

一种具有纠错能力的半脆弱水印算法

陈惠明

(忻州师范学院 计算机系, 山西 忻州 034000)

摘 要:随着数字技术和互联网的飞速发展,图像数据的安全保护问题日益严重,而数字水印技术已成为一种保护数据安全的有效途径。文中提出了一种基于汉明码的用于图像版权保护的半脆弱水印盲算法。首先对二值水印信号进行汉明编码并调制,然后将调制后的水印号采用自适应分块算法嵌入到原始图像中,最后利用了汉明码的纠错算法进行了水印的提取。算法较好地解决了水印的不可见性和鲁棒性之间的矛盾。实验结果表明,该算法对 JPEG 有损压缩和高斯低通滤波等攻击具有一定的容忍性。

关键词:半脆弱水印;图像认证;汉明码;鲁棒性;JPEG 攻击

中图分类号:TP391

文献标识码:A

文章编号:1673-629X(2012)04-0242-04

A Semi-Fragile Watermarking Algorithm with Ability to Correct Mistakes

CHEN Hui-ming

(Department of Computer Science, Xinzhou Teachers University, Xinzhou 034000, China)

Abstract: Along with the rapid development of digital technology and internet, image data safety protection is getting more and more serious and digital watermarking technology has become an effective way to protect the safety of the data. It proposes a blind semi-fragile watermarking algorithm based on Hamming code, which is used in image copyright protection. First of all, it transforms binary watermark signal to Hamming code and modulates it, then embeds the modulated watermarking into the original image. This process uses adaptive partition algorithm. At last, it adopts the correcting algorithm of Hamming code to extract the watermark, which solves the contradiction between watermark invisibility and robustness felicitously. Experimental results show that this algorithm has a bit of toleration to attacks, such as JPEG lossy compression and Gaussian low-pass filter.

Key words: semi-fragile watermarking; authentication of image; Hamming code; robustness; JPEG attack

0 引言

数字水印技术是一种新的信息隐藏技术,其基本思想是在数字图像、音频和视频等产品中嵌入秘密的信息以便保护数字产品的版权。

一般用于数据认证的数字水印又可以分为两类:脆弱水印、半脆弱水印^[1]。脆弱水印是一种在数字图像作品发生任何形式的改变时都无法检测出来的水印;半脆弱水印一般指能承受图像进行诸如 JPEG 压缩、加少量噪声的偶然修改,但会被图像内容的恶意篡改损坏的水印。

半脆弱数字水印算法根据水印的加载域可分为空间域和变换域两类。空间域方法是在空间域中将水印与原图相结合,通过改变空间域的某些像素的灰度值,

达到水印嵌入和隐藏的目的。变换域方法是通过改变变换域的一些系数的值来嵌入水印,主要变换包括 DFT、DCT、DWT^[2]等。

文中利用汉明码的纠错能力在空间域提出了一种具有纠错能力的半脆弱水印算法,并通过实验分析了该算法对 JPEG 有损压缩和高斯低通滤波等攻击的容忍性。

1 汉明码纠错

汉明码是 1950 年由 Hamming 首先构造的,它是一种能够自动检测并纠正一重错的线性纠错码。

设数据位数为 m , 校验位数为 k , 则总编码位数 $n = m + k$, 有 Hamming 不等式:

$$2^k - 1 \geq n, 2^k \geq m + k + 1 \quad (1)$$

对于这个不等式可以理解为:由于 n 位码长中有一位出错,可能产生 n 个不正确的代码(错误位也可能发生在校验位),所以加上 k 位校验后,就需要定位 m

收稿日期:2011-08-20;修回日期:2011-11-23

基金项目:山西省自然科学基金项目(2009011018-4)

作者简介:陈惠明(1968-),男,山西忻州人,硕士,讲师,研究方向为图像识别。

+ k 个状态。用 $2k$ 个状态中的一个状态指出“有无错”,其余 $2k-1$ 个状态便可用于错误的定位。要能充分地进行错误定位,则须满足式(1)的关系。由 Hamming 不等式得到校验位数与可校验的最大信息位之间的关系如表 1 所列。

表 1 Hamming 可校验的最大信息位数

校验位数 k	可校验最大信息位数	编码总位数
1	0	1
2	1	3
3	4	7
4	11	15
5	26	31
6	57	63
7	120	127

编码的最小汉明码距与编码纠错检错能力关系:

① 要发现(检测) e 个随机错误,则要求码的最小距离 $d_{\min} \geq e + 1$;

② 要纠正 e 个随机错误,则要求码的最小距离 $d_{\min} \geq 2e + 1$;

