r-1\right)!\ \#\left(2^r-1\right)^{mn-(2^r-1)}$$

因此, 即使攻击者知道了算法的内容, 也很难用穷举法破译密钥矩阵和分组矩阵. 例如当分块大小为 10 @10, 在每块中嵌入 4 位数据时, 密钥 K 共有 2^{100}种, 分组矩阵共有$\binom{100}{15}\#(15)!\ \#(15)^{100-15}$种, 而且这个数目随着分块的增大会成指数级增长.

512 算法的鲁棒性

应当指出, 本文讨论的数据隐藏算法主要用于认证和注释等目的. 这种应用场合要求的是脆弱(fragile) 水印或半脆弱(semi2fragile) 水印. 尽管要求数据嵌入后能够抵抗适度的图像畸变和打印扫描等处理, 但是对抵抗去除水印等攻击的要求并不高. 因为在图像认证的应用中, 对手的主要兴趣是伪造合法的水印, 而不是消除水印.

考虑在每个图像块中只改变一个像素嵌入 r 位二进制数据时的情况. 假设所用的分组矩阵的大小正好是所需要的最小尺寸, 即共有 2^r-1 个元素. 因此, 回想前面第二节中 I 的定义. 每个分组所包含的像素点的个数为:

$$1+\binom{r-1}{1}+\binom{r-1}{2}+\cdots+\binom{r-1}{r-1}=2^{r-1}$$

设像素块中每个像素受噪声影响而改变的概率为 p, 且相互独立. 则像素块中的第 i 个分组所代表的数据发生错误的概率为:

$$P_i=\sum_{\substack{k=1\\ k\ is\ odd}}^{2^{r-1}}\binom{2^r-1}{k}p^k(1-p)^{2^r-1-k}=\frac{1-(1-2p)^{2^{r-1}}}{2}$$

可见, 当 p 和 2^{r-1}的值越小时, P_i 的值也越小. 但是当 p 的值很大时, 错误率就会增大. 当 p= 015 时, P_i 也等于 015. 这时就难保证检测方能够正确地提取出数据. 由于本文算法的嵌入容量较大, 可以考虑采用纠错码或多个图像块冗余嵌入相同数据的方法.

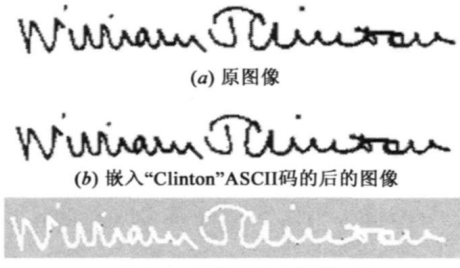
对于图像的几何畸变以及打印/ 扫描时的失真, 可以采用加几何标记等方法. 限于篇幅, 这里不再详述.

6　可能的应用及实验结果

本节给出本文算法一些可能的应用及实验结果.

611　数字手写签名中的信息隐藏

目前数字手写签名正在逐渐代替传统的以实物为载体的签名形式. 然而, 数字手写签名的应用却面临着未授权使用的严峻挑战. 为了解决这一问题, 人们提出了在数字手写签名中隐藏认证信息, 以此作为签名的认证工具. 图 2 是一个在数字手写签名中嵌入签名者信息的示例. 原图像的尺寸为 287@61, 分块尺寸为 40@40, 每块嵌入 7 位, 共嵌入了 49 位二进制数据. 可以看出, 数据嵌入后基本上是不可感知的. 对于这幅图像, Wu 的方法共修改了 30 个像素[11], 而本文方法只修改了 7 个像素.



(a) 原图像

(b) 嵌入"Clinton"ASCII码的后的图像

(c) 黑点表示原图和嵌入图的不同

图2　数字手写签名中的数据隐藏

The proposed scheme uses the weight matrix W to represent the embedded data. This section presents an illustrative example to demonstrate how to manage weights. Section II-B presents the complete scheme.

Assume that the size of K and W is 3×3. Below, we consider a 3×3 image block F_i, which is a part of the host image F. The purpose is to show how to embed $r = 2$ bits of data in F_i. Let us assume the following inputs:

(a) 原图像

The proposed scheme uses the weight matrix W to represent the embedded data. This section presents an illustrative example to demonstrate how to manage weights. Section II-B presents the complete scheme.

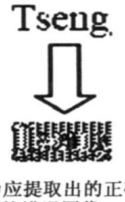
Assume that the size of K and W is 5×5. Below, we consider a 3×3 image block F_i, which is a part of the host image F. The purpose is to show how to embed $r = 1$ bits of data in F_i. Let us assume the following inputs:

(c) 将(b)窜改后的图像

The proposed scheme uses the weight matrix W to represent the embedded data. This section presents an illustrative example to demonstrate how to manage weights. Section II-B presents the complete scheme.

