零知识水印验证协议*

邹潇湘[1,3+], 戴 琼[2], 黄 晁[1], 李锦涛[1]

[1](中国科学院 计算技术研究所,北京 100080)

[2](中国科学院 软件研究所,北京 100080)

[3](国家计算机网络与信息安全管理中心,北京 100029)

Zero Knowledge Watermark Verification Protocols

ZOU Xiao-Xiang[1,3+], DAI Qiong[2], HUANG Chao[1], LI Jin-Tao[1]

[1](Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100080, China)

[2](Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

[3](National Computer Network and Information Security Administration Center, Beijing 100029, China)

+ Corresponding author: Phn: 86-10-82990356, E-mail: zxx@mail.nisac.gov.cn

http://www.ict.ac.cn

Received 2003-01-10; Accepted 2003-03-13

Zou XX, Dai Q, Huang C, Li JT. Zero knowledge watermark verification protocols. *Journal of Software*, 2003,14(9):1645~1651.

http://www.jos.org.cn/1000-9825/14/1645.htm

Abstract: Watermark technology has been developed to tackle the problem of unauthorized copying and distribution of digital data. Several different schemes have been proposed in the last few years, but most of them are symmetric, i.e., the key used for watermark embedding is just the same used for watermark detection. However, in many applications, an asymmetric scheme is needed, where the secrete information used to detect the watermark is not enough to modify, counterfeit or remove the watermark. In this paper, some watermark verification protocols based on bit commitment and zero knowledge proof are proposed. The ownership prover insert the watermark into the host signal using symmetric watermark technology based on spread spectrum. The watermark detect key is sent to the verifier hiding in bit commitment. By the interactive protocol between the prover and the verifier, the verifier can extract the embedded watermark, but he can not modify, counterfeit or remove it. Protocols are proposed to verify one watermark bit and several watermark bits respectively. Those protocols can be used to verify the watermark information inserted into image, audio and video using spread spectrum watermark technology.

Key words: digital watermark; symmetric watermark; asymmetric watermark; bit commitment; zero knowledge proof

摘 要: 在数字产品中嵌入数字水印,是对其进行版权保护的一种有力手段.近年来提出了不少数字水印方案,

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2001AA114010 (国家高技术研究发展计划(863))

第一作者简介: 邹潇湘(1976－),男,湖南望城人,博士,主要研究领域为数字水印技术.

但是它们中大部分都是对称的,即用于水印嵌入和水印检测的密钥是相同的.而许多实际的应用都要求非对称的数字水印方案,即水印检测时所知道的秘密不足以修改、伪造或移去水印.对基于比特承诺和零知识证明的水印验证协议进行了研究.所有权证明者采用基于扩频的对称水印技术,在宿主信号中嵌入水印;水印检测的密钥采用比特承诺的形式提交给验证者,通过证明者和验证者之间的交互协议,验证者可以提取到所嵌入的水印,但无法修改、伪造或移去水印.分别提出了验证一个和多个水印比特的协议,可应用于验证嵌入在图像、音频和视频数据中的扩频水印.

关键词: 数字水印;对称水印;非对称水印;比特承诺;零知识证明

中图法分类号: TP309 **文献标识码**: A

如何有效地防范网络环境下对数字产品的非法拷贝和传播,是知识产权管理和保护中的一个非常重要的问题.数字水印作为关键技术,为这一问题提供了一个有效的解决途径.所谓数字水印技术,就是将数字、序列号、文字、图像标志等信息嵌入到多媒体数据中,以起到版权保护、秘密通信、数据文件的真伪鉴别、产品标志等作用.

传统的数字水印技术大多是对称的,即用于水印检测的密钥与用于水印嵌入的密钥相同,同时,该密钥还可以用于移去水印.然而在一些应用中,单纯采用对称水印技术是不可行的.最典型的例子就是所有权鉴别,版权所有者在一幅作品中嵌入了自己的身份标识,在出现盗版的时候,他能够提取出其中的水印以证明自己是真正的所有者.在进行水印检测的时候,需要出示私人密钥,而私钥一旦暴露,攻击者就能够伪造、修改和移去水印,从而背离了版权保护的目的.同时,如果许多不同的数据都用同一个密钥添加了水印,那么,只要暴露了其中的一个,其他的也就都不安全了[1].

