Optical Engineering 46(10), 107202 (October 2007)

Concatenated construction of traceability codes for multimedia fingerprinting

Yu-Tzu Lin
National Taiwan University
Department of Computer Science and Information Engineering
No. 1, Sec. 4, Roosevelt Road
Taipei, Taiwan 106
E-mail: linyt@cmlab.csie.ntu.edu.tw

Ja-Ling Wu
National Taiwan University
Department of Computer Science and Information Engineering
and
Graduate Institute of Networking and Multimedia
No. 1, Sec. 4, Roosevelt Road
Taipei, Taiwan 106

Chun-Hsiang Huang
National Taiwan University
Department of Computer Science and Information Engineering
No. 1, Sec. 4, Roosevelt Road
Taipei, Taiwan 106

Abstract. We develop a framework of construction strategies for traceability codes on the subject of multimedia fingerprinting, which constructs traceability codes in a concatenated way. In fingerprinting systems, there are tradeoffs among the size of the customer base, the collusion resilience, and the codeword length. Our goal is to find a strategy to construct, traceability codes that reach an equilibrium among these tradeoffs. Instead of investigating a "high-performance" code, which has proven challenging in existing researches, we introduce a concatenated construction methodology to provide a flexible approach for producing more applicable traceability codes by composing several existing "low-performance" collusion-secure codes. Instead of just giving an example constructed by specific codes, we describe a general code construction strategy and give a detailed analysis on it. Moreover, we propose two concatenated construction schemes based on group fingerprinting to show the feasibility of the concatenated construction strategy. Experimental results show good performances on the code length, detection time, and tracing ability. [DOI: 10.1117/1.2790911]

Subject terms: fingerprinting; traitor tracing; collusion resistance; traceability codes; multimedia security.

Paper 070001RR received Jan. 2, 2007; revised manuscript received Apr. 20, 2007; accepted for publication Apr. 24, 2007; published online Oct. 5, 2007.

1 Introduction

In applications of multimedia distribution such as pay TV, data are authorized for use by some privileged users and not others. A fingerprinting system protects media data by embedding a unique key in the host data of each user. These keys can be used to identify the source of illegal copies. However, a coalition of users may collude to make an illegal copy that is different from all colluders' copies. Collusion-resistant fingerprinting solves this problem. Boneh and Shaw initiated the collusion-secure fingerprinting, which considered how to design the fingerprint codes to resist the coalition of c traitors, called frameproof codes.[1] In Ref. 2, a traceability (TA) scheme is proposed that describes the resilience of the threshold tracing schemes, which inherited ideas from broadcast encryption schemes.[3] Stinson presented several constructions for broadcast encryption schemes, such as orthogonal arrays, universal hash families, and t-designs.[4,5] Combinatorial properties of frameproof and traceability codes are derived in Ref. 6, in which a c-TA scheme is defined and the combinatorial structures like t-designs and packing designs are used to construct frameproof codes and traceability schemes. Besides theoretical works from the viewpoint of fingerprint design, some literature proposes practical fingerprinting systems for multimedia. In Ref. 7, a collusion-secure fingerprinting scheme based on finite geometries is presented. Several researches constructed the collusion-secure fingerprints based on orthogonal noiselike signals, which are suitable for multimedia fingerprinting.[8] By using code division multiple access (CDMA), the orthogonal noiselink signals and combinatorial designed codes were combined to produce fingerprints.[9] They constructed an anticollusion code for multimedia with the aid of balanced incomplete block design (BIBD) and applied the code-modulation technique to reduce the spreading sequences.[10] In addition to the combinatorial methods, another trend of fingerprint codes employs the concept of error-correcting codes (ECCs). The idea is that one can identify the colluders' codewords if the minimum distance of the code is large enough. In Ref. 11, Reed-Solomon (RS) codes are used to construct a traceable code, and list-decoding techniques are used to find all possible pirate coalitions efficiently. The authors also describe how theoretical coding techniques may be applied to the traitor tracing problem for linear codes.

Although related research has been conducted for more than 10 years, no systematic decoding algorithm exists for spread-spectrum-based methods, and the tracing ability is limited when the size of the collusion coalition becomes large. The combinatorial design can help find structured fingerprinting codes, which provide good properties for uniquely identifying colluders. But the combinatorial design is not trivial, so the construction of this type of code is difficult and the scalability is limited. The ECC-based fingerprinting codes treat the fingerprint generation problem as an ECC design problem. The mathematical operations on finite fields for ECCs, such as RS codes, provide more regularity and efficiency for both encoding and decoding

0091-3286/2007/$25.00

than that of the combinatorial methods. However, the attempt to increase the minimum distance between code words results in an abrupt increase in the code length.

The orthogonal noiselike signals, combinatorial designs, ECCs, and other code construction methods for fingerprinting codes all have tradeoffs among the size of customer base n, the collusion-resilience c, and code-word length l. As n or c increases, either l grows abruptly or the existence of construction is still unknown. This tradeoff makes the traceability code impractical because a large customer base and collusion resilience are needed in many applications, but these requirements will make the code-word length very long. Besides, all of the prescribed collusion-secure codes have their weaknesses, as mentioned above. Therefore, instead of making an effort to find a high-performance traceability code, we try to derive a generalized framework of designing strategy to hierarchically construct a traceability code that has high-performance parameters (i.e., short code-word length, reasonable collusion-resilience, and an applicable size of the customer base) by composing several low-performance traceability codes together. Reference 12 presented a similar idea, which expands the Boneh-Shaw scheme and analyzes the error probabilities for various combinations of different codes. We go further by stating the requirements of the inner and outer codes, give detailed analyses, and provide real examples to show good performances of the proposed method.

In the rest of this paper, we will briefly introduce the collusion-secure codes for fingerprinting. Section 2 gives the analyses, and Sec. 3 presents the concatenated construction strategy for traceability codes. We then propose in Sec. 4 two concatenated traceability schemes: the group-oriented traceability scheme and the hierarchical traceability scheme. Section 5 provides concluding remarks.

2 Collusion-Secure Fingerprinting Codes

Many existing works on designing various collusion-secure fingerprinting codes have different properties for resisting collusion attacks. Some literature constructs collusion-secure codes on the basis of combinatorial designing strategies.[2,7] This type of code identifies colluders through the peculiar allocation of symbols to the code words. Reference 1 defined the term "c-secure," which means a pirate copy can be traced back to at least one member of the collusion coalition with size no greater than c. Another term, "c-TA code," proposed by Staddon et al. takes traceability codes as ECCs.[11] They guarantee the identification of one of the traitors by the minimum distance decoding. Reference 8 used orthogonal noiselike signals as fingerprints, and Ref. 9 modulated the fingerprints on the basis of the spread-spectrum method. Although orthogonal signals are signals with large distances and the related schemes should belong to the ECC-based category, we consider them separately for the ease of explanation.

In the following sections, we first introduce several important terms and briefly describe two main types of collusion-secure codes. Some analyses are also given for the ease of later discussions.

2.1 *Combinatorial-designed Codes*

Combinatorial-designed codes achieve collusion-resiliency by their particular structures; t-design is one of the widely used designs.[13] It has been proved that if there exists a t-$(v,k,1)$-design, then there exists a c-traceability scheme, where $c=\lfloor\sqrt{(k-1)/(t-1)}\rfloor$.[6] That is, $t-(v,k,1)$-designs guarantee the needed discrimination among different collusion coalitions. A well-known example of combinatorial designed codes is the BIBD codes. The following lists the blocks of a (7,3,1)-BIBD on the set $S=\{0,1,\ldots,6\}$:

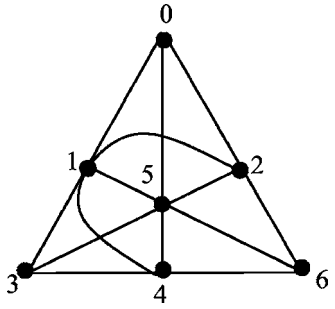

Fig. 1 Fano plane.

$$\{0,1,3\},\{0,2,6\},\{0,5,4\},\{1,5,6\},\{2,1,4\}\{2,5,3\},\{3,4,6\}.$$

This design is shown in Fig. 1 and is called the Fano plane.[13] If the blocks are represented as an incidence matrix

$$A\left(A_{ij}=\begin{cases}0 & \text{if block } i \text{ contains element } j,\\ 1 & \text{otherwise.}\end{cases}\right),$$

then rows of A are the code words of the combinatorial designed code.

Boneh and Shaw defined c-secure codes and proposed a code Γ_0 similar to the combinatorial-designed codes, in which specific bits are used to identify the user information:[1]

Definition 1: c-secure codes.[1]

A code Γ is totally c-secure if there exists a tracing algorithm A that satisfies this condition: if a coalition $\mathcal{C}$ of at most c users generates a word x, then $A(x)\in\mathcal{C}$.

