Blind Pattern Matching Attack on Watermarking Systems

Darko Kirovski and Fabien A. P. Petitcolas

***Abstract*— Billions of dollars allegedly lost to piracy of multimedia content have recently triggered the industry to rethink the way music and films are distributed on the Internet. As encryption is vulnerable to digital or analog re-recording, currently almost all copyright protection mechanisms rely to a certain extent on watermarking. A watermark is an imperceptive secret hidden into a host signal. In this paper, we analyze the security of multimedia copyright protection systems that use watermarks, by proposing a new breed of attacks on generic watermarking systems. A typical blind pattern matching attack relies upon the observation that multimedia content is often highly repetitive. Thus, the attack procedure identifies subsets of signal blocks that are similar and permutes these blocks. Assuming the permuted blocks are marked with distinct secrets, it can be shown that any watermark detector is facing a task of exponential complexity to reverse the permutations as a pre-processing step for watermark detection. In this paper, we describe the logistics of the attack and an implementation against a spread-spectrum and a quantization index modulation data hiding technology for audio signals.**

***Index Terms*— watermarking attacks, blind pattern matching, spread-spectrum watermarking, quantization index modulation.**

I. Introduction

Significantly increased levels of multimedia piracy over the last decade have put the film and music industry under pressure to develop and deploy as a standard improved anti-piracy technology that can enforce copyright protection on client media players and hence cut down the number of downloads on peer-to-peer file sharing services such as Napster – which, alone, has orchestrated almost 3 billion downloads of sound clips in February 2001. Several industry-wide initiatives have had little success in enabling client-hosted copyright screening mechanisms [1], [2].

The problem of ensuring copyright at the client side lies in the fact that traditional data protection technologies such as encryption or scrambling cannot be applied as they are prone to digital or analogue re-recording (copying). Thus, almost all modern copyright protection mechanisms rely to a certain extent on *watermarks*, imperceptive marks hidden in host signals. In a typical content screening system, the client's media player searches the content for hidden information. If the secret mark is found, the player must verify, prior to playback, whether it has a license to play the content. By default, unmarked content is considered as unprotected and is played without any barriers. A key technology required for content screening is public-key watermarking, that is, a marking scheme where breaking a single player or a subset of players does *not* compromise the security of the entire system. A public-key watermarking system, potentially efficient for content screening, has been detailed in [3].

If breaking a single player does not pose a security threat, the main target of the adversary is finding a signal processing primitive that removes the watermark or prevents a detector to find it. Several attack mechanisms have been largely successful in setting up robustness benchmarks for watermarking technologies. In fact, as soon as people have tried to develop watermarking technologies, others have attempted to break them. Early attacks, such as random geometric distortions [4] relied on the fact that most watermarking algorithms are based on some form of correlation, which itself requires good alignment properties. Breaking this alignment usually prevents reliable detection. In fact there are two main types of attacks: those who attempt to remove the watermark and those who just prevent the detector from detecting them. Random geometric distortions fall in the second category.

Attacks in the first category usually try to estimate the original non-watermarked cover-signal, considering the watermark as noise with given statistic. For instance, Langelaar et al. showed that 3×3 median filtering gives a good approximation of original pictures in the case they have been watermarked using spread-spectrum [5]. So far, estimate-and-remove attacks have introduced fairly strong blurring effects but recent work based on maximum a posteriori watermark estimation and re-modulation has given promising results [6], [7]. In the case of fingerprinting, another way to remove the watermark is to use copies from different sources and mix them (either by averaging them or concatenating pieces of them like a Mosaïc attack [4]) to generate an un-watermarked copy. These are usually referred to as collusion attacks [8], [9].

Attacks particularly tailored to specific audio watermarking technologies include echo removal or signal restoration [10]. But these attacks are not very general as they can only be applied to certain watermarking algorithms. An obvious improvement for audio attacks is the equivalent of random geometric distortions. This is in fact a mixture of time and frequency scaling. Although robustness to this type of attack is not fully solved in the case of images, there are some audio data hiding algorithms that can cope with them already [11].

In this manuscript, we propose an attack which aims at reducing the correlation of a watermarked signal with its watermark by replacing blocks of samples of the marked signal with perceptually similar blocks that are either not marked or that are marked with a different watermark. We call this type of an attack: *a blind pattern matching attack* (BPM). In some

D. Kirovski is with Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA. F.A.P. Petitcolas is with Microsoft Research, 7 J. J. Thomson Avenue, Cambridge CB3 0FB, England. E-mail: {darkok, fabienpe}@microsoft.com.

sense it is a form of a collusion attack but very different from the ones mentioned above. BPM is a new paradigm of attacks against generic watermarking systems. The strategy of a BPM attack is simple:

1	partition the content into overlapping low-granularity signal blocks,
2	identify subsets of perceptually similar blocks, and
3	randomly permute their locations in the signal.

If the number of blocks that have perceptually similar counterparts within the media clip is small, for a successful launch of the BPM attack, the adversary can alternatively seek replacement blocks in an external multimedia library. The hope is that a large percentage of the original signal can be replaced or perturbed such that watermark detection becomes nearly impossible. Even in the restricted case when the adversary does not use an external replacement library and the permuted blocks are marked with distinct secrets, it is straightforward to prove that any watermark detector faces a task of exponential complexity to reverse the permutations as a pre-processing step for correct watermark detection.

Finding perceptually similar blocks of certain music or video content is a challenging task. With no loss of generality, in this paper we restrict our focus to audio, although video is in many cases a much better source of repetitive content within a single recording. For example, within a common scene both background and objects experience geometric transformations significantly more frequently than changes in appearance. In general, repetition is often a principal part of composing music and is a natural consequence of the fact that distinct instruments, voices and tones are used to create a soundtrack. Thus, it is likely to find similarities within a single musical piece, an album of songs from a single author or in instrument solos. In this paper, we explore the challenges of the BPM attack and show how it can be launched on audio content.

II. Generic BPM attack

The BPM attack is not limited to a type of content or to a particular watermarking algorithm. For example, systems that modulate secrets using spread-spectrum [12] and/or quantization index modulation (QIM) [13] are all prone to the BPM attack. In order to launch the attack successfully, the adversary does not need to know the details of the watermark codec. The adversary needs to reduce the granularity of integral blocks of data such that no block contains enough information from which a watermark can be identified individually. Note that watermark detection involves processing large amount of data (for example, reliable and robust detection of audio watermarks requires at least several seconds of audio [11]). Thus, blocks considered for BPM must be at least one order of magnitude smaller than watermark length. For both audio and video, this requirement is not difficult to satisfy as typically blocks of 128–$1,024$ transform coefficients for audio or bitmaps of up to 64×64 pixels for video are considered for pattern matching.