由于对称水印方案的这些固有缺陷,近年来,一些学者在积极寻求解决方案.一种方法是非对称水印算法的研究[2~6],例如,Hartung 等人[2]提出的基于扩频的方案,Schyndel 等人[3]提出的基于勒让德序列的方案,Eggers 等人[4]提出的基于线性变换特征向量的方案,Furon 等人[5]提出的基于单向信号处理函数的方案,以及 Linnartz 等人[6]提出的基于 MPEG 图像类型的方案等.然而这些方案都不是严格意义上的非对称算法,即水印检测密钥包含足够的信息来检测到所嵌入的水印,但没有足够的信息来伪造、修改或移去水印.另外,存在一种称为敏感性攻击的水印攻击方法[7],这种方法专门攻击水印检测器,如包含检测算法和密钥的防篡改设备,攻击者通过精心构造水印检测器的输入,并观察输出的变化来估计水印嵌入时的密钥.非对称算法的检测算法和公开密钥也可以从逻辑上看成是一个水印检测器,因此,现有的非对称算法一般都不能抵御这种攻击.

另一种方法是水印验证协议的研究[1,8~12].设水印证明者为 Prover,水印验证者为 Verifier.这些水印验证协议的基本思想是,Prover 采用传统的对称水印技术在多媒体数据中嵌入水印,然后,通过一些交互的协议,Prover 向 Verifier 证明其中的确嵌入了水印,但又不暴露所嵌入的水印的秘密,这类似于密码学中的零知识证明思想.在本文中,我们提出了一种基于比特承诺和零知识证明的水印验证协议,采用基于扩频的对称水印技术,在宿主信号中嵌入水印;水印检测的密钥采用比特承诺的形式提交给验证者,通过交互协议,证明者向验证者验证所嵌入的水印信息.相对于以前的只能验证水印是否存在的方案,本协议能够提取出多个比特的水印信息,协议交互时所需的数据量也较少.

1 基本概念和方法

设 X 为原始宿主信号,X 可以是图像、音频或者是视频等需要嵌入水印的数据.W 为水印信息.使用一个密钥,通过某种运算在宿主信号 X 中叠加水印信号 W,得到公开的信号 S,这个过程可以表示为 $S=X\oplus W$,这里,$\oplus$ 表示各种水印处理运算.公开的信号 S 会受到许多有意的或是无意的攻击,我们不妨假设攻击所带来的失真信号为 E,则攻击之后的信号 $T=S\otimes E$,这里,$\otimes$表示各种攻击运算.

在本方案中,采用基于扩频的方法嵌入水印,扩频水印技术有许多不同的变种,像 Cox 等人[13]提出的方案中需要原始的宿主信号,Hartung 等人[14]的方法中不需要原始的宿主信号,是比较常用的方法.不同的扩频水印方

案在实现上存在一些差异,但其基本思想是很简单的,即通过将频率扩展,在一个宽带信道(宿主信号)中传送一个窄带信号(水印信息).下面简单介绍我们所采用的扩频水印方法.

1.1 扩频水印

扩频水印的基本思想是,在宿主信号中加入一段拟随机信号.设拟随机信号 P 是由整数{−1,1}组成的序列,并且其中的−1 和 1 的数目相同.设 $u\in\{-1,1\}$ 表示待嵌入宿主信号 X 中的 1 比特的信息,且 1 比特的信息嵌入到 cr 个宿主信号中,cr 称为扩展因子.将拟随机信号 P 乘以一个小的放大因子 α,通过如下的方法加到宿主信号中,就得到了嵌入水印的信号 S.

$$S=X+\alpha\cdot u\cdot P.$$

该信号在受到有意或无意的攻击之后的信号为

$$T=S+E,$$

E 为失真信号.