Assume that the size of K and W is 3×3. Below, we consider a 3×3 image block F_i, which is a part of the host image F. The purpose is to show how to embed $r = 2$ bits of data in F_i. Let us assume the following inputs:

(b) 嵌入作者姓名图像"Tseng" 后的图像



(d) 上图为应提取出的正确图像,下图为实际提取出的错误图像

图3　电子文档的认证及窜改检测

612　二值文档图像的认证以及篡改检测

本文提出的数据隐藏方法可以单独或与其它的加密认证方法相结合, 用于二值电子文档的认证及窜改检测. 基本思想是, 认证数据以一种脆弱的方式嵌入到文档中, 如果图像被窜改或者不再具有图像的某些性质, 那么嵌入的信息将被破坏. 隐藏的数据可以是某种容易识别的标志或者是和文档内容相关的摘要等信息. 图 3 为英文二值文档图像篡改检测的示例. 原图像的尺寸为 483 @180, 分块尺寸为 15 @15, 每块嵌入 4 位, 共嵌入了 1288 位二进制数据, 其中姓名/ Tseng0 的图像数据占有 1144 位.

613　二值卡通图像中注释信息的隐藏

对于艺术作品, 创作者通常希望注上作品创作的日期、地点和版权等信息, 并且希望注释信息尽可能地在视觉上不影响原作. 本文提出的数据隐藏方法可以用于二值卡通图像中的这类注释. 图 4 是在一幅卡通图

(a) 原图像

(b) 嵌入70位日期信息后

(c) 黑点表示原图和嵌入图的不同

图4　二值线画图像中不可见的注释

片中嵌入了 70 位日期信息的示例. 原图像的尺寸为 183@192, 分块尺寸为 45@45, 每块嵌入 5 位数据. 可以看出嵌入日期信息后几乎在视觉上没有影响原作.

614 与其它算法的性能比较

下面以两幅中、英文的二值文本图像(图 5, 6)作为测试图像, 将本文算法与文献[12]的算法 TCP 进行比较.TCP 算法允许在一个 m@n 的图像块中, 至多改变两个像素来嵌入$\lfloor \log_2(mn+1) \rfloor$位信息. 表 1 是算法比较的结果. 其中图像的失真程度采用距离倒数的失真度量(DRDM)[13], 与峰值信噪比(PSNR)和均方误差(MSE)的度量相比, DRDM 方法是一种比较有效的二值图像失真的评价标准.

在图 5 和图 6 中, (a)图为原始图像, (b)、(d)、(f)为本文算法在分块尺寸为 8@8, 每块内分别嵌入 2、3、4 位数据后局部放大后的图像, (c)、(e)、(g)为 TCP 算法在同样的分块尺寸 8@8, 每块内同样分别嵌入 2、3、4 位数据后局部放大后的图像. 实验中采用了同样的质量控制原则. 对于 8@8 的图像块, 本文算法与TCP 算法所能嵌入的比特数上界均为 6, 为了有更好的图像质量, 实验中牺牲了一些嵌入量. 从实验结果中可以看出, 在相同的嵌入容量下, 本文算法的修改像素数要少, 因而比 TCP 算法有更好的图像质量.