In the remainder of this section, we assume that coefficients of the marked signal are replaced only with other coefficients of the same signal. It is straightforward to redefine the attack such that coefficients from external signal vectors are considered as a substitution base.

The *host signal* to be marked $\mathbf{x} \in \mathcal{R}^N$ can be modelled as a vector, where each element $x_i \in \mathbf{x}$ is a zero-mean independent identically distributed normal random variable with standard deviation σ_x: $x_i \sim \mathcal{N}(0, \sigma_x)$. The BPM attack is not restricted to a particular signal model. However, we use the Gaussian assumption to analyze certain properties of the attack.

A *watermark* is defined as an arbitrary pseudo-randomly generated vector $\mathbf{w} \in \mathcal{R}^N$, where each element $w_i \in \mathbf{w}$ is a random variable with standard deviation $\delta \ll \sigma_x$. For example, if direct sequence spread-spectrum is used for watermark modulation then $\mathbf{w} \in \{\pm\delta\}^N$. We assume that the watermark signal $\mathbf{w}$ is mutually independent with respect to $\mathbf{x}$. The *marked signal* $\tilde{\mathbf{x}}$ is created as $\tilde{\mathbf{x}} = \mathbf{x} + \mathbf{w}$.

The BPM attack algorithm receives as input the marked signal $\tilde{\mathbf{x}}$ and outputs its modification $\tilde{\mathbf{x}}'$.

A. Attack steps

The algorithm has several steps which are described below.

1) Signal partitioning: In this step, the watermarked content $\tilde{\mathbf{x}}$ is partitioned into a set $\mathcal{B}$ of overlapping blocks, where each block B_p represents a sequence of m samples of $\tilde{\mathbf{x}}$ starting at $\tilde{x}_{(B_p)}$. (B_i) denotes the index of the first sample in the i-th block B_i. For an overlap ratio of η, the total number of blocks equals $n = \lceil \frac{N-m}{1-\eta} \rceil$. The higher the overlap, the larger the search-space for the BPM attack. We want to select the overlap such that:

1) consecutive blocks do not have similar perceptual characteristics – this upper bound on block overlap aims at reducing the search space – and
2) for two consecutive blocks B_p and B_{p+1} starting at $\tilde{x}_{(B_p)}$ and $\tilde{x}_{(B_{p+1})}$ respectively, the block starting at $\tilde{x}_a$, $a = [(B_p) + (B_{p+1})]/2$ is not perceptually similar to B_p or B_{p+1}.

2) Similarity function: This is the core function of the BPM attack. It takes as an input a pair of blocks B_p and B_q and returns a real number $\phi(B_p, B_q) \geq 0$ that quantifies their similarity. Block equality is represented as $\phi(B_p, B_q) = 0$. The adversary can experiment with a number of different functions. In this section, we restrict similarity to the quadratic euclidian distance between blocks:

$$\phi(B_p, B_q) \equiv \phi(B_p, B_q)^2 = \sum_{i=0}^{m-1} \left[y_{i+(B_p)} - y_{i+(B_q)} \right]^2, \quad \phi(B_p, B_q) \geqslant 0. \tag{1}$$

3) Pattern matching: In this step, perceptual similarities between individual signal blocks are identified. The result of this phase is a symmetric similarity bit-matrix S, defined as follows:

$$S_{p,q} = \begin{cases} 1 & \text{if } m\alpha^2 \leqslant \phi(B_p, B_q)^2 \leqslant m\beta^2 \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where α^2 and β^2 are parameters that denote the minimal and maximal average similarity respectively for a pair of blocks to be considered as substitutable. The lower limit is required because substituting a block with another of exceptional similarity does not affect watermark detection. The upper limit ensures that the resulting clip has a preserved high fidelity with respect to the marked copy.

We compute the S matrix in the following way. Initially, we compute the block-level auto-covariance of the input watermarked signal $\tilde{\mathbf{x}}$. Then, for all pairs of blocks that correspond to a high value of the covariance matrix, we compute the accurate similarity function. The decision threshold for computing the similarity function is determined empirically. Thus, we are able to accurately estimate the perceptual similarity while retaining the algorithm complexity of an FFT-based cross-correlation at $O(mn\log(n))$.

Note that the computational complexity of obtaining S can be significantly improved if we consider only block swapping. In that case, we can consider deploying a first-degree predictor that assumes that similarity is a continuous function and predicts that similar blocks are most often found after already identified similar blocks.

4) Block substitution: In the final step, we create the resulting attacked signal $\tilde{\mathbf{x}}'$ by relocating blocks according to the realized similarities. The substitution procedure is presented using the following pseudo-code:

1	Copy $\tilde{\mathbf{x}}' := \tilde{\mathbf{x}}$.
2	Mark all blocks in $\mathcal{B}$ as unvisited.
3	Find first unvisited block B_p s.t. $\exists q \neq p \vert S_{p,q} = 1$.
4	Let G_p be a set of indices s.t. $\forall q \in G_p, S_{p,q} = 1$ or $q = p$.
5	Let L_p be a random permutation of elements of G_p.
6	Reorder blocks of $\tilde{\mathbf{x}}'$ with indices G_p according to L_p.
7	Mark blocks with indices in G_p as visited.
8	Go to 3

The effectiveness of the BPM attack depends on several factors. First, block size is a variable with an important trade-off. It is difficult to find large similar blocks, so the search clearly benefits from smaller blocks. On the other hand, it is difficult to estimate perceptual factors in small blocks. In addition, smaller blocks tend to preserve higher correlation between the original and the substitute; this phenomenon reduces the impact of BPM on the reliability of the watermark detector. Finally, smaller blocks increase the number of total blocks that need to be replaced, thus significantly increasing search run-time. In the next section, we elaborate on this trade-off in the case of audio signals.

Second, the content itself may have little redundancy regardless of block size and relative looseness of the upper bound on similarity β. In that case, the adversary needs to search a larger database of content in hope that similar sounds or pixel-maps can be found. Finally, while the upper bound β on block similarity for replacement is clearly imposed by the quality of the attacked content, a variety of parameters determine the lower bound α which allows that substituted content actually affects the watermark detector. In the next subsection, we analyze how parameter α impacts spread-spectrum watermark detection.