Definition 1 says that one code is c-secure if there exists an algorithm that can identify one of the code words of the colluders. But according to Boneh and Shaw's proof, when $c>1$, totally c-secure codes do not exist if the marking assumption (defined in Ref. 1, and detailed later) is satisfied. They used randomness to derive c-secure codes with ε-error, which enables the capture of one member of the illegal coalition with a probability of at least $1-\varepsilon$. They introduced an (l,n)-code that is n-secure with ε-error for any $\varepsilon>0$. Let c_m be a column of height n in which the first m bits are 1 and the rest are 0. The code $\Gamma_0(n,d)$ consists of all columns $c_1,c_2,\ldots,c_{n-1}$, and each is duplicated d times. For example, the code $\Gamma_0(4,3)$ for four users u_0, u_1, u_2, and u_3 becomes:

u_0: 111111111

u_1: 000111111

u_2: 000000111

u_3: 000000000.

Then the bits of each code word must be permuted randomly (all code words have the same permutation) to obtain the code.

To construct a logarithmic-length c-secure code with ε-error, Boneh and Shaw then composed $\Gamma_0(n,d)$ and C, in which C is an error-correcting code. Although $\Gamma_0(n,d)$ is "n-secure," the reason for not using $\Gamma_0(n,d)$ directly is that the length of this code is $O[n^3\log(n/\varepsilon)]$, which is impractical in many applications.

2.2 *Codes with Large Minimum Distances*

Combinatorial designs, however, are not easy and need some tricks. For other types of collusion-secure codes, the error-correcting codes with large minimum distances and the orthogonal signals can be constructed in a simpler way rather than with special designing tricks (ECCs even have the systematic encoding and decoding rules). Both ECCs and orthogonal signals can be applied to generate distance-based collusion-secure codes. Based on the distance criterion, c-TA code is defined as follows:

Definition 2: c-TA Codes.[11]

Suppose C is an $(l,n,d)_q$-ECC and $c \geqslant 2$ is an integer. Let $C_i \subseteq C$, $i=1,2,\ldots,t$, be all the subsets of C such that $|C_i| \leqslant c$, then C is a c-TA code if for all i and all $x \in \mathrm{desc}(C_i)$, there is at least one code word $y \in C_i$ such that $|I(x,y)| > |I(x,z)|$ for any $z \in C \backslash C_{ij}$, in which $|x|$ denotes the cardinality of x, $I(x,y)=\{i: x_i=y_i\}$, $\mathrm{desc}(C_0)=\{x \in Q^N: x_i \in \{a_i: a \in C_0\}, 1 \leqslant i \leqslant N\}$, and Q is the alphabet set.

It should be noted that in this definition, the marking assumption is no longer satisfied. The marking assumption in Ref. 1 says that any coalition of c users is only capable of creating a pirate work lying on the feasible set, which is defined as follows:

Definition 3: feasible set $F(C;\Gamma)$.[1]

Let $\Gamma=\{x^1,\ldots,x^n\}$ is an (l,n)-code and C be a coalition of users, then $F(C;\Gamma)=\{w \in (\Sigma \cup \{?\})^l$ such that $w|_R = w^u|_R$, $u \in C\}$, in which Σ is the alphabet, "?" stands for the unreadable mark, and R is the set of undetectable positions for C.

In this paper, because of the assumption of the c-TA code and the scenario of multimedia fingerprinting, we use a stronger version of the marking assumption in which the feasible set is redefined:

Definition 4: strict feasible set $F_s(C;\Gamma)$.

$$F_s(C;\Gamma) = [(w_1,\ldots,w_n): \forall\, i, \exists\, (x_1,\ldots,x_n) \in C \text{ such that } x_i = w_i].$$

Obviously, the pirate copies lying in the strict feasible set are just what the copy-and-paste attacks could create. As mentioned above, Boneh and Shaw proved that there are no totally c-secure (l,n)-codes when $c \geqslant 2$ and $n \geqslant 3$, where l is the code length and n is the number of code words. But if the distance concept and the strict marking assumption are introduced, this theorem will no longer be true. The proof is trivial because the pirate can be caught by simply employing the minimum distance decoding:

Octal form

Codeword 1	7	2	3	2	3	6	5
Codeword 2	7	3	0	2	3	1	1

Binary form

Codeword 1	1 1 1	0 1 0	0 1 1	0 1 0	0 1 1	1 1 0	1 0 1
Codeword 2	1 1 1	0 1 1	0 0 0	0 1 0	0 1 1	0 0 1	0 0 1

Fig. 2 The distances between two code words in the octal form and the binary form, where the positions in which code word 1 is different from code word 2 are marked by blocks.

Theorem 1

c-TA codes are totally c-secure codes.

2.2.1 *Error correcting codes with larger relative minimum distances*

In order to meet the criteria of minimum distance for traceability ECCs [i.e., $d > l(1-1/c^2)$], c will be restricted to a small range because it is not easy to find ECCs with such a large minimum distance. Note that c is restricted by the relative ratio of d to l, but not d or l alone. One of the solutions is to increase the alphabet size q. If the number of code words of an ECC is fixed, the probability of having a large minimum relative distance (or distance ratio) increases when q becomes large:

Proposition 1

A b-ary ECC has a larger minimum relative distance if it is represented as the q-ary form (in which $q \geqslant b$).

Proposition 1 can be simply illustrated by the following example: Suppose an ECC C over GF(8) has code words (7 2 3 2 3 6 5) and (7 3 0 2 3 1 1), the Hamming distance between these two code words is 4, and the relative distance (defined as the ratio of distance to the code word length) is 4/7. If the code words are represented in binary form, the distance becomes 7 and the ratio is 7/21, which is smaller than that of the octal one. As shown in Fig. 2, one octal digit is transformed into three binary digits, but one position difference of two octal code words may introduce only one or two differences in the three binary digits. In fact, as an example of ECCs, if RS codes are adopted, the relationship among c, k, and q can be derived as

$$c < -\frac{1}{2(k-1)} + \sqrt{\frac{1}{4(k-1)^2} + \frac{q-1}{k-1}}, \tag{1}$$

where k is the dimension of the code.[14]

Theorem 6 of Ref. 11 gave a similar result: if k and c are positive integers, q is a prime power, $q > c^2 \geqslant 4$, and δ is a real number such that $0 < \delta \leqslant q/c^2 - 1/q - 1$, then there exists an explicit linear c-TA code over the field F_q of length $l = O(k^2/\delta^3 \log(l/\delta))$. Therefore, q-ary fingerprints with larger q will be more applicable than binary ones in traitor tracing systems because it is easier to find an ECC with a larger alphabet size having a larger minimum relative distance. Note that, without loss of generality, we will interchangeably use "minimum relative distance" and "minimum distance" in the rest of this paper.

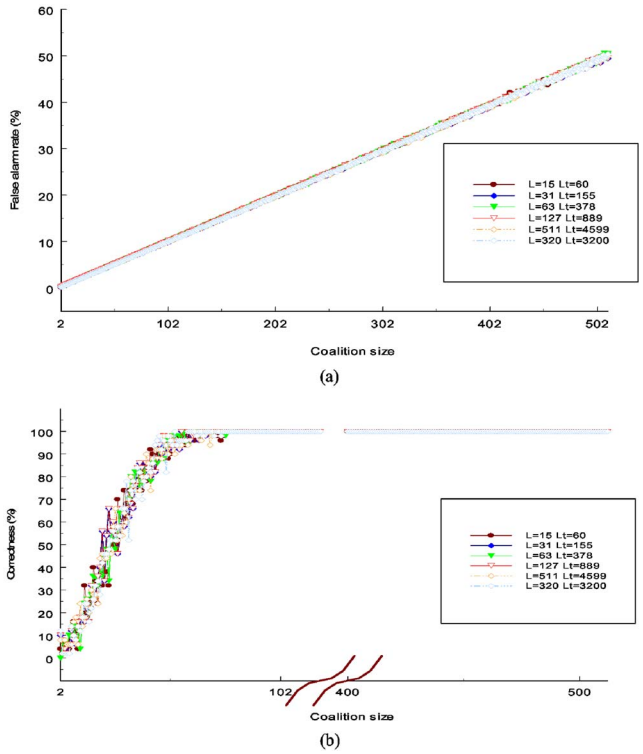

Fig. 3 Performances of the pseudorandom sequences: (a) false alarm rates, and (b) correctness. In (b), we cut the range of the coalition size from 120 to 400 because the correctness is constant in this range.

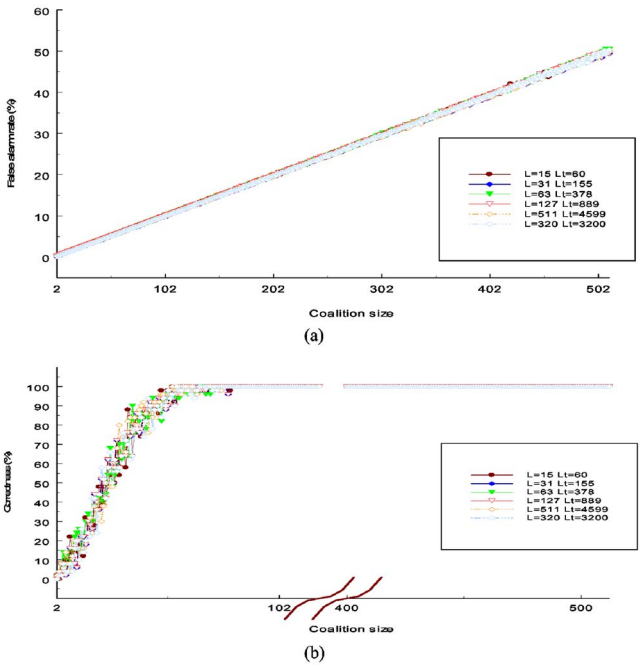

Fig. 4 Performances of the Reed-Solomon codes: (a) false alarm rates, and (b) correctness. In (b), we cut the range of the coalition size from 120 to 400 because the correctness is constant in this range.

