第 36 卷 第 2 期 计 算 机 工 程 2010 年 1 月
Vol.36 No.2 Computer Engineering January 2010
·安全技术· 文章编号：1000—3428(2010)02—0132—03 文献标识码：A 中图分类号：TP309

基于 BTC 与秘密共享的图像水印算法

姜明芳[1,2]，朱宁波[1]
(1. 湖南大学计算机与通信学院，长沙 410082；2. 湖南第一师范学院图书馆，长沙 410205)

摘 要：结合块截短编码与秘密共享，提出一种新的鲁棒图像水印算法。该算法利用 BTC 编码由原图像构建特征共享，利用水印图像与特征共享一起生成私有共享。由待验证图像构建的特征共享与私有共享一起恢复水印图像。私有共享的生成与特征共享有关，可实现对同一图像的多水印注册。特征共享的稳健性确保了算法的水印鲁棒性，水印嵌入没有引起图像质量的改变。实验结果表明该算法对一般信号处理攻击有较高的鲁棒性。

关键词：图像水印技术；块截短编码；秘密共享

Image Watermark Algorithm Based on BTC and Secret Sharing

JIANG Ming-fang[1,2], ZHU Ning-bo[1]
(1. School of Computer and Communication, Hunan University, Changsha 410082;
2. Library of Hunan First Normal University, Changsha 410205)

【Abstract】 A novel robust image watermark technology is presented by combining Block Truncation Coding(BTC) and secret sharing. The feature share of the original image constructed by BTC is combined with the watermark to generate the private share. In watermark extraction, the watermark can be retrieved by XOR operation between the private share and the feature share from the suspected image. The private share is dependent of the feature share, and it is possible to register multiple watermarks for a single host image, and the robustness of the feature share ensures the robustness of the watermark algorithm. This algorithm does not alter the host image when the private share is generated. Experimental results show high robustness of the proposed algorithm against common signal processing attacks.

【Key words】 image watermark technology; Block Truncation Coding(BTC); secret sharing

1 概述

自互联网出现以来，数字水印技术作为一种数字媒体版权保护手段得到了广泛关注，引起了 IT 领域众多研究人员的兴趣。数字水印技术是将创作者或版权所有者信息(水印)嵌入原始数字媒体中以保护原始媒体版权的技术，当需要验证版权时就可从含水印图像中提取水印，含水印图像中水印可以是可见或不可见[1-2]的呈现方式。本文研究不可见水印技术。一般，数字水印系统至少应满足鲁棒性、透明性、安全性等基本要素，一个好的数字水印算法应是针对某种应用的数字水印基本要素的综合平衡[3]。当前水印算法可以分为空域水印算法[1]和变换域水印[2]算法两大类，然而这些算法所嵌入的水印容易通过统计方法检测或去除，且大多数方法均会引起原始媒体的改变。文献[4]提出利用视觉密码设计图像水印算法，该方法不会引起原图像质量的下降，但水印鲁棒性不高。文献[5]结合块截短编码(Block Truncation Coding, BTC)方法与视觉秘密共享，提出一种更为鲁棒的水印方案，但其鲁棒性仍有待提高，且提供给每个用户的秘密图像相同，不适用于为不同用户的同一图像嵌入不同水印版本。为进一步增强水印算法的鲁棒性和实现对同一图像的多水印注册，本文提出一种结合块截短编码 BTC 编码和图像秘密共享的鲁棒水印算法。

2 块截短编码

块截短编码是图像处理中常见的图像压缩方法之一，因其简单且编码速度快而受到青睐，并于 1987 年成为 JPEG 压缩标准的候选压缩算法。图 1 给出了绝对矩块截短编码(Absolute Moment Block Truncation Coding, AMBTC)示例。图 1(a)是一个有 16 个像素的原始图像，其均值 $\mu=127.937\ 5$。原始块被编码成图 1(b)所示的位示图，小于 μ 的像素用“0”标示，其对应区块像素均值 $\mu_1=125$；大于均值 μ 的用“1”标示，其对应区块像素均值 $\mu_2=129.7$，大于块均值 μ 的像素个数 q=10。于是由三元组 (B,μ_1,μ_2) 可得图 1(c)所示的重建图像。

124	121	126	128
127	128	129	130
128	128	130	127
138	130	125	128

μ=127.937 5 (a)原图像

0	0	0	1
0	1	1	1
1	1	1	0
1	1	0	1

q=10 (b)位示图

125	125	125	130
125	130	130	130
130	130	130	125
130	130	125	130

μ_1=125.0, μ_2=129.7 (c)重建图像

图 1 AMBTC 示例

本文利用 AMBTC 构建原图像特征块(特征共享)，然后利用水印图像与特征共享一起生成一个私有共享。在接收端利用特征共享与私有共享即可恢复水印图像，以进行版权验证。