B. Determining the lower bound α on block similarity for replacement

Lets assume that vector $\mathbf{x}+\mathbf{w}$ is deemed similar to and replaced by vector $\mathbf{y}+\mathbf{v}$, where $\mathbf{x}$ and $\mathbf{y}$ are original signals marked with two distinct watermarks $\mathbf{w}$ and $\mathbf{v}$, where $\mathbf{w},\mathbf{v} \in \{\pm\delta\}^m$. All vectors are assumed to have the same length as a single block: m. In addition, we assume that the watermarks are spread-spectrum sequences, which means that watermark $\mathbf{w}$ is detected in a signal $\mathbf{z}$ by matched filtering: $C(\mathbf{z},\mathbf{w}) = \mathbf{z}^{\mathrm{T}}\mathbf{w}$. If $\mathbf{z}$ has been marked with $\mathbf{w}$, $\mathrm{E}[C(\mathbf{z},\mathbf{w})] = m\delta^2$, otherwise $\mathrm{E}[C(\mathbf{z},\mathbf{w})] = 0$, with variance $\mathrm{Var}[C(\mathbf{z},\mathbf{w})] = m\sigma_z^2$. Watermark is detected if $C(\mathbf{z},\mathbf{w})$ is greater than a certain detection threshold τ. In order to have symmetric probability of a false alarm and misdetection, τ is commonly set to $m\delta^2/2$.

From the requirement for two blocks to be eligible for substitution in Eqn.2:

$$m\alpha^2 \leqslant ||(\mathbf{y}+\mathbf{v})-(\mathbf{x}+\mathbf{w})||^2 \leqslant m\beta^2 \Rightarrow \quad (3)$$

$$m\alpha^2 \leqslant \mathrm{E}(||\mathbf{y}-\mathbf{x}||^2) + 2m\delta^2 - 2\mathrm{E}(C(\mathbf{v},\mathbf{w})) \leqslant m\beta^2, \quad (4)$$

we can compute the expected resulting correlation $\mathrm{E}[C(\mathbf{y}+\mathbf{v},\mathbf{w})]$ under the assumption that vectors $\mathbf{v}$ and $\mathbf{w}$ are independent with respect to $\mathbf{x}$ and $\mathbf{y}$:[1]

$$\mathrm{E}[C(\mathbf{y}+\mathbf{v},\mathbf{w})] \leqslant \frac{1}{2}\mathrm{E}(||\mathbf{y}-\mathbf{x}||^2) + m(\delta^2 - \frac{\alpha^2}{2}) \quad (5)$$

Assuming that there exists true repetition of the original content ($\mathbf{y}=\mathbf{x}$), then setting $\alpha \geqslant \delta\sqrt{2}$ would set the expected minimum correlation to zero after substitution.

III. A BPM ATTACK FOR AUDIO

In this section, we demonstrate how the generic principles behind the BPM attack can be applied against an audio watermarking technology. We first describe how an audio signal is partitioned and pre-processed for improved perceptual pattern matching. Next, we analyze the similarity function we used for our experiments. The effect of our implementation of the BPM attack on spread-spectrum and QIM watermark detection is presented in the following sections.

A. Audio processing for the BPM attack

Since most psycho-acoustic models operate in the frequency spectrum [14], we launch the BPM attack in the logarithmic (dB) frequency domain. The set of signal blocks $\mathcal{B}$ is created from the coefficients of a modulated complex lapped transform (MCLT) [14]. The MCLT is a $2\times$ oversampled DFT filter bank, used in conjunction with analysis and synthesis windows that provide perfect reconstruction. The MCLT analysis blocks

[1] $\mathrm{E}[C(\mathbf{y},\mathbf{v})] = \mathrm{E}[C(\mathbf{x},\mathbf{v})] = \mathrm{E}[C(\mathbf{y},\mathbf{w})] = \mathrm{E}[C(\mathbf{x},\mathbf{w})] = 0.$

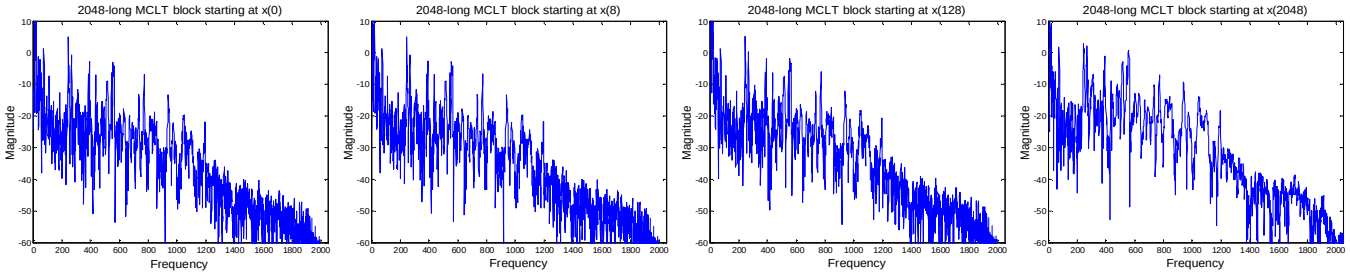

Fig. 1. MCLT blocks of a signal $\mathbf{x}$ created starting from coefficient x_0, x_8, x_{128}, and x_{2048}. MCLT block length is 2048 frequency coefficients. As block contents change little if the shift is smaller than 512 samples (at sampling rate of 44.1kHz), we adopt $\eta = 0.25$.

we consider here can potentially range in size from 128 to 1024 coefficients with an $\eta = 0.25$ overlap. Figure 1 depicts how the content of an MCLT block changes as its $2,048$-long analysis window shifts for 8, 128 and 2048 samples (content is sampled at 44.1kHz in the example).

Each block of coefficients is psycho-acoustically masked using an off-the-shelf model [14]. Similarity is explored exclusively in the audible part of the frequency spectrum. Because of psycho-acoustic masking, the similarity function in Eqn.1 is not commutative. The second operand of the function (substitution) is always masked with the psycho-acoustic mask of the first operand in addition to its own masking.

Prior to masking, the frequency spectrum of the signal is low-pass filtered to enable the detection of signal similarities regardless of the possibly different filters applied to different blocks (e.g., graphic equalization, signal amplification). Cepstrum filtering of an input MCLT block $\mathbf{y}$ is performed as follows:

1	$\mathbf{z} = \mathrm{DCT}(\mathbf{y})$	Compute the cepstrum of the dB magnitude MCLT vector $\mathbf{y}$ under test via DCT.
2	$p_i = z_i, i = 1 \ldots K$	Store the first K cepstrum coefficients ($5 < K < 20$).
3	$z_i = 0, i = 1 \ldots K$	Filter out the first K cepstrum coefficients.
4	$\mathbf{u} = \mathrm{IDCT}(\mathbf{z})$	Reconstruct the frequency spectrum via an inverse DCT. Filtered frequency spectrum $\mathbf{u}$ is used for similarity computation between blocks.

After substitution, the removed signal envelope ($p_i = z_i, i = 1 \ldots K$) is added to the substituted signal which is also cepstrum filtered. It has been demonstrated that a cepstrum filtered signal retains the level of its correlation with its spread-spectrum watermark embedded in the frequency domain [11]. Thus, cepstrum filtering does not aid in preserving the correlation that the original block had with the spread-spectrum watermark. Figure 2 illustrates the signal processing primitives used to prepare blocks of audio content for substitution.