As for the RS code, because it is a maximum-distance code, it can be derived that $d=l-k+1$ (where k is still the dimension of the code). Therefore, $l-k+1>l(1-1/c^2)$, and $l>c^2(k-1)$. That is to say, for a fixed dimension size, a longer code length of an RS code implies higher collusion resiliency.

2.2.2 *Orthogonal signals*

Orthogonal-signal based fingerprint can be orthogonal itself or be modulated by orthogonal signals. Cox et al. proposed the spread-spectrum watermarking method, which embeds watermarks that add noiselike signals to the host signals.[15] Some literature constructs fingerprints by orthogonal modulation, and the fingerprints produced are spread-spectrum sequences. Reference 8 employed Gaussian-distributed fingerprints by orthogonal modulation and used a likelihood-based approach to estimate the number of colluders. Reference 9 modulated BIBD codes by orthogonal bases, which again are spread-spectrum sequences. The noiselike signals of this type of fingerprint are obviously suitable for multimedia contents.

2.3 *Comparisons of Existing Collusion-Secure Codes*

Figures 3 and 4 present the performances of the orthogonal signals and the ECCs, in which L is the length of q-ary codes, and L_t is the length of binary ones. We use pseudorandom sequences and RS codes as examples of orthogonal signals and ECCs, respectively. False alarm rates and correctness are provided to show the performances, where the correctness is defined as the probability of catching at least one of the colluders correctly, and the false alarm rate is the probability of pronouncing the innocent user guilty. The minimum distance decoding is applied to find the top w suspects, where w is the collusion size. That is, we simplify the problem by supposing the collusion size is known, and we always catch w suspects despite the distribution of possibilities. This assumption is reasonable because the purpose of this experiment is to compare the code performances of pseudorandom sequences with that of ECCs. In later experiments, a carefully designed tracing algorithm will be employed.

We found that pseudorandom sequences provide comparable performances with ECCs having equal lengths, although the carefully designed ECCs guarantee fairly large minimum distances. The experiment shows that the minimum distance of the pseudorandom sequences is indeed smaller than that of ECCs (see Table 1). A basic question that may arise is "what is the theoretical minimum distance of the random (i.e., pseudorandom-sequence-based) codes?" Theorem 2 says that a random code will have a large minimum distance with high probability if the corresponding code length is long enough. In fact, Table 1 shows that the average distances of these two codes (ECCs and random codes) are very close. Theorem 2 and its proof follow:

Theorem 2

Suppose R is a random code with code length l, the distance between any two code words from R is large if the alphabet size q is large enough.

Proof

Let c_1 and c_2 be two code words randomly selected from R, and X_i be equal to 1 if $c_{1i}=c_{2i}$ (where c_{ui} is the ith digit

Table 1 Minimum distances and average distances of pseudorandom sequences and Reed-Solomon codes (in which q is the alphabet size).

			Pseudorandom Sequence		Reed-Solomon Code	
Code length	Dimension	q	Minimum distance	Average distance	Minimum distance	Average distance
15	3	16	8	14	13	13
31	2	32	23	30	30	30
63	2	64	55	62	62	62
127	2	128	119	126	126	126
255	2	256	246	254	254	254
511	2	512	503	510	510	510

of c_u). It is clear that $X_1, X_2, \ldots, X_l$ are l independent and identically distributed stochastic variables. Let $B(n,p)$ denote the binomial distribution with probability p for n trials, then X follows $B(l,p)$. Let μ be the expected value.

According to the Chernoff bound, for any $\delta > \mu$, $P(\Sigma_{i=1}^{i} X_i \geq l\delta) \leq 2^{-lD(\delta\|\mu)}$, where $D(p\|q)$ is the relative entropy between two probability distributions p and q. It follows that $P(X \geq l\delta) \leq 2^{-lD(\delta\|\mu)}$ for $\delta > \mu$. If $\varepsilon = 2^{-1D(\delta\|\mu)}$, we get $l = \log 1/\varepsilon / D(\delta\|\mu)$. Clearly, a small ε is expected, but this setting will lengthen l. Therefore, we hope to increase the denominator term $D(\delta\|\mu)$ to diminish the steep increase of l—that is, to increase the distance between δ and μ. Since the relative entropy function is always positive and convex, it can be drawn approximately as shown in Fig. 5. It should be noted that a small δ is expected (to guarantee a high probability of a large minimum distance), and $\delta > \mu$. Also, $\mu = q \cdot 1/q \cdot 1/q = 1/q$. Hence, it follows that if q is large enough [to ensure a large $D(\delta\|\mu)$], the probability that a random code R with moderate code-length l has a large minimum distance is high.

Table 2 presents some examples of relations among the collusion resilience c, the number of users n, and the code length l for some existing researches. If c or n increases to a more applicable value, l will become impractically long. Codes do not exist for some (n,c) pairs in the combinatorial-based methods (e.g., BIBD,[6] the TWWL method,[9] and SFPC[16]). In fact, it is not easy to find a combinatorial-designed code or an ECC that satisfies the required parameters. Besides, an orthogonal signals-based fingerprinting system can be broken by only a few dozen colluders.[17] Moreover, some of these methods fail to resist cut-and-paste attacks. A "cut-and-paste attack" is defined as an attack that is carried out by randomly selecting one of the ith symbols from all colluders' code words as the ith symbol of the pirate code. The combinatorial-designed codes will crash if the collusion-resilient information is cut out in a cut-and-paste attack.

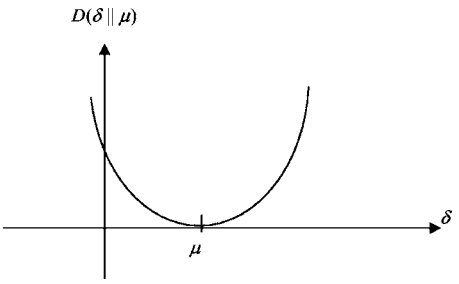

Fig. 5 Relationship between δ and the relative entropy $D(\delta\|\mu)$.

Since every method has its merits and shortcomings, the above-mentioned failings motivated us to combine different existing collusion-secure codes to construct a more applicable traceability code. In the following sections, we derive a new code by a concatenated construction strategy that can combine different types of collusion-secure codes and benefit from the concatenated structure.

2.4 *Concatenated Scheme of Traceability Codes*

One example of the concatenated constructed codes for fingerprinting is the logarithmic length collusion-secure code proposed by Boneh and Shaw, in which an ECC and a Γ_0 are concatenated to produce fingerprints.[1] Schaathun proved that the Boneh-Shaw scheme is more efficient than originally proven but proposed adaptations to further improve the scheme.[12] We will devise a more generalized framework for constructing the fingerprint codes based on concatenated schemes, which allows effective combinations of different collusion-secure codes.

The proposed concatenated scheme operates similar to the concatenated ECCs but provides greater ability to correct errors. A combination of the outer code and the inner code distributes the efforts to create a good code with applicable parameters. Based on the idea of concatenated construction strategy, a hierarchical scheme is then derived. The hierarchical scheme divides users into hierarchical groups. Groups are encoded by a collusion-secure code, and users in one group are encoded by another code. This scheme distributes the requirements for code length and tracing ability among groups.

The proposed concatenated traceability codes encode hi-

Table 2 Comparison of existing collusion-secure codes.

	BS (1995)[1]				Stinson (1998)[6]	SW (2002)[14]			TWWL (2003)[9]		TS (2005)[16]
Code type	Γ_0 (n-secure code with $\frac{1}{1024}$-error)	c-secure code with $\frac{1}{1024}$-error			t-designs	sequential c-TA code			AND-ACC		SFPC
c (some example)	2	2		3	2	2		3	2	3	2
n	1024	1024	10626	1024	10626	10626	262080	912576	7	20	1606
l	31228440620	76958	90295	458390	253	2420	24576	65863	code-length=7 l depends on the length of the orthogonal signals	code-length=16 l depends on the length of the orthogonal signals	377
Scalability of c	l grows abruptly as c increases.	Depends on the existence of the ECC used in the code construction.			Depends on the existence and construction of t-designs.	c is scalable when the required GF(q) exists.			Depends on the existence of BIBD and the orthogonal signals that are used.		Depends on the existence of the applied code.

erarchically and employ different collusion-secure codes flexibly. As a result, an expected high-performance code with small c or n can be constructed based on existing codes.