3 BTC 与视觉秘密共享相结合的水印方案

本文方案由两部分组成：秘密共享生成(水印编码)和水印提取，分别如图 2 和图 3 所示。

基金项目：湖南省教育厅基金资助项目(08C018)
作者简介：姜明芳(1979－)，女，助理馆员、硕士研究生，主研方向：图书情报，信息安全；朱宁波，副教授、博士
收稿日期：2009-06-26 **E-mail：**mingfangjean@gmail.com

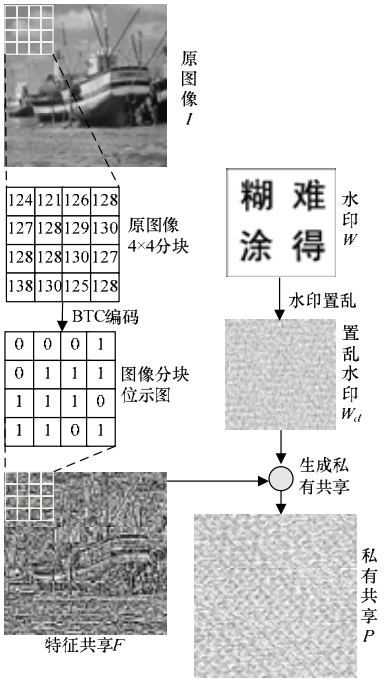


图 2　秘密共享生成过程

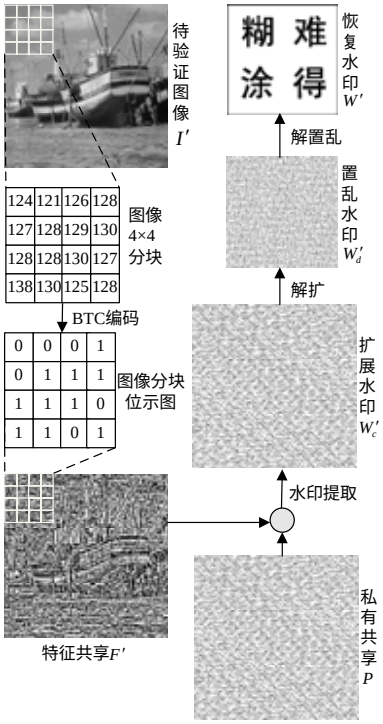


图 3　水印提取过程

算法 1 秘密共享生成算法

输入 大小为 $m\times n$ 的原图像 I，大小为 $m_1\times n_1$ 的水印 W，密钥 key(伪随机数发生器种子)

输出 大小为 $m\times n$ 的秘密共享 P

步骤 1 将原图像 I 分成 4×4 大小的无重叠块，对各分块作 AMBTC 编码得位示图即特征共享 F。

步骤 2 为增强系统鲁棒性和安全性，在水印嵌入前，先由密钥 key 利用伪随机数发生器对水印图像 W 进行置乱，得置乱水印序列

$$W_d=\left\{W_d(i,j)\middle|W_d(i,j)=0,1,0\le i<m_1-1,0\le j<n_1-1\right\}$$

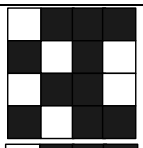
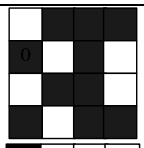
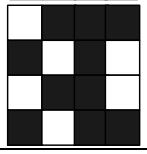
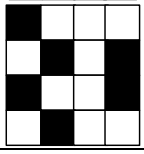
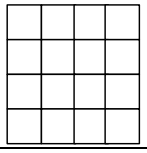
步骤 3 利用图像特征 F 和置乱水印序列 W_d 进行图像秘密共享编码(即生成秘密共享 P)，见表 1，具体编码方案如下：

(1)如果置乱水印像素值 $W_d(i,j)$ 为 0，则私有块 PB(4×4 块)与对应特征块 FB 一样；

(2)如果置乱水印像素值 $W_d(i,j)$ 为 1，则私有块(4×4 块)与对应特征块的补见表 1(⊗表示异或)。

重复步骤 3 直到所有像素值编码完毕，即生成了秘密共享 P。

表 1　图像秘密共享编码(私有块生成)示例

水印 $W_d(i,j)$	特征块 FB	私有块 PB	$FB\otimes PB$
0			
1			

水印提取基本上是秘密共享生成(水印编码)的逆过程，见图 3，算法描述如下：

算法 2 水印提取算法

输入 大小为 $m\times n$ 的待验证图像 I'，秘密共享 P，密钥 key(伪随机数发生器种子)

输出 大小为 $m_1\times n_1$ 的提取水印 W'