Watermark length is assumed to be greater than one second. In addition, we assume that watermark chips may be replicated along the time axis at most for 1 second[2] [11]. Thus, we restrict that for a given block its potential substitution blocks are *not* searched within 1 second.

[2]Higher level of redundancy may enable effective watermark estimation.

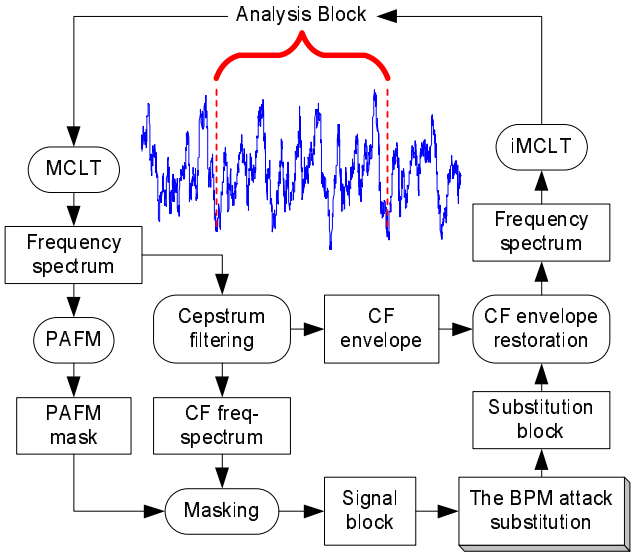

Fig. 2. Block diagram of the signal processing primitives performed as pre- and post-processing to the BPM attack. The BPM attack has the current block as an input and replaces it with another perceptually similar block. Envelope restoration: the stored envelope $p_i = z_i, i = 1 \ldots K$ is added to the substituted block in the cepstrum domain.

B. Analysis of the similarity function

We performed several experiments in order to evaluate the effectiveness of the BPM attack. The first set of experiments aimed at quantifying similarity between blocks of several audio clips marked with spread-spectrum watermarks at $\delta =$ 1dB. As an example, Figure 3 shows the values of the similarity function for five 256-long MCLT blocks randomly selected from two 30 second sub-clips taken from two different songs of different style: rock (left sub-figure) and techno (right sub-figure). In the example, block similarity is computed over the 2–7kHz sub-band. This is a realistic assumption because commonly watermark codecs hide data in a sub-band that is not strongly distorted by compression and medium quality low- and high-pass filtering [11]. For the illustrated data, 2 out of 5 MCLT blocks can be substituted with other MCLT blocks that are within $\beta = 4$dB. We make an important observation that in many cases the detected similarity is not a result of

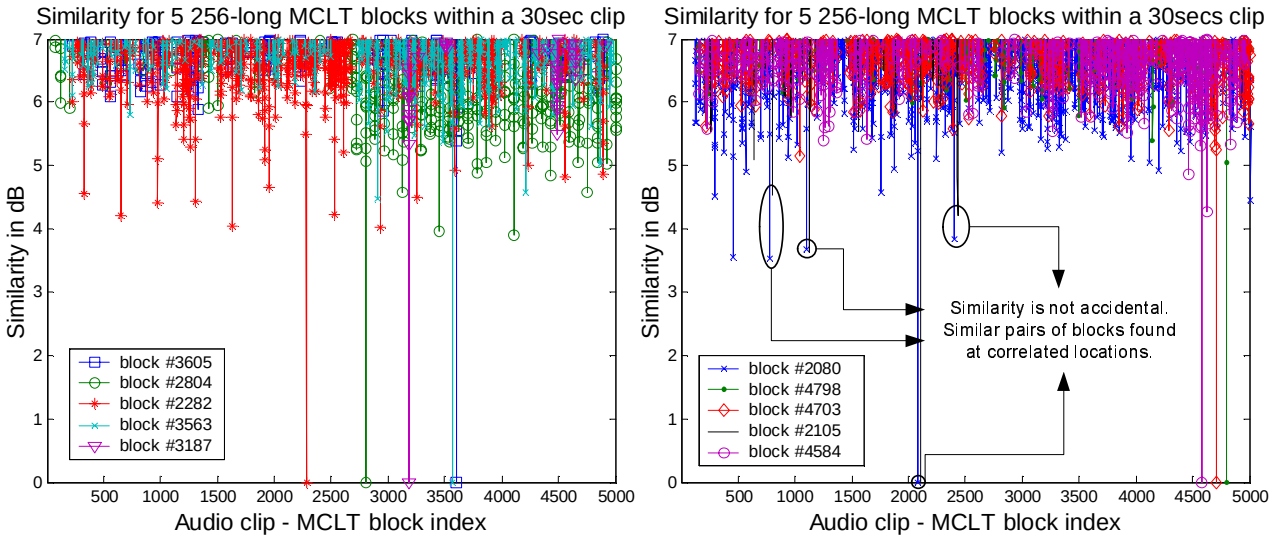

Fig. 3. Similarity computation for two 30s audio clips (rock – left; techno – right) and 5 randomly selected 256-long MCLT blocks. Zero-similarity denotes equality. Even for such small examples, 2 out of 5 MCLT blocks have at least one substitution block within $\beta = 4$dB relative noise.

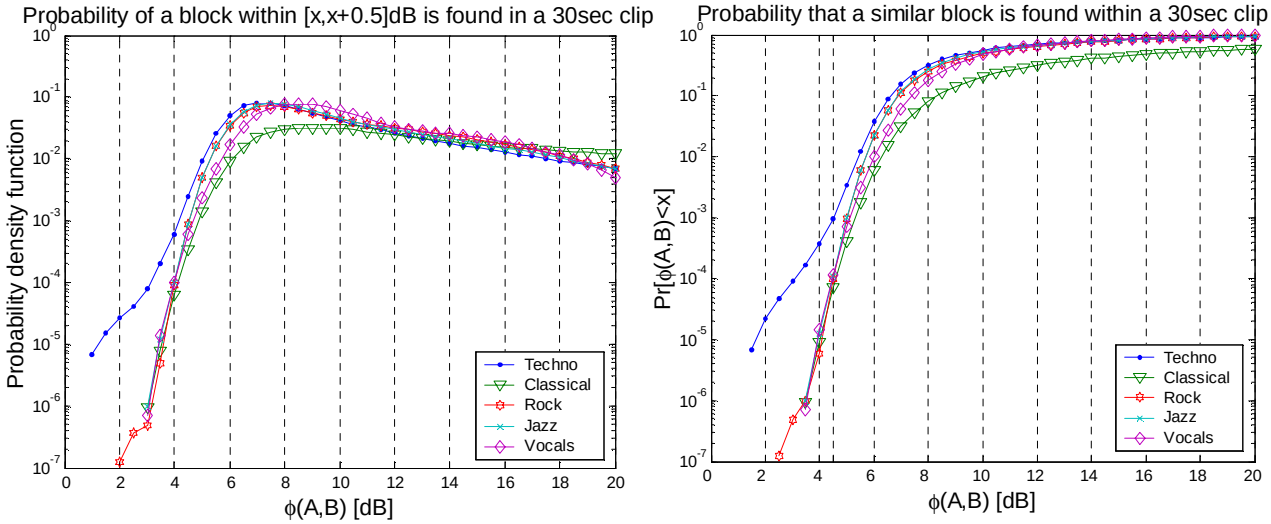

Fig. 4. Probability density function of the similarity function $\phi(A, B)$ on two blocks within a 30s audio clips for five different types of music: rock, classical, pop, vocals and techno. A certain value x on the abscissa of the left subfigure represents a histogram bin from x to $x + 0.5$.