3 Concatenated Fingerprinting Scheme

As mentioned above, it is easier to find q-ary ($q>2$) codes having higher collusion-resiliency than binary ones. But in most real applications, embedding binary fingerprints is more applicable. However, transferring the q-ary symbols to binary ones directly risks a loss of the traceability properties after various collusion attacks. The q-ary fingerprinting scheme is not collusion-secure without appropriate watermarking strategies. For example, if user A with code word $(2\ 3\ 3\ 7)_8$ and user B with code word $(2\ 5\ 3\ 1)_8$ collude by the cut-and-paste attack, then the forged copy will obey the marking assumption. But if the octal symbols are transformed into binary ones before embedding, then a possible forged result $(2\ 1\ 3\ 5)_2$ that violates the marking assumption will cause the tracing algorithms to fail. Determining how to discriminate different symbols and how to identify all possible symbols involved in a collusion attack may be interesting problems to investigate. Some researchers have investigated methodologies of embedding q-ary fingerprints into multimedia assets.[18] Rather than finding the embedding strategy for q-ary fingerprints, we tried to derive a fingerprint construction method that could transfer the q-ary fingerprints into a binary form and then could be embedded in the content by many existing watermarking methods. Besides the need for a larger alphabet size for ECC-based codes, we can also use other types of collusion-secure codes (e.g., combinatorial-designed codes and orthogonal signals) to construct the traceability codes in a concatenated way, by which several low-performance codes can be combined to obtain a high-performance code. Details of the code construction will be addressed in the next section.

3.1 *Concatenated Construction Strategy of Traceability Codes*

Figure 6 shows the concatenated structure of the proposed fingerprinting scheme. The Boneh-Shaw scheme employs ECCs and Γ_0 as the outer code and inner code, respectively. The Schaathun scheme is generalized from the Boneh-Shaw scheme and demands the inner code to be strongly c-secure, but the outer code does not have to be collusion-secure itself. In the proposed concatenated scheme, each q-ary symbol of the outer code is encoded by another traceability code (i.e., the inner code). In other words, both the outer code and the inner code should be collusion-secure. The outer code must be collusion-secure because we treat the concatenated code as a q-ary collusion-secure code. The inner code also should be collusion-secure because our analysis is based on the strict marking assumption.

The outer code can be one of the existing q-ary c-secure codes with n code words, and the inner code can be a binary collusion-secure code with q code words. The produced outer code word is used to identify the colluders, and the inner code words are used to determine the symbols involved in the collusion for each position—that is, to guarantee the strict marking assumption. In the next section, we give analyses for both the code length and error rates.

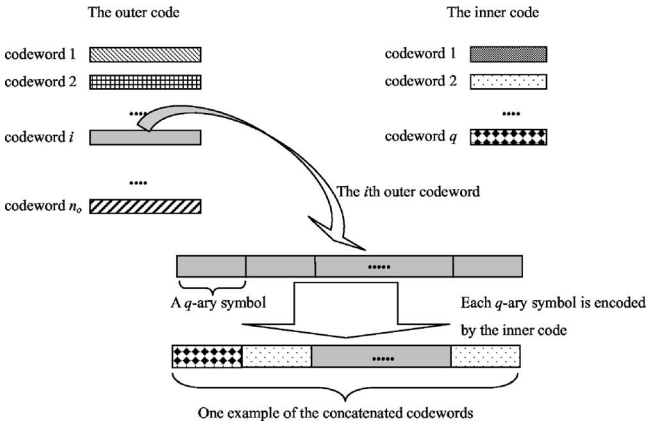

Fig. 6 Concatenated structure of the proposed traceability code construction scheme.

3.2 *Analysis of Concatenated Collusion-Secure Codes*

As mentioned above, the concatenated collusion-secure codes can be constructed by composing different types of codes. That is, the q-ary outer codes and binary inner codes can be combinatorial-designed codes, ECCs, or orthogonal signals. We analyze both the code length and the error rates, including type I (miss) and type II (false alarm) errors.

3.2.1 *Code length*

If linear ECCs are used for both inner codes and outer codes, the code length of the outer code will be $O(c^4 \log_q n/\log^2 c)$, according to the analysis of Ref. 11. For the inner code, n should be replaced by q and b replaced by 2, and the additional constraint is $c \leqslant q$, then the code length is $O(q^4 \log q/\log^2 q)$. Therefore, the total length of the concatenated code is $O(c^4 q^4 \log n/\log^2 c \log^2 q)$. Obviously, the extra length cost of the inner code must be paid for fulfilling the marking assumption, which depends on the alphabet size q.

3.2.2 *Type I errors*

Type I errors are introduced if the decoding algorithm does not output all the colluders, that is, some of the colluders are not caught. We will analyze the error rates for the ECC-ECC case (i.e., the combination of ECC and ECC for the outer code and inner code, respectively). Let $y_i, i \in C$ be the code words owned by the collusion coalition, and w be the pirate code word. If the minimum distance of the outer ECC is less than $l_{\text{outer}}(1-1/c^2)$, then the threshold T can be set to I_{outer}/c if the list decoding algorithm is applied:

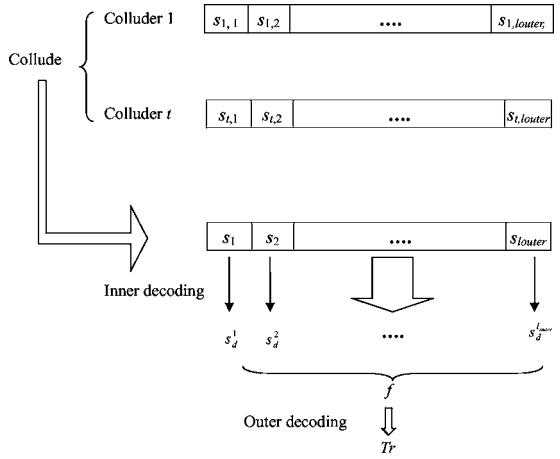

Fig. 7 Decoding procedure of the proposed concatenated traceability codes.

$$\begin{cases} \text{if } |I_{\text{outer}}(z,\omega)| \geq \dfrac{l_{\text{outer}}}{c}, \text{ then } z \text{ is one of the colluders} \\ \text{if } |I_{\text{outer}}(z,\omega)| < \dfrac{l_{\text{outer}}}{c}, \text{ then } z \text{ is an innocent user,} \end{cases} \tag{2}$$

in which $I_{\text{outer}}(x,y)$ is the identical function for the outer code $l(x,y)=\{i:x_i=y_i\}$, and l_{outer} is the code length of the outer code.

However, the decoding error of the inner code may cause the previous decision to be false. In addition, if the tradeoff between the miss rate and the false alarm rate is considered, the threshold T is not necessarily I_{outer}/c. Hypothesis testing can be applied to decide whether a user is one of the colluders if we use the list decoding method:

$$\begin{cases} H_0: |I_{\text{outer}}(z,w)| \geq T, & \text{if } z \text{ is one of the colluders,} \\ H_1: |I_{\text{outer}}(z,w)| < T, & \text{if } z \text{ is an innocent user.} \end{cases} \tag{3}$$

Suppose there are F decoding errors in the I_{outer} symbols after the inner decoding process. Then every colluder $y_i, i \in C$ has at least $l_{\text{outer}}-F/c$ identical positions with w. If $F \leq l_{\text{outer}}-c\cdot T$, then the traitors can be correctly traced. This means the miss rate $\varepsilon_{\text{miss}}$ is at most

$$\begin{aligned} \Pr(F > l_{\text{outer}} - c\cdot T) &\approx \Pr(F' > l_{\text{outer}} - c\cdot T) \\ &= \Pr[F' > (1-c\lambda)l_{\text{outer}}], \end{aligned}$$

where F' follows $B(l_{\text{outer}}, \varepsilon_{\text{inner}})$ (in which $\varepsilon_{\text{inner}}$ is the error rate of the inner decoding), and $\lambda = T/l_{\text{outer}}$. By the Chernoff bound, the following theorem is given without proof:

Theorem 3

$$\varepsilon_{\text{miss}} \leq 2^{-l_{\text{outer}} D(1-c\lambda \| \varepsilon_{\text{inner}})} \text{ if } \varepsilon_{\text{inner}} < 1 - c\lambda.$$

Obviously, λ is expected to be small enough to tolerate a large c, but this will introduce a higher type-II error because T increases with λ. This implies that the tradeoff between type I and type II errors depends on the selection of T.

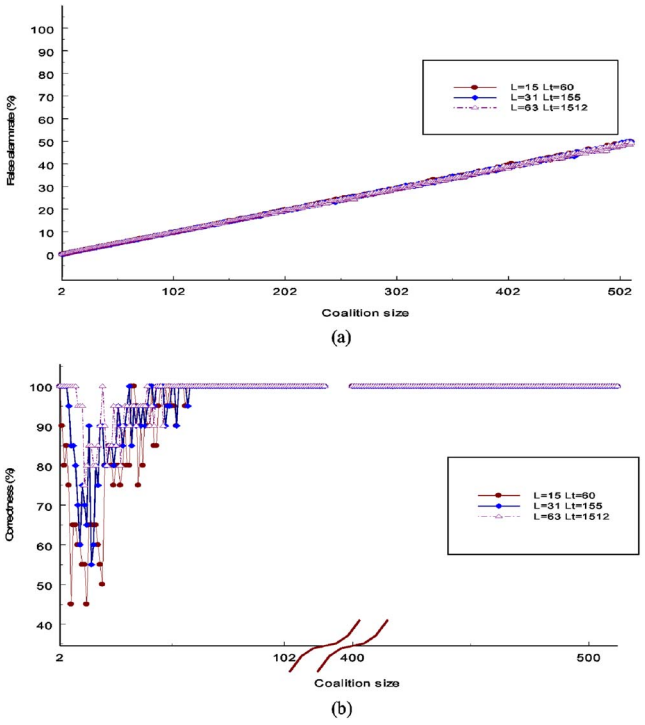

Fig. 8 Performances of the concatenated code: (a) false alarm rates, and (b) correctness. In (b), we cut the range of the coalition size from 120 to 400 because the correctness is constant in this range.