③ 要纠正 e 个随机错误,同时检测 f 个错误,则要求码的最小距离 $d_{\min} \geq e + f + 1$ 。

可以看出,检验位的长度越长,合法码字所占的比例就越小,如果这些码字能够尽可能地在所有的码字中均匀分布,合法码字间的最小汉明码距就越大,编码的抗干扰能力也就越强。因此设计编码方法的最重要的任务就是尽量使合法码字尽可能地均匀分布。

(7,4) 分组码是一种常用的汉明码,其最小码距 $d = 3$,它能纠 1 个错或检 2 个错。下面以 (7,4) 分组码为例介绍汉明码的纠错过程:

设 $A = [a_6, a_5, a_4, a_3, a_2, a_1, a_0]$ 为 (7,4) 分组码,其中前 4 位是信息位,后 3 位是校验位。引入 S_1, S_2, S_3 三个校正子,其中

$$\begin{cases} S_1 = a_6 \oplus a_5 \oplus a_4 \oplus a_2 \\ S_2 = a_5 \oplus a_4 \oplus a_3 \oplus a_1 \\ S_3 = a_6 \oplus a_5 \oplus a_4 \oplus a_0 \end{cases} \quad (2)$$

通过校正子 S_1, S_2, S_3 组成的码序列可以准确纠正信息串中的 1 位错误,其错码位置如表 2 所列。

表 2 (7,4) 分组码的纠错位置

S_1	S_2	S_3	错码位置
0	0	1	a_0
0	1	0	a_1
0	1	1	a_2
1	0	0	a_3
1	0	1	a_4
1	1	0	a_5
1	1	1	a_6
0	0	0	无错误

2 具有纠错能力的半脆弱水印算法

考虑到人类视觉系统对纹理平滑区域图像的改变敏感性较强,对纹理较复杂区域图像的改变敏感性较弱这一特点,文中采用了一种自适应算法,其主要思想是首先对图像进行分块处理^[3],然后根据图像块灰度的方差和能量将块分成 8 种类型,对纹理较平滑的块给予较小的水印嵌入强度,对纹理较复杂的块给予较大的水印嵌入强度,较好地解决了不可见性和鲁棒性^[4,5]之间的矛盾。

下面以 (7,4) 分组码为例介绍具有纠错能力的半脆弱水印的嵌入和提取过程。

2.1 水印嵌入

(1) 设 $W_{36 \times 4}$ 为二值水印,首先对水印 $W_{36 \times 4}$ 的每行按式(2)规则进行汉明编码,生成矩阵 $X_{36 \times 7}$, X 的第 m 行可表示为:

$$X = [W[m, :], s_1, s_2, s_3]$$

其中 $1 \leq m \leq 36$ 。

(2) 对 X 进行扩展,得到一个由 $(-1, 1)$ 组成的扩展序列 Y 。

设 cr 为扩展因子,其扩展过程如下:

$cr = 256$;

for $i = 1:252$

if $X(i) == 1$

$Y(i, 1:cr) = 1$;

else $Y(i, 1:cr) = -1$;

end;

end;

$Y(253:256, :) = 0$;

(3) 产生一高斯正态分布序列 $G_{256 \times 256}$,再由 G 生成一个由 $(-1, 1)$ 组成的伪随机序列 $P_{256 \times 256}$ 。利用 P 对扩展序列 Y 进行调制,得到含水印信息的序列 $Yp_{256 \times 256}$ 。

$$Yp = Y * P;$$

至此完成了对水印的加密和预处理工作^[6,7]。

(4) 读入将要嵌入水印的载体图像 $I_{256 \times 256}$,并对其进行分块处理,块的大小 $n = 8 \times 8$,分别以 A_k 表示, k 为总块数。计算每块的方差 std 和能量 E 。

$$A_k = \begin{bmatrix} a_{k1} & a_{k2} & \cdots & a_{k8} \\ a_{k9} & \ddots & \ddots & a_{k16} \\ \cdots & \ddots & \ddots & \cdots \\ a_{k57} & a_{k58} & \cdots & a_{k64} \end{bmatrix}$$

$$std(A_k) = \left[\frac{1}{n-1} \sum_{i=1}^n (a_{ki} - \bar{a}_k)^2 \right]^{\frac{1}{2}} \quad (3)$$

$$\text{其中, } \bar{a}_k = \frac{1}{n} \sum_{i=1}^n a_{ki}$$

$$E(A_k) = \sum_{i=1}^n a_{ki}^2 \quad (4)$$

(5) 按能量 E 升序对 I 中的块进行排序, 得到 I 的变换矩阵 $B_{64 \times 1024}$, 其中 B 一列代表一个块。

$$B[:, k] = \begin{bmatrix} a_{k1} \\ a_{k2} \\ \dots \\ a_{k64} \end{bmatrix}, \text{ 其中 } 1 \leq k \leq 1024$$