coïncidence, but a consequence of repetitive musical content. For example, in the right sub-figure of Figure 3, we observe that a set of detected pairs of blocks similar (their similarity value is less than 4.5dB) to a pair of neighboring blocks indexed 2080 and 2105 (all pairs circled), preserve the same index distance as illustrated in the figure.

Figure 4 illustrates the probability that for a given 256-long MCLT block A, there exists another block B within 30s of the same audio clip that is within $\phi(A, B) \in [x, x + 0.5)$dB (left subfigure) or that has a similarity function smaller than $\phi(A, B) \leqslant x$dB (right subfigure), where x is some real number. For a benchmark set of distinctly different musical pieces, from Figure 4, one can conclude that within a 30s audio clip approximately one half of all blocks can be substituted with similar blocks that are within 5dB of noise. The probability of finding a similar block should rise proportionally to the length of the audio clip considered for substitution. Finally, note that electronically generated musical content (in our benchmark a techno song) is significantly more likely to contain perceptually correlated blocks than music that is a performed art.

The second set of experiments aimed at quantifying the effect of MCLT block size on the similarity function between blocks. Figure 5 illustrates how the pdf of the similarity function $\phi(A, B)$ changes with the increase of the MCLT block size. Clearly, having larger MCLT blocks reduces the complexity of computing the similarity matrix as there are fewer blocks to be compared. However, with the increase in MCLT block size, the probability of finding similar blocks under certain noise limit β decreases. In the region of interest, $\phi(A, B) \leqslant 4.5$dB, the likelihood of finding larger MCLT blocks is relatively higher, thus, we restrict our analysis to

256-long MCLT blocks.

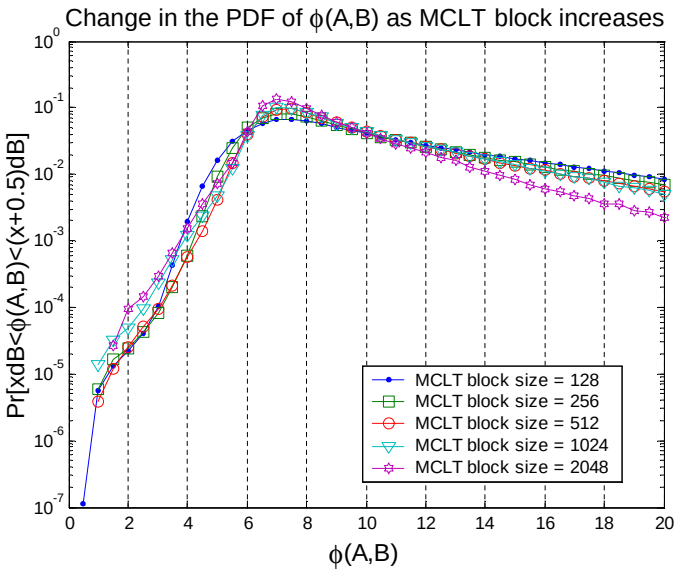

Fig. 5. Illustration of how the increase of MCLT block size affects the probability density function of the similarity function $\psi(A, B)$ of two blocks within a 30s audio clip. The result is collected from a techno song. A certain value x on the abscissa represents a histogram bin from x to $x + 0.5$.

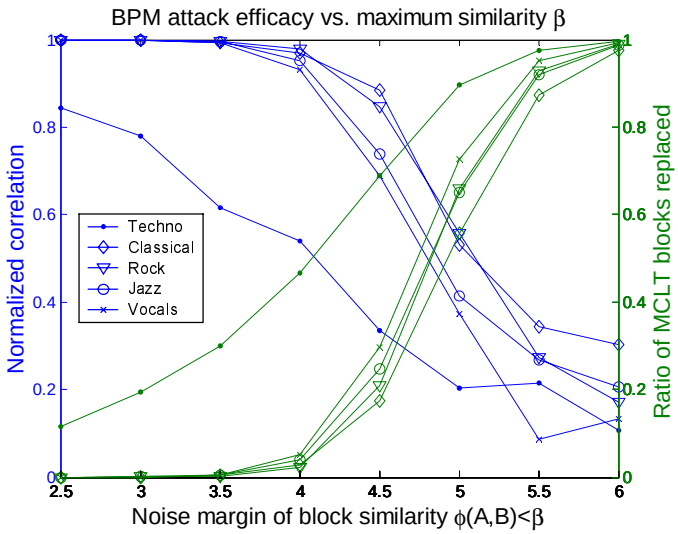

Fig. 6. Response of a spread-spectrum watermark detector to the BPM attack. The abscissa depicts the increase in β for fixed $\alpha = 1.5$dB and watermark amplitude of $\delta = 1$dB. The left ordinate shows the decrease of the normalized correlation as β increases. The right ordinate shows the number of 256-long MCLT blocks substituted for a 2000-block watermark within a 5000-block audio clip.

IV. Effect of the BPM attack

In this section we study the effect of the BPM attack on two popular types of modulations used for digital watermarking: spread-spectrum and QIM.

A. Effect of the BPM attack on spread-spectrum watermarks

In order to evaluate the effect of a BPM attack on spread-spectrum watermarks, we conducted two experiments. For both experiments, we used spread-spectrum watermarks that spread over $2,000$ consecutive 256-long MCLT blocks (approximately 13s long), where only the frequency magnitudes in the 2–7kHz sub-band were marked. We did not use chip replication as its effect on watermark detection is orthogonal with respect to the BPM attack.

Figure 6 shows how normalized correlation of a spread-spectrum watermark detector drops with the increase of search freedom β for fixed $\alpha = 1.5$dB. Block substitution was performed within the watermarked part of the signal itself and additionally, within the following 3000 consecutive MCLT blocks of the same audio clip with a full length of $5,000$ MCLT blocks or approximately 30 seconds.