在水印检测时,计算信号 T 和拟随机信号 P 之间的相关性:

$$cor(T,P)=\sum_{i=0}^{cr-1}t_i\cdot p_i=\sum_{i=0}^{cr-1}x_i\cdot p_i+\sum_{i=0}^{cr-1}e_i\cdot p_i+\sum_{i=0}^{cr-1}\alpha\cdot u\cdot p_i^2. \tag{1}$$

由于 P 的拟随机性,P 与 X 和 E 是不相关的,因此式(1)右边的第 1 项和第 2 项可以忽略,即

$$cor(T,P)\approx cr\cdot\alpha\cdot u.$$

从而提取的信息 u' 为

$$u'=\mathrm{sign}(cor(T,P)),$$

其中

$$\mathrm{sign}(x)=\begin{cases}1, & x>0\\ -1, & \text{otherwise}\end{cases}.$$

这里用相关值的正负性来判断所嵌入的水印信息,在扩展因子和放大因子已经选定的前提下,一个合理的相关值是有界的,因此,不妨设存在 L,使得

$$-2^L<cor(T,P)<2^L.$$

在水印的检测过程中,只是判断相关值的正负性是不太严格的.一种严格的方法是采用阈值判断,要求检测到的相关值与理论值(这里是 $cr\cdot\alpha\cdot u$)的差值应该在某个范围之内,即存在 M,使得

$$-2^M<\left|cor(T,P)\right|-cr\cdot\alpha<2^M.$$

例如,扩展因子 $cr=2^{10}$,放大因子 $\alpha=2^1$ 时,L 可取值为 12,即要求检测到的相关值不超过理论值的两倍,M 可取值为 6,即要求检测到的相关值与理论值的差不超过 64.

在扩频水印方案中,拟随机序列 P 是水印检测的关键信息,然而,公开 P 使得攻击者可以伪造、修改甚至移去水印.因此,需要隐藏有关 P 的秘密,在这里,我们采用比特承诺方案来实现这一点.

1.2 比特承诺方案

这里,采用文献[15]中的比特承诺方案.设 N 是两个质数 p 和 q 的乘积,且 $p'=(p-1)/2$ 和 $q'=(q-1)/2$ 也是质数, $N'=p'q'$ .g 是 Z_N^* 的 N'阶循环子群 G 的生成元, $h=g^{\alpha}\bmod N$,其中α随机取自 $Z_{N'}^*$.对于 $x\in\{0,1,...,N-1\}$ 的承诺为

$$BC(x)=g^xh^r\bmod N,$$

r 是一个辅助随机数,从一个足够大的范围,比如 $[0,2^lN)$ 内选取.

我们采用比特承诺来隐藏水印检测的密钥序列 P,对于随机序列 $p_i\in\{-1,1\}$, $i=0,1,...,cr-1$ 的承诺为

$$BC(p_i)=g^{p_i}h^{r_i}\bmod N.$$

Prover 需要向 Verifier 证明隐藏在 $BC(p_i)$ 中的数的确是 1 或−1,因此他需要证明自己知道 $BC(p_i)/g$ 或 $BC(p_i)g$ 的基于 h 的离散对数,这可以通过利用文献[16]中证明知道离散对数的方法,以及文献[17]中的证明知道两个秘密之一的方法来实现.在协议中还要证明承诺满足 $BC(2^i)$ 和 $BC(0)$ 的形式,这也可以采用同样的方法

来实现.

2 水印验证方案

2.1 验证1比特信息

采用上面的扩频水印方案在宿主信号中嵌入水印之后,证明者 Prover 需要向验证者 Verifier 证明 T 中包含水印信息.为了不暴露拟随机信号 P,将其表示成比特承诺的形式:

$$BC(p_i)=g^{p_i}h^{r_i}.$$

这样,在计算相关性 $cor(T,P)$ 时,乘法运算转化为求幂,加法运算转化为求乘积,即

$$\prod_{i=0}^{cr-1}BC(p_i)^{t_i}=g^{cor(T,P)}h^b=BC(cor(T,P)),$$

这类似于文献[11]中求相关的方法.

接下来,Prover 需要向 Verifier 验证包含在上述承诺中的相关性 $cor(T,P)$ 的正负性,Prover 向 Verifier 出示 $cor(T,P)$ 和 b 是最直接的方法,但是,一旦暴露了相关值,就有可能暴露随机序列 P 的秘密,因为如果能够知道相关值,攻击者就可以采用类似于文献[7]中的敏感性攻击方法:先用一组信号 T,通过验证得到相关值 $cor1$;将该组信号的第 i 个信号值加 1,再次进行验证得到另一个相关值 $cor2$.根据相关值的计算公式,显然,如果 $cor1>cor2$,则 $p_i=-1$,否则 $p_i=1$.这样,将上述过程进行 cr 次,就可以得到整个随机序列 P.