(6) 利用 E 和 std 对块进行分类, 确定块的水印嵌入强度。

① 确定 E 的分界阈值 $e1 < e2 < e3$, 将 B 中的块分成 4 类;

② 对上述 4 类块中的每一类以类内方差中值为界线继续划分, 共得到 8 种类型的图像块, 分别用 $R1, R2, \dots, R8$ 表示。其算法描述如下, 其中 $m1 \sim m4$ 分别为四个方差中值; q 为水印嵌入强度系数; $t1, t2, \dots, t8$ 分别代表 8 种类型块的基准噪声阈值^[8], 其值需通过实验确定。

```
for i = 1:k
    if E(i) <= e1
        if std(i) >= m1
            t(i) = t1; //Ak属于 R1
        else t(i) = t2; //Ak属于 R2
        end;
    else if (e1 < E(i) & E(i) <= e2)
        if std(i) >= m2
            t(i) = t3; //Ak属于 R3
        elseif(i) = t4; //Ak属于 R4
        end;
    else if (e2 < E(i) & E(i) <= e3)
        if std(i) >= m3
            t(i) = t5; //Ak属于 R5
        elseif(i) = t6; //Ak属于 R6
        end;
    else if (e3 < E(i))
        if std(i) >= m4
            t(i) = t7; //Ak属于 R7
        else t(i) = t8; //Ak属于 R8
        end;
    end;
end;
```

③ 生成附加噪声 $temp$, 得到每个块的嵌入强度系数 $q = \text{基础噪声} + \text{附加噪声}$;

```
deta = 0.0035;
temp = deta * B;
for i = 1:k
```

```
q(:, i) = t(i) + temp(:, i);
```

```
end;
```

(7) 嵌入经调制后的水印 Y_p , 并重构含水印的图像^[9,10]。

```
TempB = reshape(B, 256, 256);
```

```
Tempq = reshape(q, 256, 256);
```

```
// 嵌入水印
```

```
WaterMark = TempB + Tempq * yp;
```

```
// 重构嵌入水印的图像 Out
```

```
TpW = reshape(WaterMark, 64, 1024);
```

```
Out = col2im(TpW, [8, 8], [256, 256], 'distinct');
```

2.2 水印的提取

(1) 从嵌入水印的图像 Out 中提取 (7, 4) 分组码;

```
for i = 1:252
```

```
sk(i) = sum((Tempq(i, :) - TempB(i, :)).
```

```
* p(i, :));
```

```
end;
```

```
// 根据 sk(i) 的符号提取分组码 rp
```

```
for i = 1:252
```

```
if sign(sk(i)) == -1
```

```
rp(i) = 0;
```

```
else rp(i) = 1;
```

```
end;
```

```
end;
```

```
// 将 rp 转换成 (7, 4) 分组码的表示形式
```

```
Temprp = reshape(rp, 36, 7)
```

(2) 对提取到的分组码按表 2 的要求进行纠错, 最后将分组码的前 4 列提取得到一个 36×4 的水印 $W1$ 。

(3) 计算原始水印 W 和提取到的水印 $W1$ 的相关性, 若相关性大于某阈值, 则水印提取成功, 否则水印图像可能受到攻击。

3 实验与分析

使用 256×256 的 Cameraman 灰度图像作为实验图像。实验分两步进行, 首先进行不可见性测试, 验证算法的有效性; 然后再进行 JPEG 有损压缩和高斯低通滤波攻击测试, 验证算法的鲁棒性。

(1) 随机生成二值水印 W , 如图 1(a) 所示; 然后对其采用 (7, 4) 分组码加密扩展; 采用自适应分块算法确定块的嵌入强度, 其中基础噪声取值为: $t1 = 1.7$ 、 $t2 = 2.1$ 、 $t3 = 2.5$ 、 $t4 = 2.9$ 、 $t5 = 3.3$ 、 $t6 = 3.7$ 、 $t7 = 4.1$ 、 $t8 = 4.5$; 将调制后的水印嵌入原始图像, 最后使用水印提取算法提取水印 $W1$, 实验结果如图 1 所示。用 PSRN 值作为衡量水印对载体图像质量的影响^[11,12], 从图 1(c) 中可以看出水印的不可见性很好, 与原图相