It is important to observe the effect of the BPM attack on synthetic (e.g., techno) and performed musical content. The attack is far more effective on synthetic content. However given large enough content libraries the BPM attack should be substantially more effective for performed music. As an example, note that the reduction in correlation due to BPM (left ordinate in Figure 6) is strongly proportional to the ratio of blocks actually replaced within the watermarks (right ordinate in Figure 6. A larger substitution base directly impacts the ratio of substituted blocks and hence proportionally affects the correlation test.

The power of the BPM attack is most notably observed by comparing the effect of adding a white Gaussian noise (AWGN) pattern $\mathbf{n} = \mathcal{N}(0, \sigma_n)$ of certain standard deviation $\sigma_n = [2.5 \dots 6]$dB, with a BPM attack of equivalent similarity tolerance $\beta = \sigma_n$. Whereas the dramatic effect of BPM can be observed in Figure 6, AWGN with realistic β affects the correlation detector only negligibly. In the latter case, the expected correlation value remains the same $E[C(\tilde{\mathbf{x}}+\mathbf{n}, \mathbf{w})] = E[C(\tilde{\mathbf{x}}, \mathbf{w})]$, with increased variance $Var[C(\tilde{\mathbf{x}} + \mathbf{n}, \mathbf{w})] = Var[C(\tilde{\mathbf{x}}, \mathbf{w})] + m\sigma_n^2$. Finally, additive noise of 4–5dB in the 2–7kHz is a relatively tolerable modification.

Figure 7 illustrates the dependency between the watermark amplitude δ and the correlation decrease due to the BPM attack. The parameters of the watermark codec are the same as in the previous set of experiments. Clearly, with the increase of watermark amplitude, the search process of the BPM attack becomes harder for two reasons: (*i*) block contents become more randomized and (*ii*) the substituted blocks are more correlated with the original blocks because β is fixed (in this case at 6dB). Although stronger watermarks may sound like a solution to the BPM attack, high watermark amplitudes cannot be accepted because of two reasons: first, the requirement for high-fidelity marked content and second, strong watermarks can be efficiently estimated using an optimal watermark estimator [3], i.e. estimate $\mathbf{v} = \text{sign}(\mathbf{x} + \mathbf{w})$ makes an error per bit $\varepsilon = \Pr[v_i \neq w_i] = \frac{1}{2}\text{erfc}(\frac{\sigma_x}{\delta\sqrt{2}})$ exponentially proportional to δ.

B. Effect of the BPM attack on quantization index modulation schemes

Quantization Index Modulation (QIM) is another type of commonly used data hiding technology. It was introduced

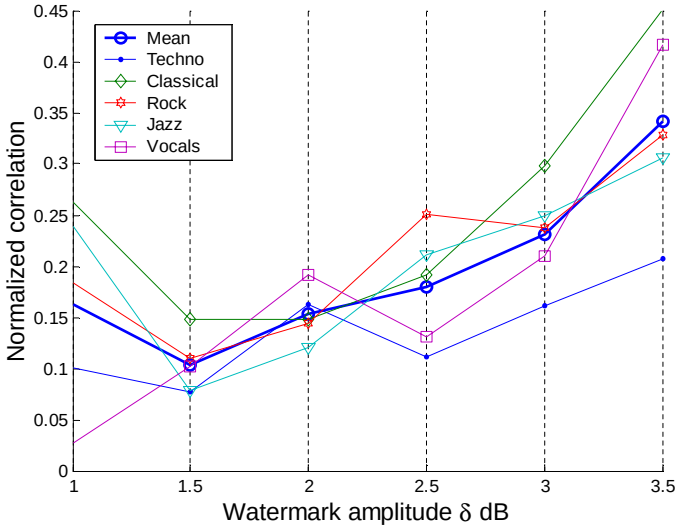

Fig. 7. Effect of watermark amplitude on the effectiveness of the BPM attack. The abscissa depicts the amplitude δ of a spread-spectrum watermark applied to a set of benchmark clips. The left ordinate shows the decrease of the normalized correlation as δ increases. The right ordinate shows the number of 256-long MCLT blocks substituted for a 2000-block watermark within a 5000-block audio clip.

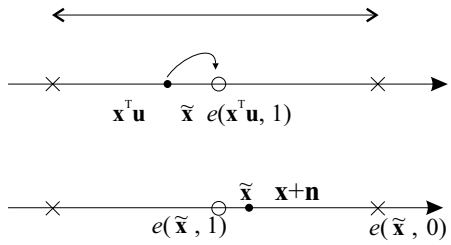

Fig. 8. STDM embedding process. The binary spread-transform dither modulation embeds a bit of payload by quantizing the projection of the original signal vector $\mathbf{x}$ on a random unit vector $\mathbf{u}$ using a given quantizer step Δ. Two dither values are used for one bit of payload. In this figure, they are represented by the crosses and circles. Detection is done by checking to which quantizer the signal is closer.

as a provably robust watermarking technique under specific assumptions for the communication and, more importantly, the attack model [13]. One of the features of QIM is that, as opposed to spread-spectrum, by revealing the hidden secret, the likelihood that the adversary can recreate the original signal is exceptionally low. However, in almost all real-life applications of a watermarking technology, the adversary can afford to introduce several times greater distortion than the data hiding agent due to much stricter requirements for high-fidelity imposed upon the data hiding agent. In this section, we adopt an advanced variant of QIM [13] and show the extent to which QIM applied to audio signals is vulnerable to the BPM attack.

One possible practical implementation of QIM is called binary spread-transform dither modulation (STDM) [13]. STDM is illustrated in Figure 8. It uses a uniform scalar quantizer $q_\Delta(.)$ of step size Δ which is selected such that the distortion induced by the hiding process remains lower than a constant factor D_x. For example, $\frac{1}{N}||\tilde{\mathbf{x}} - \mathbf{x}||^2 \leq D_x$.