步骤 1 按算法 1 的步骤 1 的方法，将待验证图像 I'分成 4×4 大小的无重叠块，对各分块作 AMBTC 编码得位示图即特征共享 F'。

步骤 2 将特征共享 F' 与私有共享 P 进行异或运算得扩展水印序列 W_c'。

步骤 3 由于水印编码阶段中的图像秘密共享编码将一个像素映射为一个大小为 4×4 的图像块，引起了像素扩展，因此在解码阶段(水印提取)应进行相应解扩操作，具体步骤如下：

(1)将扩展水印序列 W_c' 分成大小为 4×4 的无重叠块；

(2)若扩展水印序列 W_c' 中某 4×4 块含白像素(像素值为 1)个数小于 8，则其对应的解扩水印值为 0；

(3)如果 W_c' 中某 4×4 块含白像素个数大于等于 8，则其对应的解扩水印值为 1；

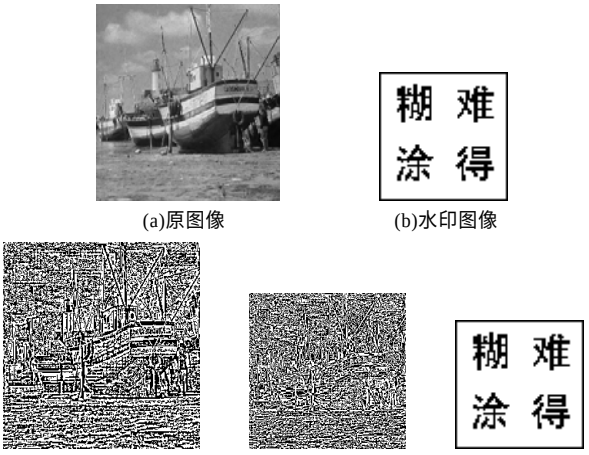
(4)直到 W_c' 中所有块都映射完，即得到解扩后的置乱水印 W_d'。

步骤 4 由密钥 key 利用伪随机数发生器对置乱水印序列 W_d' 进行反置乱得恢复水印序列

$$W'=\left\{W'(i,j)\middle|W'(i,j)=0,1,0\le i<m_1-1,0\le j<n_1-1\right\}$$

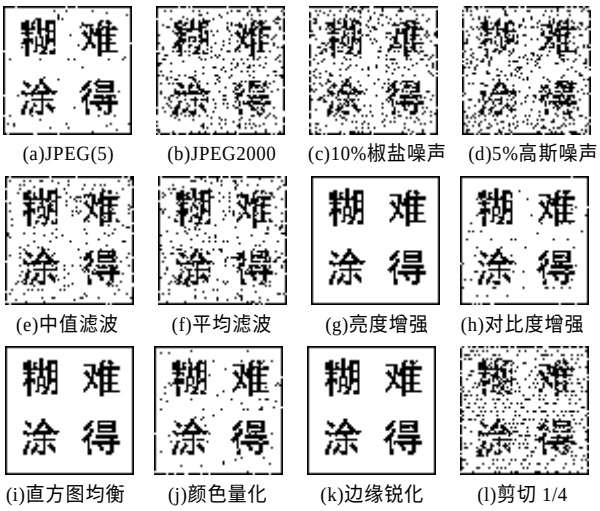
4　实验结果与性能分析

本文算法在 Matlab7.0 环境下得以实现。在水印编码过程中，私有共享的生成与原图像特征和水印相关，因此，依赖于原始图像的特征共享稳健性充分保证了算法的鲁棒性。实际进行的大量实验也验证了该算法的鲁棒性。实验中取原始图像为 256×256 的灰度图像，水印图像为 64×64 的二值图像。图 4 以 boat 图像为例给出了一个水印嵌入示例(密钥为 0.740 212)，图 5 给出了一些常见信号处理攻击下算法的鲁棒性实验结果。



(a)原图像　(b)水印图像　(c)特征共享 *F*　(d)私有共享 *P*　(e)恢复水印

图 4　水印嵌入示例



(a)JPEG(5)　(b)JPEG2000　(c)10%椒盐噪声　(d)5%高斯噪声　(e)中值滤波　(f)平均滤波　(g)亮度增强　(h)对比度增强　(i)直方图均衡　(j)颜色量化　(k)边缘锐化　(l)剪切 1/4