比没有视觉上的差异, $PSNR = 31.2909\text{dB}$; 使用相关性衡量水印 W 和 $W1$ 的相似程度, 结果其相关性为 1, 说明 W 和 $W1$ 完全相同。



图1 水印嵌入试验

(2) JPEG 有损压缩攻击测试。将嵌入水印的图像按不同的压缩率存储成 JPEG 格式, 然后使用文中的水印提取算法提取水印, 实验结果如表 3 所列。

表3 不同压缩率的 JPEG 攻击测试

压缩率	45%	40%	35%	30%
PSNR	25.2192	24.8424	24.4728	24.0283
相关性	0.9722	0.9306	0.8473	0.7368

从表 3 可以看出随着压缩率的降低, 图像损失的有效信息逐渐增加, 其中包括水印信息, 当压缩率低于 35% 时提取到的水印失真已经变得较大。事实上当图像压缩率接近 30% 时, 通过肉眼已经能够看出图像的失真, 判别图像的真伪已经变得很简单了。因此, 文中的算法对识别 JPEG 攻击是有效的。

(3) 高斯低通滤波攻击测试。对嵌入水印的图像进行高斯低通滤波(分别取不同的标准差 δ), 然后使用文中的水印提取算法提取水印, 实验结果如表 4 所列, 可以看出当 $\delta < 1$ 时文中的算法对高斯低通滤波攻击有较高的容忍度。

表4 不同滤波系数下的高斯低通滤波攻击测试

δ	0.6	0.7	0.8	1
PSNR	24.5998	22.9305	21.9635	20.9616
相关性	1	0.9726	0.9181	0.7233

4 结束语

文中提出一种具有纠错能力的数字水印算法。算法首先使用汉明码对二值水印进行编码和加密扩展, 然后使用自适应算法在分块图像中嵌入水印, 最后使用汉明码的纠错算法提取水印。实验表明文中提出的算法均衡了水印的不可见性和鲁棒性之间的矛盾, 对 JPEG 有损压缩和高斯低通滤波等攻击具有较好的容忍度。

对文中提出的算法可以做如下进一步的研究:

(1) 将算法由空域扩展到变换域;

(2) 使用其它一些效率更高的纠错码, 进一步提高算法的纠错效率, 增加算法的鲁棒性。

参考文献:

- [1] 李东勤, 林克正. 基于混沌映射的半脆弱图像水印算法[J]. 计算机技术与发展, 2008, 18(11): 156-159.
- [2] 李京兵, 黄席褀. 一种基于 DWT 抗几何攻击数字水印鲁棒算法[J]. 计算机仿真, 2007, 24(3): 303-307.
- [3] 张建伟, 鲍政王, 王顺风. 图像小波域分块奇异值分解的自适应水印算法[J]. 中国图象图形学报, 2007, 12(5): 811-818.
- [4] Sun Y, Wang F, Zhou Y, et al. Digital Watermarking Techniques Based on Double Chaos System[J]. Microelectronics & Computer, 2005, 22(8): 114-116.
- [5] 陈东方, 张有清. 基于提升方案小波和混沌映射的盲水印算法[J]. 计算机工程与设计, 2008, 29(20): 5372-5375.
- [6] Kwak D W, Joo L S, Won K J, et al. An Efficient, LKH Tree, Balancing Algorithm for Group Key Management[J]. IEEE Communication, 2006, 10(3): 222-224.
- [7] 王化丰, 张桂香, 邵勇. 基于 Logistic 映射的混沌流密码设计[J]. 计算机工程, 2007, 33(10): 164-168.
- [8] 孔国杰, 张培林, 曹建军, 等. 基于提升小波变换的信号降噪及其工程应用[J]. 计算机工程与应用, 2008, 44(10): 234-237.
- [9] 朱佳婷, 吕建平. 抗旋转攻击的整数小波变换数字水印算法[J]. 计算机技术与发展, 2007, 17(7): 145-147.
- [10] 朱长青, 符号军. 基于整数小波变换的栅格数字地图数字水印算法[J]. 武汉大学学报·信息科学版, 2009, 34(5): 619-625.
- [11] Pei S C, Zeng Y C. Hiding Multiple Data in Color Image by Histogram Modification[C]//Proceeding of the 17th International Conference on Pattern Recognition. Piscataway: Institute of Electrical and Electronics Engineers Inc, 2004: 799-802.
- [12] 李建华, 李万社. 基于整数提升小波变换的盲数字水印算法[J]. 兰州交通大学学报, 2008, 27(1): 127-130.