When embedding a single bit of payload information, the technique can be summarized as follows. A unit length spreading vector $\mathbf{u}$ along with dither value d_0 are chosen pseudo-randomly, possibly using a watermarking key as seed to the random number generator. A second dither value d_1 is derived from d_0 using the following constraint:

$$d_1 = \begin{cases} d_0 + \Delta/2 & \text{if } d_0 < 0; \\ d_0 - \Delta/2 & \text{otherwise.} \end{cases} \tag{6}$$

Embedding is done by quantizing the projection of the original signal onto $\mathbf{u}$ using one of the two dither values depending on the payload bit b:

$$\tilde{\mathbf{x}} = e(\mathbf{x}, b) = \mathbf{x} + (q_\Delta(\mathbf{x}^{\mathrm{T}}\mathbf{u} + d_b) - (\mathbf{x}^{\mathrm{T}}\mathbf{u} + d_b))\mathbf{u} \tag{7}$$

At detection time, the received signal $\tilde{\mathbf{x}}' = \tilde{\mathbf{x}} + \mathbf{n}$, where $\mathbf{n}$ represents the noise introduced by the channel (e.g., an attack) is compared against the quantizers and the decoded bit $\hat{b}$ corresponds to the minimal distance decoder, that is the decoder which chooses the reconstruction point closest to the received vector $\tilde{\mathbf{x}}'$:

$$\begin{aligned} \hat{b} = \arg\min_i ||\tilde{\mathbf{x}}' - e(\tilde{\mathbf{x}}', i)|| = \\ \arg\min_i ||q_\Delta(\tilde{\mathbf{x}}'^{\mathrm{T}}\mathbf{u} + d_i) - (\tilde{\mathbf{x}}'^{\mathrm{T}}\mathbf{u} + d_i)||. \end{aligned} \tag{8}$$

In order to demonstrate the effect of the BPM attack on the adopted QIM-based data hiding scheme, we have conducted an experiment on our selected audio benchmark suite. In the experiment, we chose $\mathbf{x}$ to be the magnitude of the MCLT coefficients whose frequency is between 2–7kHz (in correspondence to the spread-spectrum tests). We marked the coefficients of the marking sub-band of the first 500 MCLT blocks using STDM and adopted the following 4500 MCLT blocks as a substitution base.

In the experiment, we show how the distance to the quantizers changes after applying the BPM attack with increasing similarity tolerance β. While watermarking, we impose a fixed distortion to signal ratio of $D_x = 1$dB on the MCLT coefficients in the marking sub-band. Assuming a uniform distribution of the carrier audio signal, the desired distortion corresponds to a quantizer step of $\Delta = \sqrt{12D_x}$ [13].

Figure 9 illustrates how the Euclidian distance with respect to the corresponding and opposite quantizer increases as β increases from 2 to 9dB. For example, for a signal with embedded '**1**', a pair of curves for each benchmark clip depicts the average distance for 30 different markings (different random unit vectors $\mathbf{u}$) with respect to the corresponding **1**-quantizer (the increasing curves) and the opposite **0**-quantizer (the decreasing curves). We denote as $\mathbf{q0}$ and $\mathbf{q1}$ the functions that return the Euclidian distance of a given marked signal from the corresponding and opposite quantizer respectively. Note that the distance between a quantizer and a non-marked signal is a random variable uniformly distributed within $[0, 1]$.

Curves specified in the legend of Figure 9 illustrate the average of the attacks that result in bit alteration (i.e. the expected distance for this case is 0.75 for the corresponding and 0.25 for the opposite quantizer). In addition, bold lines

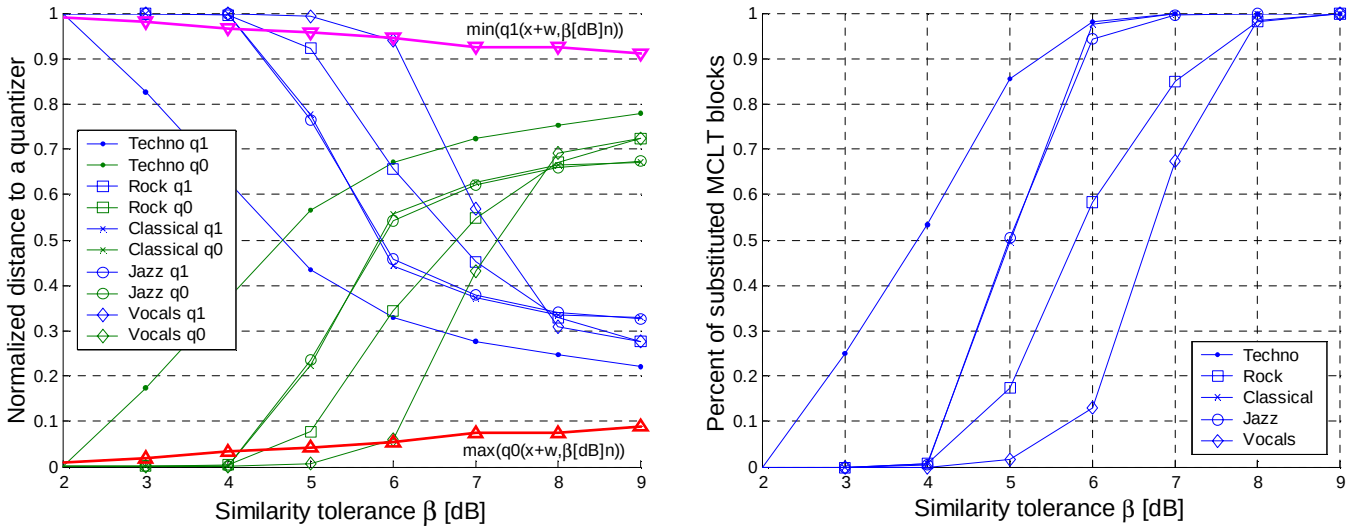

Fig. 9. Response of a QIM watermark detector to the BPM attack. Abscissas on both subplots depict the increase in β for fixed $\alpha = 2$dB and expected watermark distortion of $D_x = 1$dB. The ordinate of the left subplot shows the decrease of the normalized distance with respect to the first (d_0) and second (d_1) quantizers as β increases. The ordinate of the right subplot shows the ratio of 256-long MCLT blocks of a 500-block QIM watermark replaced with blocks from the remainder of a 5000-block audio clip as similarity tolerance increases within $\beta = [2 \dots 9]$dB.

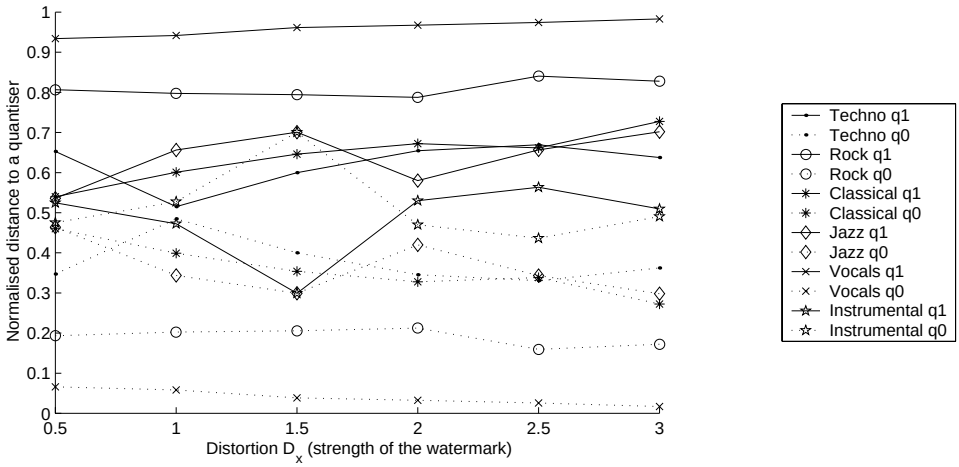

Fig. 10. Response of a QIM watermark detector for a fixed similarity tolerance of the BPM attack after the content has been watermark with different strength. Each value represents an average of 50 different markings. Abscissa depicts the increase of the watermark distortion D_x. The ordinate shows the normalized distance to both quantizers for a given similarity tolerance $\beta = 5$dB.