因此,在我们的协议中,不是直接出示相关值,而是基于相关值是有界的前提,Prover 将 $|cor(T,P)|$ 表示成 L 个比特承诺的乘积形式,并向 Verifier 证明这种表示方法的正确性.下面给出具体的水印验证协议,这里通过相关值的正负性来判断所嵌入的比特.

协议 1.

(1) Prover 提交 cr 个比特承诺 $BC(p_i)=g^{p_i}h^{r_i}$,其中 $i=0,1,...,cr-1$.Prover 同时提交 L 对比特承诺 $\{BC(2^i),BC(0)\}=\{g^{2^i}h^{\bar{r}_i},h^{\bar{r}_i}\}$,其中 $i=0,1,...,L-1$,但每一对中两个承诺的顺序对 Verifier 是保密的.Prover 以零知识证明协议向 Verifier 证明以上承诺是正确的形式(见第 1.2 节).

(2) Verifier 计算

$$\prod_{i=0}^{cr-1}BC(p_i)^{t_i}=g^ah^b,$$

其中:

$$a=\sum_{i=0}^{cr-1}p_it_i=cor(T,P),$$

$$b=\sum_{i=0}^{cr-1}r_it_i.$$

(3) Prover 直接计算 a 和 b,将 a 的绝对值表示成二进制形式:

$$|a|=\sum_{i=0}^{L-1}a_i2^i,$$

$a_i\in\{0,1\}$,对 $i=0,1,...,L-1$,从第 i 对承诺 $\{BC(2^i),BC(0)\}$ 中,选择是 a_i2^i 的比特承诺的那一个,用 $b_i\in\{0,1\}$ 表示 Prover 的选择.Prover 计算

$$\prod_{i=0}^{L-1}BC(a_i2^i)=g^{|a|}h^d,$$

Prover 向 Verifier 出示 $c=sign(a)$,b_i 和 $m=d-c\cdot b$.

(4) 通过 b_i,Verifier 计算

$$\prod_{i=0}^{L-1}BC(a_i2^i)=g^{|a|}h^d.$$

Verifier 验证

$$(g^a h^b)^c h^m = (g^{|a|} h^d)$$

是否成立,如果成立,则 Verifier 相信检测到了 1 比特的信息 c.

正如我们前面所分析的,在很多时候我们并不只是简单地判断相关值的正负性,而要判断所检测到的相关值与理论值的差值,是否在一个给定的阈值范围 $(-2^M, 2^M)$ 之内.类似于协议 1,我们这里只要将这个差值表示为 M 个比特承诺的乘积形式即可.基于阈值的水印验证协议如下:

协议 2.

(1),(2)类似于协议 1 的(1),(2).不同之处在于需要提交的不是 L,而是 M 对比特承诺 $\{BC(2^i), BC(0)\} = \{g^{2^i} h^{\bar{r}_i}, h^{\bar{r}_i}\}$,其中 $i = 0,1,...,M-1$.

(3) Prover 直接计算 a 和 b,将 $k = |a| - cr \cdot \alpha$ 的绝对值表示成二进制形式:

$$|k| = \sum_{i=0}^{M-1} k_i 2^i,$$

$k_i \in \{0,1\}$,对 $i = 0,1,...,M-1$,从第 i 对承诺 $\{BC(2^i), BC(0)\}$ 中,选择是 $k_i 2^i$ 的比特承诺的那一个,用 $b_i \in \{0,1\}$ 表示 Prover 的选择.Prover 计算

$$\prod_{i=0}^{M-1} BC(k_i 2^i) = g^{|k|} h^d.$$

Prover 向 Verifier 出示 $c = \text{sign}(a)$, $c' = \text{sign}(k)$, b_i 和 $m = c' \cdot d - c \cdot b$

(4) 通过 b_i,Verifier 计算

$$\prod_{i=0}^{M-1} BC(k_i 2^i) = g^{|k|} h^d.$$

Verifier 验证

$$(g^a h^b)^c h^m = (g^{|k|} h^d)^{c'} g^{cr \cdot \alpha}$$

是否成立,如果成立,则 Verifier 相信检测到了 1 比特的信息 c.