3.2.3 *Type II errors*

Type II errors occur when innocent users are framed, that is, the number of matches is at least T. Likewise, if the ECC-ECC case is given as an example again, type II error occurs when $\Pr[|I(z,w)| \geq T]$.

The l_{outer} positions are divided into two groups: the first group G_d is the set of positions where z is different from all the pirates, and the second group G_d^c is the complement set. Let X_i be the random variable that is 1 if and only if $z_i = w_i$, then $|I(z,w)| = \Sigma_{i\in G_d} X_i + \Sigma_{i\in G_d^c} X_i$. Recalling that we consider only c-TA ECCs so the minimum distance is larger than $l_{\text{outer}}(1-1/c^2)$, then we determine that z matches one of the pirates in at most l_{outer}/c^2 positions. It follows that $\Sigma_{i\in G_d'} X_i \leq |G_d^c| \leq l_{\text{outer}}/c^2 \cdot c$. Next, let $G_d' \subseteq G_d$ and $|G_d'| = l_{\text{outer}}(1-1/c)$. Clearly, if $z_i = w_i$ when $i \in G_d'$, then the ith position must have the decoding error, that is, when $i \in G_d$, X_i is 1 with the probability $\varepsilon_{\text{inner}}$. Then

$$\begin{aligned} \Pr[|I(z,w)| \geq T] &= \Pr[|I(z,w)| \geq \lambda \cdot l_{\text{outer}}] \\ &= \Pr\left[\left(\sum_{i\in G_d} X_i + \sum_{i\in G_d^c} X_i\right) \geq \lambda \cdot l_{\text{outer}}\right] \\ &\leq \Pr\left(\sum_{i\in G_d'} X_i + \frac{l_{\text{outer}}}{c} \geq \lambda \cdot l_{\text{outer}}\right) \\ &= \Pr\left[\sum_{i\in G_d'} X_i \geq \left(\lambda - \frac{1}{c}\right) l_{\text{outer}}\right] \end{aligned}$$

$$= \Pr\left[\sum_{i \in G'_d} X_i \geqslant \frac{(\lambda - 1/c) l_{\text{outer}} \cdot (1 - 1/c) l_{\text{outer}}}{(1 - 1/c) l_{\text{outer}}} \right]$$

$$= \Pr\left(\sum_{i \in G'_d} X_i \geqslant |G'_d| \cdot \frac{\lambda - 1/c}{1 - 1/c} \right).$$

The following theorem can be proved by using the Chernoff bound again:

Theorem 4

$$\varepsilon_{fa} < 2^{-a l_{\text{outer}} D(b/a \| \varepsilon_{\text{inner}})} \text{ if } \varepsilon_{\text{inner}} < \frac{b}{a},$$

where $a = 1 - 1/c$ and $b = \lambda - 1/c$

3.3 *Decoding Algorithm*

The following pseudocodes present the decoding procedure of the concatenated tractability code, in which **Inner_decode(·)** is a decoding algorithm for the inner code, **Concatenate(·)** means that each of the symbols in the parameter list is concatenated, and **Outer_decode(·)** is a decoding algorithm for the outer code.

Trace( )

{ for i=1 to l_{outer}

s_d^i:= **Inner_decode**(S_i)

(S_i is the ith symbol in the suspected code and s_d^i is the most possible collusion-symbol in the ith position)

end

f= **Concatenate**($s_d^1, \ldots, s_d^{l_{outer}}$)

Tr:= **Outer_decode**(f)

Output(Tr)

}

First, the q-ary symbols in each position in the outer code is decoded by the decoding algorithm of the inner code. The produced q-ary sequence is then fed into the outer decoding procedure of the outer code so at least one of the colluders can be identified. Figure 7 illustrates the decoding procedure.

3.4 *Examples of Concatenated Traceability Codes*

In this section, we give some examples to show the applicability of concatenated traceability codes.

Example 1

For a customer base of 64, we can find a binary (31,6,15)-Bose, Ray-Chaudhuri, Hocquenghem (BCH) code with code length 31. But the minimum distance of 15 is not guaranteed to have a corresponding c because $l(1 - 1/c^2)$ is at least 24 (when $c=2$). If a (15,3,11)-BCH code over the Galois Field (GF)(4) is used, the binary length is 30 (which is a little shorter than 31), and c can be set to 2. The 4-ary symbol can be encoded by any 2-secure code with the dimension size 4 (e.g., four orthogonal signals), and this operation will not lengthen the code too much (e.g., pseudorandom sequences of length 4). To prevent the colluders from interleaving their code segments to produce the pirate code, which would result in burst errors, random permutation should be applied to the fingerprints before the fingerprints are embedded in the content. The resulting concatenated code has a slightly longer code length and a much stronger tracing ability.

Example 2

If the size of the customer base is 16,807, we can use $(50,5,40)_7$-BCH as the outer code and (7,3,1)-BIBD (or even orthogonally modulated by Gaussian sequences as done in Ref. 9) as the inner code. Thus the total length is 350, which is much shorter than that of the 2-secure codes in Table 2 (this is because the need for large c and n is shared between the inner code and the outer code) and the outer code can resist copy-and-paste attacks.

Example 3

In order to show the practicality of our proposed, here we give the experimental result of a concatenated code as compared with pseudorandom sequences and ECCs (see Figs. 3, 4, and 8). The outer code is the RS code and the inner code consists of pseudorandom sequences. With a similar code length, the concatenated code has much greater correctness than the other two when the collusion size is smaller than 50.

4 Group-based Fingerprinting

In this section, we adapt the proposed concatenated traceability code to group-based fingerprinting. Group-based fingerprinting considers the characteristic that users can be divided into groups according to their regionalism. Reference 19 proposed a joint fingerprint design and distribution design by the tree-based fingerprint scheme to reduce the bandwidth requirement. In this design, users are grouped into a tree structure based on regionalism. In the group-oriented fingerprinting presented in Ref. 17, users who are likely to collude are assigned correlated fingerprints. In this section, we apply the proposed concatenated construction method to group-based fingerprinting.

As shown in Fig. 9, users are placed into several groups. The groups are encoded by outer code, and the users in each group are encoded by the inner code. If n users are divided into g groups and each group has m users, then $n = g \cdot m$. Therefore, a c-TA code for n users can be replaced by a c_{outer} TA code for g groups and a c_{inner}-TA code for m users. Note that c_{outer} can be small if users are grouped appropriately because the probability that colluders come from different groups is believed to be small. In addition, the problem of finding a collusion-secure code with large n and c can be simplified to find a c_{outer}-TA ($c_{\text{outer}} \ll c$) code for g ($g \ll n$) groups and a c_{inner}-TA code for m users ($c_{\text{inner}} \ll c, m \ll n$), which is much easier. Moreover, the extra bits paid for the inner code to ensure the marking assumption are now used to encode group members (the encoding efforts for all users are dispersed into two hierarchies, i.e., the group level and the member level), and hence will no longer be "extra." We propose two group-based fingerprinting schemes by the concatenated construction strategy, which will be detailed in Secs. 4.1 and 4.2, respectively.

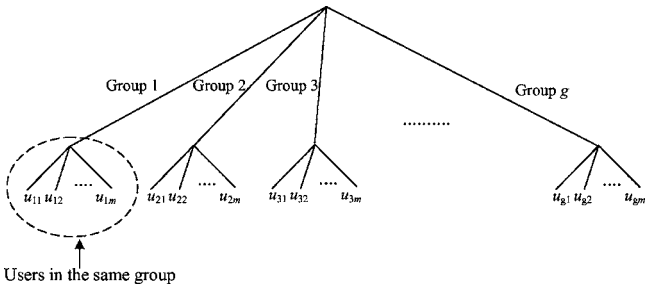

Fig. 9 Structure of the group-based fingerprinting.

4.1 *Scheme 1: Group-Oriented Traceability Codes*

According to the code construction method of the concatenated traceability code, the outer code encodes the groups and the inner code encodes the group members for the group-oriented traceability codes. That is, the code word number of the q-ary outer code reduces to g and the collusion resiliency is still c. But how are the m group members encoded? We encode the q-ary symbols of the outer code as well as the m group members by another code (i.e., the inner code) with $q \cdot m$ code words. Figure 10 shows one example of the group-oriented traceability codes. For each of the q symbols in the alphabet, m code words are needed to represent m members in one group. The inner code consequently has $q \cdot m$ code words. In addition, the information for the member number reduplicates l_{outer} times in this scheme because every one of the l_{outer} symbols in the outer code word contains the member information. Although it will produce additional cost to encode the member number, it really enhances the robustness of the inner code.