图 5　基于 BTC 与秘密共享水印算法鲁棒性测试

其中，图 5(a)为质量因子为 5 的 JPEG 压缩实验；图 5(b)为压缩比为 13∶1 的 JPEG2000 压缩实验；图 5(g)为 30%亮度增强攻击实验；图 5(h)为 20%对比度增强；图 5(j)的颜色量化实验是将图像从 256 色量化为 16 色。NC 值为提取水印与原始水印归一化相关值，括号中数据为相应信号处理攻击下，文献[5]方法提取出的水印 NC 值，图 5(a)~图 5(l)的 NC 值分别为：0.978 5(0.830 2), 0.890 6(0.712 1), 0.881 8(0.675 6), 0.859 1(0.628 0), 0.939 9(0.899 3), 0.915 0(0.853 2), 0.998 8(1.000 0), 0.882 1(0.728 8), 1.000 0(0.991 7), 0.976 8(0.926 5), 0.999 8(0.946 4), 0.974 4(0.945 8)。图 5 实验结果表明算法对常见的图像处理攻击有较好的鲁棒性，且比文献[5]方法更好，这是因为算法中由 BTC 编码所产生的特征共享具有较好的稳健性。另外，私有共享的生成并没有引起原图像像素值的改变，易于实现多水印注册。

5　结束语

本文提出一种结合 BTC 编码与秘密共享的鲁棒图像水印算法。算法利用 BTC 编码构建原图像特征共享，由水印图像与特征共享一起生成私有共享。通过对待验证图像特征共享与私有共享的异或操作即可恢复水印图像。私有共享与特征共享的相关性确保了本文算法的水印鲁棒性；算法没有改变原始图像，实现了无损数据嵌入，易于实现对同一图像的多水印注册；通过密钥控制提高了算法安全性。实验结果表明该算法对常用信息处理攻击的良好鲁棒性。该方案可应用于医学、卫星遥感、军事等领域的图像数据版权保护与认证。

参考文献

[1] 邵利平, 覃　征, 衡星辰. 一种基于图像置乱变换的空域图像水印算法[J]. 计算机工程, 2007, 33(2): 122-124.

[2] 杨恒伏, 陈孝威. 小波域鲁棒自适应水印技术[J]. 软件学报, 2003, 14(9): 1652-1660.

[3] Cox I, Miller M, Bloom J, et al. Digital Watermarking and Steganography[M]. 2th ed. San Fransisco, CA, USA: Morgan Kaufmann Publishers, 2007.

[4] Hassan M A, Khalili M A. Self Watermarking Based on Visual Cryptography[J]. Transactions on Enformatika, System Sciences and Engineering, 2005, 8(1): 159-162.

[5] Tai Shen-Chuan, Wang Chuen-Ching, Yu Chong-Shou. Digital Image Watermarking Based on VSS in BTC Domain[J]. Journal of the Chinese Institute of Engineers, 2003, 26(5): 703-707.

编辑　张正兴

(上接第 131 页)

告中的危险函数是否把污点数据的地址作为参数，如果是，则此危险函数能引起缓冲区溢出。(4)记录人工分析检测结果，多次构造污点数据，进行人工分析。

5　缓冲区溢出检测模型的验证

静态检测生成的候选缓冲区溢出报告中的危险函数所在位置称为可能的溢出点，可能的溢出点如经动态检测和人工分析验证存在缓冲区溢出，则此溢出点称为确定的溢出点。选用免费代理服务器软件 SapporoWorks WinProxy(2.0.0)作为测试用例，测试环境为 WinXP SP2 +IDA Pro5.0+VC++6.0。通过测试查找到 35 处可能溢出点、6 处确定溢出点。在 6 处确定溢出点中有 3 处是官方公布的漏洞。

6　结束语

该检测模型将静态检测和动态检测相结合，并且对检测结果进行人工分析，降低了漏报率。在静态检测中进行缓冲区溢出判定时，不能准确得到运行时危险函数参数所代表空间的大小，只能通过分析反汇编代码中的声明区域估算其大小。在动态检测中，选择的注入数据有时很难引起缓冲区溢出发生，这些都造成了较高的误报率。该检测模型只能检测到危险函数引起的缓冲区溢出，而无法检测其他原因如循环复制操作引起缓冲区溢出，这些将是下一步研究的重点。

参考文献

[1] 余俊松, 张玉清, 宋　杨, 等. Windows 下缓冲区溢出漏洞的利用[J]. 计算机工程, 2007, 33(17): 162-164.

[2] 叶永青, 李　晖, 郑燕飞, 等. 基于二进制代码的缓冲区溢出检测研究[J]. 计算机工程, 2006, 32(18): 141-143.

[3] Evans D, Larochelle D. Improving Security Using Extensible Lightweight Static Analysis[J]. IEEE Software, 2002, 19(1): 42-51.

[4] Ruwase O, Lam M S. A Practical Dynamic Buffer Overflow Detector[Z]. (2003-07-22). http://suif. stanford.edu/papers/tunji04. pdf.

编辑　张正兴