2.2 验证多个比特信息

如果需要提取 n 比特的水印信息,一种方法是将上述水印验证协议重复执行 n 次,但实际上,在水印协议中,有许多可以利用的重复信息,下面是验证 n 比特水印信息的协议.

协议 3.

(1) 执行协议 1(或协议 2,以下以协议 1 为例)的第(1)步,Prover 向 Verify 提交 cr 个比特承诺 $BC(p_i) = g^{p_i} h^{r_i}$ 和 L 对比特承诺 $\{BC(2^i), BC(0)\} = \{g^{2^i} h^{\bar{r}_i}, h^{\bar{r}_i}\}$.

(2) 循环执行协议 1(或协议 2)的第(2)~(4)步 n 次,每次 Prover 向 Verify 验证 1 比特的水印信息.

可以看到,需要交换大量数据的提交比特承诺的步骤放在了协议的第(1)步,这里需要提交 $cr + L$ 个比特承诺,并采用零知识证明协议证明其为正确的形式.但这一步只需要执行 1 次,因此完全可以在预处理中进行,而协议循环执行的部分,每验证 1 比特的水印信息,Prover 向 Verify 传送的数据量为

$$L + 1 + O(|N|)$$

比特.同时,Prover 的计算量约为

$$(2cr) \text{ 次乘法运算}, (2cr + L) \text{ 次加法运算},$$

Verifier 的计算量约为

$$(cr) \text{ 次指数运算}, (cr + L) \text{ 次乘法运算}.$$

两者的计算复杂度为 cr 的多项式时间.

扩频水印对于压缩、添加噪声、过滤等处理具有较强的鲁棒性,考虑到这些有意或无意的攻击,可以选取较大的 L 和 M 值(见第 1.1 节),使得能够从攻击后的信号中可靠地提取出水印信号来.

3 分 析

上面我们提出了验证一个和多个比特水印信息的水印验证协议.一个需要注意的问题是,上面验证 1 比特

信息的基本协议不能防止一些水印伪造攻击.例如,T 中并没有嵌入任何水印,但 Prover 可以这样来构造 P[18]:如果想让用户检测到+1,则选取序列中+1 所对应的位置的宿主信号的平均值比−1 所对应的位置的宿主信号的平均值大,比如说将信号 T 排序,信号值大于中值的位置,相应的 p_i 取值为+1,其余 p_i 取值为−1,这样构造的序列 P 与 T 显然具有很大的相关值,为了不使相关值过大,可以将 P 中的+1 和−1 随机交换一部分,直到得到合适的相关值,同时 P 也满足了一定的随机性为止.同理,也可以让用户检测到−1.由于知道宿主信号,这样的"随机序列"是很容易构造出来的.这样,Verifier 就有可能检测到实际不存在的水印.

在一般的检测方案中,需要出示 P,该随机序列是某个随机数生成器通过一个种子值生成的,或者是宿主信号的某个单向函数值,Prover 不能预测随机数生成器或单向函数的输出,因此可以避免上述"精心挑选 P"的攻击方法.而在水印验证协议中,P 是以比特承诺的形式出现的,因此 Verifier 无法确信 P 不是 Prover 伪造的.为了避免这一问题,可以将比特承诺 $BC(p_i)=g^{p_i}h^{r_i}$ 由一个值得信赖的仲裁者 Trent 签名,这样,水印嵌入和验证的过程如下:

协议 4.

(1) Prover 向可靠的 Trent 提供 P,其中 P 是一个随机数生成器或者单向函数的输出.

(2) Trent 验证 P 的真实性,并对 P 的比特承诺 $BC(p_i)$ 进行签名.

(3) Verifier 验证对 $BC(p_i)$ 的签名,然后执行上述的协议来提取水印比特.

这样,为了避免伪造水印的问题,需要引入一个仲裁者,但这给协议增添了不少复杂性.Trent 必须是完全值得信赖的,Prover 相信他不会滥用自己的水印嵌入密钥 P (Prover 可以和 Trent 执行盲签名协议对 P 进行签名,但这使得问题更加复杂化),Verifier 相信他不会和 Prover 串通来伪造 P .而且,既然 Prover 和 Verifier 都需要相信某个仲裁者,那么可以直接引入这样一个值得信赖的仲裁者 Trent:Prover 向 Trent 出示密钥 P 验证水印,再由 Trent 告诉 Verifier 他检测到的水印.