The octal outer code
Alphabet: {0,1,2,3,4,5,6,7}

Group #	Codeword
1	4 2 0 … 6
2	7 0 3 … 2
…	…
g	1 0 4 … 1

(a)

The binary inner code
Alphabet: {0,1}
Number of codewords: $8m$

m_i \ s_i	0	1	2	3	4	5	6	7
1	100...1	101…1	000…1	010…1	101…1	001…1	010…0	011…1
2	000…0	111…1	010…0	101…0	110…1	010…0	101…1	000…1
....								
m	111…0	000…1	101…0	000…1	010…0	101…1	000…1	000…1

(b)

Fig. 10 One example of the group-oriented traceability codes: (a) the octal outer code, and (b) the binary inner code.

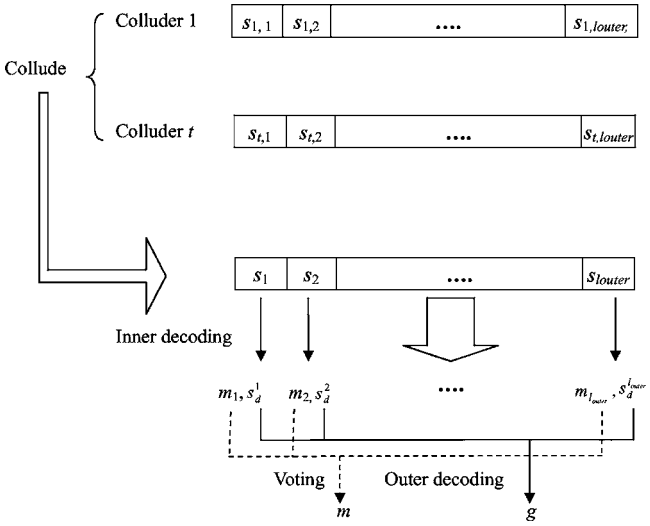

Fig. 11 Decoding procedure of the proposed group-oriented traceability codes.

To avoid the collusion attack by comparing fingerprints among members in the same group, an additional random permutation should be made for symbols of the outer code before embedding fingerprints into the content.

4.1.1 *Tracing algorithm*

The following pseudocodes present the decoding procedure of the group-oriented traceability code, in which **Inner_decode(·)**, **Concatenate(·)**, and **Outer_decode(·)** are defined the same as before, g is determined by concatenating the symbol number s_i, $i \in [1, l_{\text{outer}}]$, and **Voting(·)** is the policy to decide the member number according to $m_i s$ obtained from **Inner_decode(·)**.

Trace( )

{for i=1 to l_{outer}

$(m_i, s_d^i) :=$ **Inner_decode**(S_i)

(S_i is the ith symbol in the suspected code and s_d^i is the most possible collusion symbol in the ith position)

end

$g :=$ **Outer_decode** (**Concatenate**$(s_d^1, \ldots, s_d^{l_{outer}})$)

$m :=$ **Voting**$(m_i, i \in [1, l_{outer}])$

Output(g, m)

}

Figure 11 shows the decoding procedure of the group-oriented traceability code.

4.1.2 *Analysis of group-oriented traceability codes*

In this section we suppose linear ECCs are used again. Then the code length of the naïve code is

$$l = O\left(\frac{c^4 \log_q n}{\log^2 c}\right) = O\left[\frac{c^4(\log g + \log m)}{\log q \log^2 c}\right]^{g \geqslant c, m \geqslant c}$$
$$= O\left[\frac{g^4 m^4 (\log g + \log m)}{\log q \log^2 c}\right], \quad (4)$$

and the code length of the group-oriented traceability code is

$$l_{\text{new}} = l_g \cdot l_m$$
$$= O\left[\frac{\min(c,g)^4 \log_q g}{\log^2 \min(c,g)}\right] \cdot O\left[\frac{\min(c,m)^4 \log_2 qm}{\log^2 \min(c,m)}\right]^{g \geqslant c, m \geqslant c}$$
$$= O\left[\frac{g^4 m^4 \log g \log qm}{\log^2 g \log^2 m \log q}\right]. \quad (5)$$

It should be noted that Eqs. (4) and (5) are derived under the assumption that $g \geqslant c$ and $m \geqslant c$, thus, $l_{\text{new}} \leqslant l$ can be proved by supposing $\log qm \cong \log gm$. That is, the code length of the group-oriented code is shorter than that of the naïve code if c is not too large, and g and m are properly assigned. In fact, the conditions $g \geqslant c$ and $m \geqslant c$ are reasonable in real applications.

4.1.3 *Experimental results*

In the experiments, 1024 users are grouped into eight groups so each group has 128 members. A $(15,3,13)_{16}$-RS code is adopted as the outer code, and 20-bit pseudorandom sequences are used as the inner code. According to the criteria of minimum distance for c-TA codes, $d > l(1 - 1/c^2)$, the distance ratio 13/15 of the outer $(15,3,13)_{16}$-RS code can resist two collusion groups and even approximate to three groups, which is enough for all eight groups. The 20-bit inner code is used to encode 16 symbols of the alphabet set and 128 members of a group, i.e., the alphabet size of the inner code is $16 \cdot 128 = 2^{11}$.

We conducted two experiments: the first one tests the traceability directly on the code words, and the second applies the proposed group-oriented scheme to image fingerprinting. In the second experiment, the 256×256 Lena image is used as the test data. The fingerprint code is first produced by the construction method described above. Then we embed the fingerprint vector $(v_0, v_1, \ldots, v_{l-1})$ into the coefficients of 8×8 discrete cosine transform (DCT) blocks of the image by

$$\begin{cases} \tilde{x}_{k,b} = x_{k,b-1} + \alpha \cdot x_{0,b}, & \text{if } v_i = 1 \\ \tilde{x}_{k,b} = x_{k,b-1} - \alpha \cdot x_{0,b}, & \text{if } v_i = 0, \end{cases} \quad (6)$$

where $x_{0,b}$ and $x_{k,b}$, respectively, are the DC and the kth AC coefficients of the bth DCT block, $k \in S, S \subseteq \{1,2,\ldots,63\}$ is the set of indices of DCT coefficients selected for watermark embedding, and α is the watermark strength.

The fingerprint can be extracted by simply comparing the values between $\tilde{x}_{k,b}$ and $\tilde{x}_{k,b-1}$. The extracted watermark value is

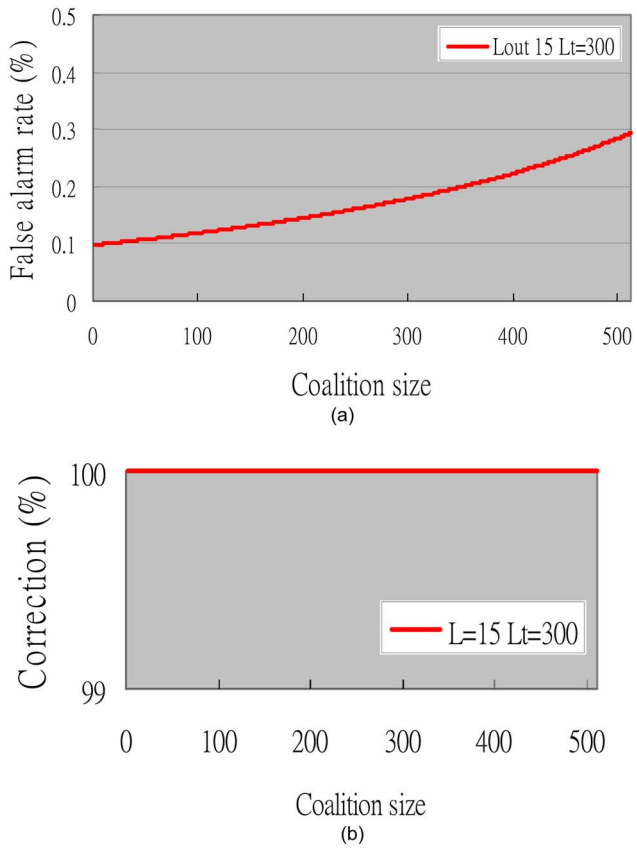

Fig. 12 Performances of the group-oriented scheme applied directly to code words: (a) false alarm rates, and (b) correctness. The false alarm rate increases smoothly and the correctness is 100%.

$$\tilde{v}_i = \begin{cases} 1, & \text{if } \tilde{x}_{k,b} > \tilde{x}_{k,b-1} \\ 0, & \text{if } \tilde{x}_{k,b} < \tilde{x}_{k,b-1} \\ ?, & \text{otherwise} \end{cases}. \quad (7)$$

If the extracted value is "?," we decode it as an erasure, which should be tolerated by the traitor detection procedure. The extracted fingerprint $(\tilde{v}_0, \tilde{v}_1, \ldots, \tilde{v}_{l-1})$ is then decoded into one of the code words of the traceability code by computing the corresponding Hamming distances.