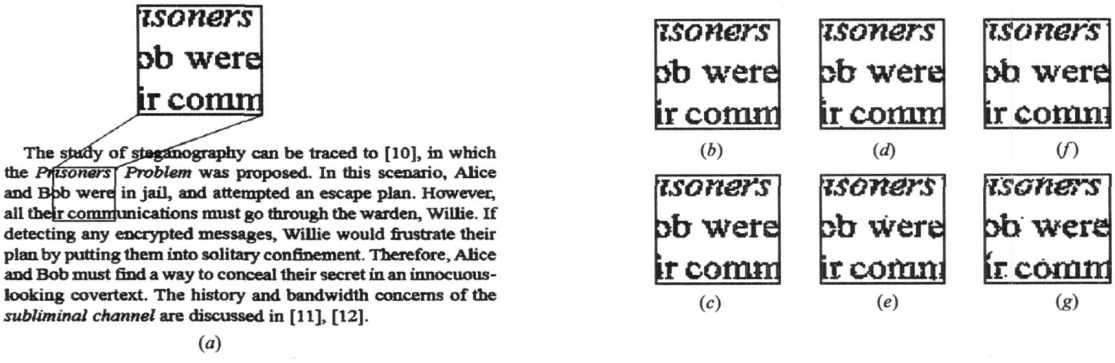


图5 英文文档中嵌入数据结果

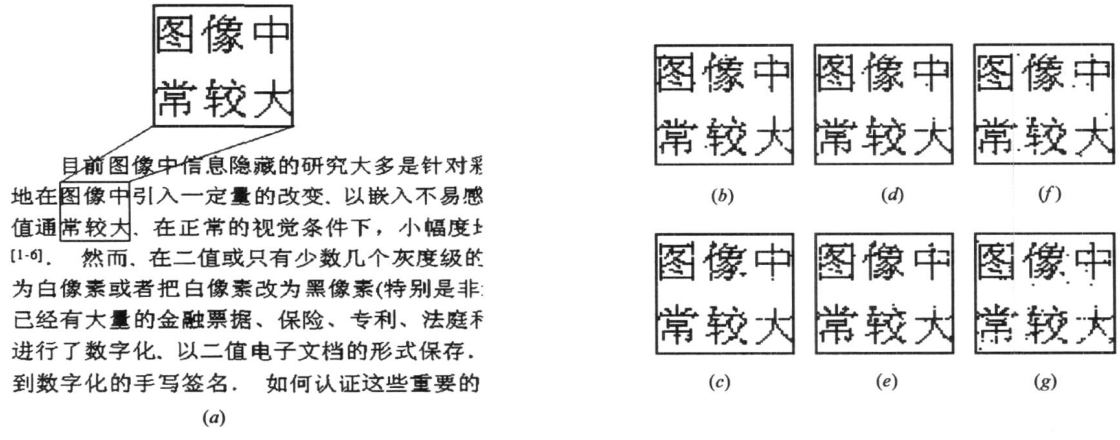


图6 中文文档中嵌入数据结果

表 1 与 TCP 算法比较结果(分块大小均为 8@8)

	每块嵌入 2 比特		每块嵌入 3 比特		每块嵌入 4 比特	
	本文算法	TCP 算法	本文算法	TCP 算法	本文算法	TCP 算法
图 5 修改像素个数	549	724	594	938	618	1088
图 6 修改像素个数	551	772	644	1025	717	1182
图 5 对应的 DRD 值	0.437	0.563	0.502	0.787	0.540	0.952
图 6 对应的 DRD 值	0.436	0.605	0.514	0.819	0.585	0.985

7 结论

本文提出了一种二值图像中的数据隐藏算法. 证明了在 1 个 m@n 的图像块中至多改变 1 位像素时所能嵌入的二进制位的上界, 并且构造了达到此上界的算法. 为了尽可能减少数据嵌入对图像质量的影响, 论文提出了一些选择修改像素位置的原则. 与其它算法的实验比较说明, 本文算法具有容量大和图像质量较好的优点.

参考文献:

[1] I J Cox, M L Miller. The first 50 years of electronic watermarking[J]. Applied Signal Processing, 2002, 56(2): 126-132.

[2] F A P Petitcolas, R J Anderson, M G Kuhn. Information hiding-a survey[J]. Proceedings of the IEEE, 1999, 87(7): 1062-1078.

[3] F Hartung, M Kutter. Multimedia watermarking techniques. proceedings of the IEEE[J]. 1999, 87(7): 1079-1107.

[4] I J Cox, M L Miller, J A Bloom, J Fridrich, T Kalker. Digital Watermarking and Steganography (Second Edition)[M]. San Mateo, CA: Morgan Kaufmann, 2008. 1-13.

[5] M Yeung, F Mintzer. Invisible watermarking for image verification[J]. J. of Electronic Imaging, 1998, 7(3): 578-591.

[6] I Cox, J Kilian, T Leighton, T Shamoon. Secure spread spectrum watermarking for multimedia[J]. IEEE Trans. Image Processing, 1997, 6(12): 1673-1687.

[7] K Matsui, K Tanaka. Video-steganography: how to secretly embed a signature in a picture[J]. Proc. IMA Intellectual Property Project, 1994, 1(1): 187-206.

[8] M Fu, O Au. Data hiding by smart pair toggling for halftone images[A]. Proc. IEEE Inter. Conf. On Acoustics, Speech, and Signal Processing[C]. Istanbul, Turkey, 2000. 2318-2321.

[9] N F Maxemchuk, S Low. Marking text document[A]. Proc. IEEE ICIP. 97[C]. Santa Barbara, CA, USA, 1997. 13-13.

[10] E Koch, J Zhao. Embedding robust labels into images for copyright protection[A]. Proc. Inter. Congr. Intellectual Property Rights for Specialized Information, Knowledge and New Technologies[C]. Hamburg, Germany, 1995. 242-251.

[11] M Wu, B Liu. Data hiding in binary image for authentication and annotation[J]. IEEE Trans. Multimedia, 2004, 6(4): 528-538.

[12] Y-C Tseng, Y-Y Chen, H-K Pan. A secure data hiding scheme for two-color images[J]. IEEE Trans. Communications, 2002, 50(8): 1227-1231.

[13] Haiping Lu, Alex C Kot, Yun Q Shi. Distance-reciprocal distortion measure for binary document images[J]. IEEE Signal Processing Letters, 2004, 11(2): 228-231.

作者简介:

郭　萌　男, 1983 年 10 月生于北京. 北京工业大学博士生, 研究方向是模式识别等.

E-mail: guome@126.com

张鸿宾　男, 教授, 博士生导师. 目前主要的研究工作是模式识别、图像处理与分析、计算机图形学、数字水印和数据隐藏等.

E-mail: zhb@public.bta.net.cn

魏　磊　男, 2006 年北京工业大学硕士退学赴美攻读学位.