下面考虑这种攻击对验证多个(n 个)比特的协议是否有效.例如,在一幅分辨率为 512×512 的彩色 RGB 图像的亮度分量中嵌入水印,设扩展因子 $cr=1024$,则可以嵌入 256 比特,即 32 字节的信息,完全可以表示一段有意义的信息,如所有权者的名字、数字作品的创作日期等.Verifier 只有在检测到一段有意义的信息时,才确信其中的水印.在判断相关值正负性的检测协议中,上面的攻击方法伪造 P 来控制一个相关值的正负性是比较容易的.这是因为攻击者选取的 P 并不是嵌入算法所要求的那样与 X 不相关,正好相反,伪造的 P 与宿主信号有一个小的相关值,为了伪造多个比特, P 必须与每段宿主信号具有特定(正或负)的相关值,但实际上,当宿主信号的长度足够长的时候,每段宿主信号之间的相关性就已经很小了,因此,当 n 足够大的时候,攻击者会面临以下两个问题:构造 P ,与两个不相关的宿主信号都同时具有正(或负)的相关值;构造 P ,与两个相关的宿主信号一个具有正的相关值,一个具有负的相关值.正如可以选取足够大的 cr ,使得 P 与宿主信号不相关一样,同样可以选取足够大的 n (比如,经验值是 $n\geq128$),使得无法伪造 P .

另外,采用阈值检测的方法,更增加了伪造密钥的难度,因为这使得攻击者必须构造一个随机序列 P , P 与 T 的相关值在一个给定的范围之内.如果 n 足够大,前面协议的第(1)步中利用零知识证明[16,17]来证明 $BC(p_i)$ 的确是对 1 或−1 的承诺的步骤也可以省略,因为要找到一个伪造的 P 是困难的.

现有的一些水印验证协议[1,8~12]不是很完善,在协议执行时需要传输大量的数据,并且这些方案都是验证水印是否存在,即通过能否正确执行验证协议来判断是否存在水印:如果能够正确执行协议,则表明嵌入了水印,否则表明没有嵌入水印.而在我们的基本方案中,不仅能判断是否嵌入了水印,而且还能给出一个二值判断,即嵌入的水印信号是 1 还是−1.同时,还将该方案推广到验证多个比特水印信息的情况,其特点是,需要传输大量数据的交互阶段可以放在预处理阶段,以后每验证 1 比特的水印信息,只需传输少量的数据.因此,在图像、甚至是视频水印验证的应用中,采用该水印验证协议也是可行的.

4 结 语

在本文中,我们提出了一种基于比特承诺和零知识证明的水印验证协议,以保证在水印检测时,不暴露所嵌入的水印的秘密,使得攻击者难以利用水印检测时的信息来修改、伪造或移去水印.在许多应用中,比如版权所有者在多媒体数字产品中嵌入自己的身份标记,需要向他人证明的确嵌入了这些信息,而又不愿暴露自己的水

印秘密的情况下,可以采用这里的多个比特的水印验证协议.

我们进一步的工作是,在验证 1 比特水印信息的协议中,如何利用协议和密码算法更有效地防止水印伪造问题,以及在验证多个比特水印信息的协议中,进一步降低预处理的时间和空间复杂度.

致谢 感谢中国科学院计算技术研究所北纬通信无线多媒体技术联合实验室为我们的研究提供资助,感谢中国科学院计算技术研究所领域前沿青年基金(20016280-15)提供资助.中国科学院计算技术研究所博士谭建龙、刘书昌、胡春光对本文的完成提出了很多有益的建议,在此一并表示感谢.

References:

[1] Craver S, Katzenbeisser S. Copyright protection protocols based on asymmetric watermark: The ticket concept. In: Proceedings of the 6th Conference on Communication and Multimedia Security (CMS 01). 2001. 159~170. http://www.ifip.tu-graz.ac.at/TC6/events/CMS/cms01.htm.

[2] Hartung F, Girod B. Fast public-key watermarking of compressed video. In: Proceedings of the IEEE International Conference on Image Processing (ICIP'97). 1997. 528~531. http://clip.informatik.uni-leipzig.de/~toelke/Watermark/ip971113.pdf.