The average operation is adopted as the collusion attack for both experiments. Figures 12 and 13 show the performances of the proposed group-oriented fingerprinting scheme, in which the experiments are applied directly to the code words and to the Lena image, respectively. In both of the experiments, the false alarm rates are lower than 0.3%, and the correctness is 100%. Compared with the codes that do not have the group-oriented design, e.g., the pseudorandom sequences and the RS code (whose performances are presented in Figs. 3 and 4, respectively), the proposed design strategy is obviously superior for both the false alarm rate and correctness. Moreover, the code length of 300 is much shorter than the existing fingerprinting codes. Table 3 gives the detail description of the code.

Although in our implementation we applied the proposed hierarchical traceability code only to image fingerprinting, the proposed technology can be applied to other types of media in a similar way.

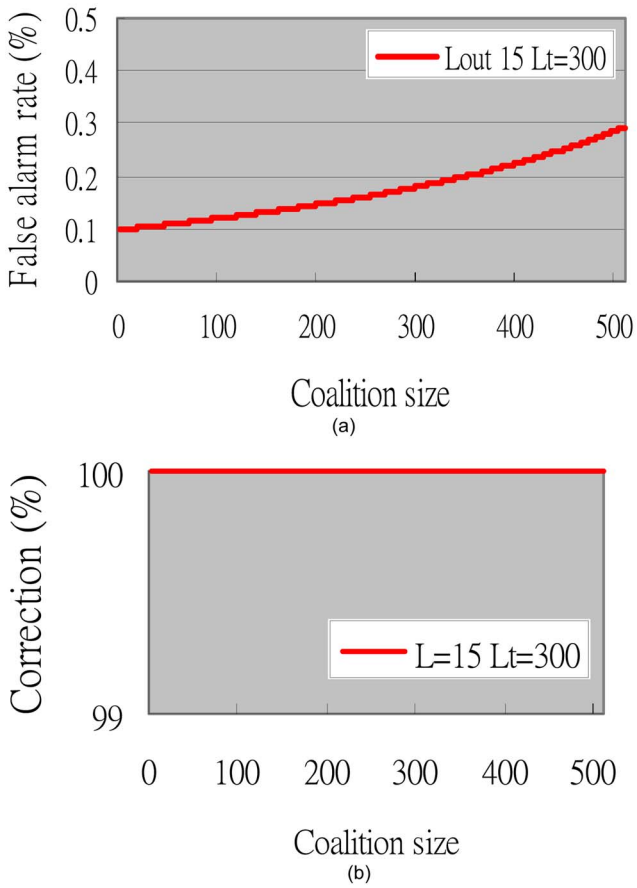

Fig. 13 Performances of the group-oriented scheme applied to the 256×256 Lena image: (a) false alarm rates, and (b) correctness. The results are similar to the experiments performed directly on code words. The false alarm rate increases smoothly and the correctness is 100%.

4.2 *Scheme 2: Hierarchical Traceability Codes*

The group-oriented codes in Sec. 4.1 are constructed in a nested way, where the coding effort is dispersed to the group and membership levels. In this section, we employ the same multilevel structure to derive the traceability code by a hierarchical construction strategy. Further analyses and interpretations can be found in Ref. 20. We present this hierarchical traceability code here to show the efficiency of group-based fingerprinting.

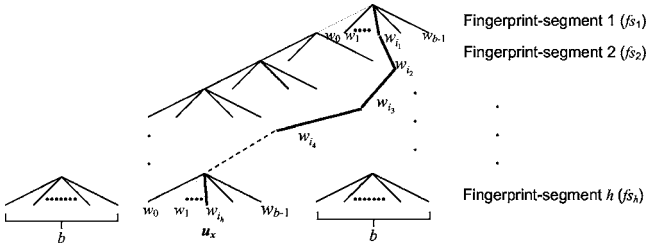

Fig. 14 Structure of the proposed hierarchical traceability codes.

4.2.1 *Hierarchical construction strategy of traceability codes*

Segment codes. Figure 14 shows the structure of the proposed hierarchical traceability code. The traceability code is divided into several hierarchies. Each hierarchy represents a segment of the fingerprint code. Let b be the number of branches of the code tree, n be the number of users, and l be the length of the code. To encode the fingerprint segment for each hierarchy, the segment code needs b code words, one for each branch. The fingerprint is then the concatenation of fingerprint segments of all hierarchies, i.e., $f_{s1}f_{s2}\ldots f_{sh}$. For example, if the segment code has code words $w_0, w_1, \ldots, w_{b-1}$, then the fingerprint of u_x is constructed by concatenating each one of the segment codewords: $w_{i_1}, w_{i_2}, w_{i_3}, \ldots, w_{i_h}$, where w_{i_k} is the code word of the kth hierarchy for u_x, and $\forall k \in (1,h), w_{i_k} \in \{w_0, w_1, \ldots, w_{b-1}\}$ (as illustrated in Fig. 14).

But in the jth level of the tree, all b^j groups of branches are encoded by the same code, i.e., all groups have the same code words. For example, if w_i is a code word of the segment code, then all groups have the code word w_i. The question is, how can one w_i be discriminated from other w_is in different groups at the same level? Obviously extra bits

Table 3 Examples of codes that are constructed by the proposed concatenated fingerprinting schemes.

		Group-based Fingerprinting Codes	
Construction Method	$(50,5,40)_7$-BCH +(7,3,1)-BIBD	Group-oriented code	Hierarchical code
c (some examples)	2	2 group	Near 1024
n	16807	1024	1024
l	350	300	4320
Scalability of c	It is easier to increase c than the naive method without concatenated construction.	The scalability of c is good	The scalability of c is good.

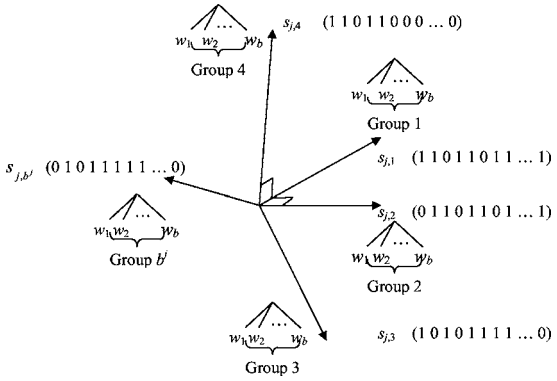

Fig. 15 b^j orthogonal signals used to encode b^j groups in the jth level of the tree for the proposed hierarchical fingerprinting scheme. These pseudorandom signals are nearly orthogonal to one another.

must be paid for encoding the groups. The additional code length depends on the group code (i.e., another collusion-secure code employed for encoding the groups) used, and the group code can vary in different levels. In the next subsection, we will explain how to discriminate groups by orthogonal signals and construct hierarchical fingerprint codes using existing collusion-secure codes as the segment code.

Group codes. Figure 15 shows how to discriminate different groups in the hierarchical tree: by employing b^j nearly orthogonal signals to modulate b^j groups in the jth level of the tree. The pseudorandom modulation is employed because it can avert extra bits and efforts to design the group code. That is, we pseudorandomly modulate the segment code words in each of the b^j groups of the jth level by a different random sequence s_{jg} according to the corresponding group key so different groups are orthogonal to each other. Then the pseudorandom signals become our group code. Thus, each of the groups has the same segment code, and the segment code word w_i of group j_1 can be discriminated from that of group j_2 because different modulating sequences are used. Equation (8) states the pseudorandom modulation:

$$\tilde{w}_{j,g,i} = w_i \oplus s_{j,g} = w_i \oplus P(K_{j,g}), \tag{8}$$

where w_i is the code word vector of the ith branch, $\oplus$ is the exclusive-OR operation, $P(\cdot)$ is a pseudorandom-sequence generator, $K_{j,g}$ is the group key of the gth group at the jth level, and $\tilde{w}_{j,g,i}$ is the pseudorandomly modulated code word for w_i in the gth group at the jth level. That is, groups are orthogonal to each other after the pseudorandom modulation. The hazard illustrated in the previous subsection will not occur because w_i is modulated by different pseudorandom sequences for different groups.[20] The decoding method and the tracing algorithm are detailed in Ref. 20.

4.2.2 *Experimental results*

In the implementation, we construct a 4-ary tree (i.e., b is 4) and use the 4-secure code with 0.001-error $\Gamma_0(4,288)$ to encode every two branches. The resulting length of the code segment is 864, the total length of five hierarchies for encoding 1024 code words is 4320, and the theoretical value of error rate ε is $1-(1-0.001)^5=0.01995$. The same two experiments as in scheme 1 are conducted.

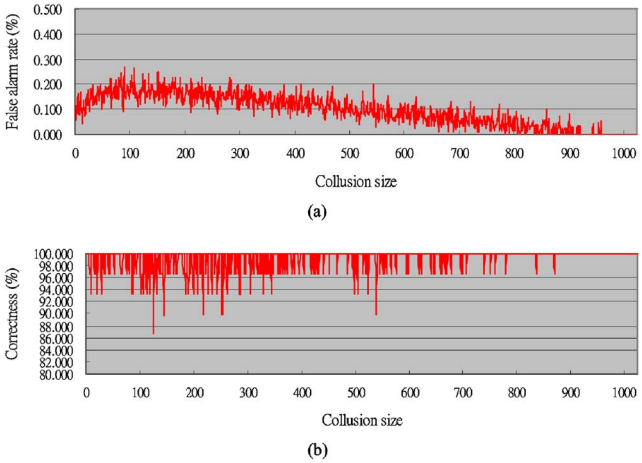

Fig. 16 Performances of the traitor tracing for hierarchical fingerprinting scheme applied directly to code words: (a) false alarm rates, and (b) correctness.

Figures 16 and 17 illustrate the false alarm rates and the correctness of the traitor tracing. In the first experiment, rather promising results are obtained. Almost all of the correctness is larger than 90%, and the false alarm rates are lower than 0.3%, even if the collusion size is large. Even though the performance of the second experiment is not as good as that of the first one, we can refine it by adaptively adjusting the thresholds when calculating the Hamming distances in the list decoding for the segment code, or we can improve the watermarking approach. Moreover, replacing the 4-TA code (i.e., the segment code) by a b-TA code with larger b will improve the detection correctness, because the accumulated error is reduced (but the cost is the

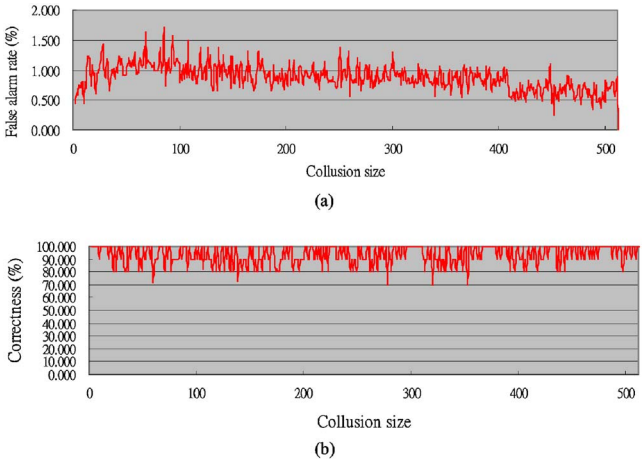

Fig. 17 Performances of the traitor tracing for the hierarchical fingerprinting scheme applied to the 256×256 Lena image: (a) false alarm rates, and (b) correctness.

length of the segment code). In real applications, the branch size can be varied and different for every hierarchy.

The most important aspect is that the code length of the naive encoding method, without using the tree structure, is 31228440620, which is much longer than the 4320 required in the proposed approach. And the collusion resiliency c increases from 2 to close to 1024. For ease of comparison with other methods, we list parameters of the code in Table 3.

5 Conclusions

In this work, a general framework of practical traceability code construction for multimedia fingerprinting has been proposed. The concatenated scheme provides a flexible approach to combine different traceability codes for constructing a traceability code with more applicable parameters. In current researches, it is not easy to find a traceability code having practical parameters. Instead of investigating one ideal traceability code, we reached our goal by composing several different existing codes that may not have had practical parameters initially. The concatenated construction method produces traceability codes by combining the outer code and the inner code together. In the outer code word, q-ary symbols are encoded by a collusion-secure code to preserve the traceability property. After collusion attacks, each symbol can still be reconstructed because of the error-correction ability or traceability of the inner code. And further, the concatenated code can shorten the traceability codes with an applicable n. Traitors can then be identified by decoding the outer collusion-secure code.

We have also proposed two concatenated construction schemes of the traceability codes base don group fingerprinting: the group-oriented traceability scheme, and the hierarchical traceability scheme. The group-oriented traceability scheme encodes the groups by the outer code and the members by the inner code. The problem of finding a collusion-secure code with large n and c then becomes finding two codes for encoding g groups and m members, respectively, which is much easier. Besides, the code length can be shortened by properly grouping the users. The hierarchical construction strategy effectively shortens the traceability code and results in a much larger collusion resiliency without influencing the detection correctness too much. Traitors can be identified with high correctness by a fast-tracing algorithm. The tree structure can also be used to group the users according to regionalism to avoid collusion attacks.

The proposed concatenated construction methods of traceability codes can be applied appropriately to multimedia content. It also provides a general methodology to share the requirements for the code length and the collusion resiliency among q-ary symbols or groups of users based on composing existing codes. As a result, the larger c, n, and shorter l make the fingerprinting system more practical. We can replace the employed traceability codes with other suitable codes according to various applications.

Acknowledgments

This work was partially supported by the National Science Council and the Ministry of Education of China under contact nos. NSC92-2622-E-002-002, NSC92-2213-E-002-023, 89E-FA06-2-4-8, and NSC93-2752-E-002-006-PAE. We would also like to acknowledge Dr. Hans Georg Schaathun for his helpful discussions and valuable suggestions.

References

1. D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *Proc. CRYPTO'95*, pp. 452–465 (1995).
2. B. Chor, A. Fiat, and M. Naor, "Tracing traitors," *Proc. CRYPTO'94*, Vol. **839**, pp. 480–491 (1994).
3. A. Fiat and M. Naor, "Broadcast encryption," *Proc. CRYPTO'93*, pp. 480–491 (1993).
4. D. R. Stinson and T. V. Trung, "Some new results on key distribution patterns and broadcast encryption," *Designs, Codes, Cryptogr.* **14**, pp. 261–279 (1998).
5. D. R. Stinson, "On some methods for unconditionally secure key distribution and broadcast encryption," *Designs, Codes, Cryptogr.* **12**, pp. 215–243 (1997).
6. D. R. Stinson and R. Wei, "Combinatorial properties and constructions of traceability schemes and frameproof codes," *SIAM J. Discrete Math.* **11**, 41–53 (1998).
7. J. Dittmann, P. Schmitt, E. Saar, and J. Ueberberg, "Combining digital watermarks and collusion secure fingerprints for digital images," *J. Electron. Imaging* **9**(4), 456–467 (2000).
8. Z. J. Wang, M. Wu, H. Zhao, and K. J. R. Liu, "Resistance of orthogonal Gaussian fingerprints to collusion attacks," *ICASSP 2003*, pp. IV-724–IV-727 (2003).
9. W. Trappe, M. Wu, J. Wang, and K. J. R. Liu, "Anti-collusion fingerprinting for multimedia," *IEEE Trans. Signal Process.* **51**, 1069–1087 (2003).
10. http://www.maths.qmul.ac.uk/~pjc/preprints/design.pdf
11. A. Silverberg, J. Staddon, and J. L. Walker, "Applications of list decoding to tracing traitors," *IEEE Trans. Inf. Theory* **49**, 1312–1318 (2003).
12. H. G. Schaathun, "The Boneh-Shaw fingerprinting scheme is better than we thought," *IEEE Trans. Information Forensics and Security* **1**(2), 248–255 (2006).
13. C. J. Colbourn and J. H. Dinitz, *The CRC Handbook of Combinatorial Designs*, pp. 3–87.
14. R. Safavi-Naini and Y. Wang, "A code for sequential traitor tracing," *ASPCS*, pp. 211–224 (2002).
15. I. Cox, J. Kilian, F. Leighton, and T. Shamoon, "Secure spectrum watermarking for multimedia," *IEEE Trans. Image Process.* **6**(12), 1673–1687 (Dec. 1997).
16. D. Tonien and R. Safavi-Naini, "Explicit construction of secure frameproof codes," *Cryptology ePrint Archive 2005/275*.
17. Z. J. Wang, M. Wu, W. Trappe, and K. J. R. Liu, "Group-oriented fingerprinting for multimedia forensics," *EURASIP*, pp. 2153–2173 (2004).
18. R. Safavi-Naini and Y. Wang, "Collusion secure q-ary fingerprinting for perceptual content," *DRM 2001, LNCS 2320*, pp. 57–75 (2002).
19. H. Zhao and K. J. Liu, "A secure multicast scheme for anti-collusion fingerprinted video," *IEEE Global Telecommunications Conference 2004*, pp. 571–575 (2004).
20. Y. T. Lin and J. L. Wu, "Practical fingerprinting system for Images," *Opt. Eng.* **46**(5), 057004 (2007).

Yu-Tzu Lin received the BS and MS degrees in information and computer education from National Taiwan Normal University, Taipei, Taiwan, in 1994 and 1997. She is currently pursuing the PhD degree in the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan. Her research interests include copyright protection for multimedia, pattern recognition, and image processing.

Ja-Ling Wu received the BS degree in electronic engineering from TamKang University, Tamshoei, Taiwan in 1979, and the MS and PhD degrees in electrical engineering from Tatung Institute of Technology, Taipei, Taiwan, in 1981 and 1986. Since 1987, he has been with the Department of Computer Science and Information Engineering, National Taiwan University, where he is presently a professor. He has published more than 200 journal and conference papers. His research interests include algorithm design for DSP, data compression, digital watermarking, and multimedia systems. Professor Wu was the recipient of the Excellent Research Award from NSC, Taiwan, in 1999, 2001, and 2004: He was elected to be one of the lifetime Distinguished Professors, National Taiwan University, in November 2006.

Chun-Hsiang Huang received the BS, MS, and PhD degrees in computer science and information engineering from National Taiwan University (NTU), Taipei, Taiwan, in 1997, 1999, and 2005, respectively. He is currently a postdoctoral research associate at the Communications and Multimedia Laboratory, Department of Computer Science and Information Engineering, NTU. His current research interests include digital watermarking, image processing, and information theory.