[3] Schyndel RGV, Tirkel AZ, Svalbe ID. Key independent watermark detection. In: IEEE International Conference on Multimedia Computing and Systems. 1999. 580~585. http://lci.det.unifi.it/Staff/Piva/Watermarking/Docs/icmcs99.html.

[4] Eggers JJ, Su JK, Girod B. Public key watermarking by eigenvectors of linear transforms. In: Proceedings of the European Signal Processing Conference (EUSIPCO 2000). 2000. http://graphics.tu-bs.de/v3d2/pubs.collection/diwa_eus2000eggers.pdf.

[5] Furon T, Duhamel P. Robustness of asymmetric watermarking technique. In: Proceedings of the IEEE International Conference on Image Processing (ICIP 2000). 2000. 21~24. http://lci.det.unifi.it/Staff/Piva/Watermarking/Docs/programicip.html.

[6] Linnartz JPMG, Talstra JC. MPEG PTY-marks: Cheap detection of embedded copyright data in DVD-video. In: Computer Security -ESORICS 98, 5th European Symposium on Research in Computer Security. Lecture Notes in Computer Science 1485, 1998. 221~240. http://buffy.eecs.berkeley.edu/~linnartz/wpapers.html.

[7] Kalker T, Linnartz JP, Dijk MV. Watermark estimation through detector analysis. In: Proceedings of the IEEE International Conference on Image Processing (ICIP'98). 1998. 425~429. http://buffy.eecs.berkeley.edu/~linnartz/wpapers.html.

[8] Craver S. Zero knowledge watermark detection. In: International Workshop on Information Hiding (IHW'99). Lecture Notes in Computer Science 1768, 1999. 101~116. http://citeseer.ist.psu.edu/craver00zero.html.

[9] Craver S, Katzenbeisser S. Security analysis of public-key watermarking schemes. In: Schmalz MS, ed. Mathematics of Data/Image Coding, Compression, and Encryption IV, with Applications. Proceedings of the SPIE Vol. 4475, 2001. 172~182. http://www.dbai.tuwien.ac.at/staff/katzenb/pubs.html.

[10] Gopalakrishnan K, Memon N, Vora P. Protocols for watermark verification. In: Multimedia and Security Workshop at ACM Multimedia'99. 1999. 103~106. http://www.hpl.hp.com/personal/Poorvi_Vora/Pubs/acm99.pdf.

[11] Adelsbach A, Sadeghi AR. Zero-Knowledge watermark detection and proof of ownership. In: International Workshop on Information Hiding (IHW 2001). Lecture Notes in Computer Science 2137, 2001. 273~288. http://www-krypt.cs.uni-sb.de/download/papers/AdSa_01.pdf.

[12] Adelsbach A, Katzenbeisser S, Sadeghi AR. Cryptography meets watermarking: detecting watermarks with minimal or zero knowledge disclosure. In: Proceedings of the European Signal Processing Conference (EUSIPCO 2002). 2002. http://www.dbai.tuwien.ac.at/ staff/katzenb/download/eusipco02.ps.gz.

[13] Cox IJ, Kilian J, Leighton T, Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing, 1997,6(12):1673~1687.

[14] Hartung F, Girod B. Watermarking of uncompressed and compressed video. Signal Processing, Special issue on Copyright Protection and Access Control for Multimedia services, 1998,66(3):283~301.

[15] Fujisaki E, Okamoto T. A practical and provable secure scheme for publicly verifiable secret sharing and its applications. In: Nyberg K, ed. Advances in Cryptology EUROCRYPT'98. Lecture Notes in Computer Science 1403, 1998. 32~46. http://www.uni-giessen.de/crypto/kryptoag/Bibliothek/Bibliothek.htm.

[16] Schnorr CP. Efficient signature generation by smart cards. Journal of Cryptology, 1991,4(3):161~174.

[17] Cramer R, Damgard I, Schoenmakers B. Proofs of partial knowledge and simplified design of witness hiding protocols. In: Proceedings of the Crypto'94. Lecture Notes in Computer Science 839, 1994. 174~187. http://www.win.tue.nl/~berry/papers/crypto94.pdf.

[18] Craver S, Memon N, Yeo BL, Yeung MM. Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications. IEEE Journal on Selected Area in Communications, 1998,16(4):573~586.