marked '$\triangle$' and '∇' depict the maximal and minimal distance from the corresponding and opposite quantizer for 30 different tests when AWGN $\mathbf{n} \sim \mathcal{N}(0, \beta)$, equivalent in energy to the BPM attack βdB, is superposed to the marked clip. Notably, just as in the case of spread-spectrum watermarks, the effect of the BPM attack is substantially more tangible to watermark robustness then just AWGN of the same amplitude.

In addition, the results illustrated in Figure 9 are only a presentation of the power of the BPM attack as the substitution base is of very limited size (30s) for the experiments. Note that the effect of the BPM attack proportional to the ratio of blocks actually replaced in the clip (presented in the right subplot of Figure 9). With a sufficiently large substitution base, we expect that all MCLT blocks within $\beta < 5$dB can be substituted resulting in watermark removal with a tolerable effect on sound fidelity.

Figure 10 illustrates the dependency between the watermark amplitude (the distortion introduced by the marking process) and the performance decrease due to the BPM attack. The parameters of the watermark codec are the same as in the previous set of experiments. As expected the increase of the watermark amplitude diminishes the effect of the attack but, just like in the case of spread-spectrum, increasing the amplitude of the mark to survive the attack is not possible in practice due to the high-fidelity constraints imposed by the content owner.

V. Conclusion

For any watermarking technology and any type of content, an ultimately powerful attack is to re-record the original content; i.e. perform again the music or capture the image of the same original visual scene. In this paper, we simulate

this attack: given a library of multimedia content, the BPM attack aims at replacing small pieces of the marked content with perceptually similar but unmarked[3] substitutions from the library. The hope is that the substitutions have little correlation with the embedded mark. Although the attack is generic and can be applied to all marking strategies, we demonstrate how the BPM attack can be launched for audio content and two traditional watermarking technologies: spread-spectrum and quantization index modulation. From the presented experimental results, we conclude that block substitution within a 30 second audio clip that creates approximately 4–5dB noise with respect to the marked content, is sufficient to bring a spread-spectrum correlation detector to half the expected value without an attack. Similar adversarial effects can be obtained against QIM-based watermarking schemes.

At this point, we identify two possible prevention strategies against a BPM attack. For example, a data hiding primitive may identify rare parts of the content at watermark embedding time and mark only these blocks. However this reduces significantly the practical capacity of the scheme and increases dramatically the complexity of the embedding process. In the case of spread-spectrum watermarks, longer watermarks and increased detector sensitivity may enable watermark detection at lower thresholds ($\tau < m\delta^2/5$). Unfortunately, such a solution comes at the expense of having significantly longer watermarks which results in a significantly lowered robustness with respect to de-synchronization attacks such as fluctuating time- and frequency-scaling.

ACKNOWLEDGMENTS

The authors thank Dr Cédric Fournet for helpful comments that improved the clarity of the manuscript.

REFERENCES

[1] A. Patrizio, "DVD piracy: It can be done." http://www.wired.com/news/technology/0,1282,32249,00.html, Nov. 1999.
[2] "Secure digital music initiative." http://www.sdmi.org.
[3] D. Kirovski, H. Malvar, and Y. Yacobi, "A dual watermarking and fingerprinting system." Tech. Rep. MSR-TR-2001-57, Microsoft Research, Jun. 2001.
[4] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems." In Aucsmith [15], pp. 218–238, ISBN 3-540-65386-4.
[5] G. C. Langelaar, R. L. Lagendijk, and J. Biemond, "Removing spatial spread spectrum watermarks by non-linear filtering." In *9th European Signal Processing Conference (EUSIPCO'98)*, pp. 2281–2284, Island of Rhodes, Greece, 8–11 Sep. 1998, ISBN 960-7620-05-4.
[6] S. Voloshynovskiy, S. Pereira, A. Herrigel, N. Baumgartner, and T. Pun, "Generalised watermarking attack based on watermark estimation and perceptual remodulation." In Wong and Delp [17], pp. 358–370, ISBN 0-8194-3589-9.
[7] M. Kutter, S. Voloshynovskiy, and A. Herrigel, "The watermark copy attack." In Wong and Delp [17], pp. 371–380, ISBN 0-8194-3589-9.
[8] G. D. Cohen and G. Zemor, "Intersecting codes and independent families." *IEEE Transactions on Information Theory*, vol. 40, no. 6, pp. 1872–1881, Nov. 1994.
[9] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data." *IEEE Transactions on Information Theory*, vol. 44, no. 5, pp. 1897–1905, Sep. 1998.
[10] F. A. P. Petitcolas and R. J. Anderson, "Evaluation of copyright marking systems." In *IEEE Multimedia Systems (ICMCS'99)*, pp. 574–579, Florence, Italy, 7–11 Jun. 1999.
[11] D. Kirovski and H. Malvar, "Robust covert communication over a public audio channel using spread spectrum." In Moskowitz [18], pp. 354–368, ISBN 3-540-42733-3.
[12] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "A secure, robust watermark for multimedia." In Anderson [16], pp. 183–206, ISBN 3-540-61996-8.
[13] B. Chen and G. W. Wornell, "Quantisation index modulation: a class of provably good methods for digital watermarking and information embedding." *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.
[14] H. Malvar, "A modulated complex lapped transform and its application to audio processing." In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP'99)*, pp. 1421–1424, Phoenix, Arizona, U.S.A., 15–19 Mar. 1999.
[15] D. Aucsmith, ed., *Information Hiding: Second International Workshop*, vol. 1525 of *Lecture Notes in Computer Science*, Portland, Oregon, U.S.A., 1998, Springer-Verlag, Berlin, Germany, ISBN 3-540-65386-4.
[16] R. J. Anderson, ed., *Information hiding: first international workshop*, vol. 1174 of *Lecture Notes in Computer Science*, Isaac Newton Institute, Cambridge, England, May 1996, Springer-Verlag, Berlin, Germany, ISBN 3-540-61996-8.
[17] P. W. Wong and E. J. Delp, eds., *Security and Watermarking of Multimedia Contents II*, vol. 3971, San Jose, California, U.S.A., 24–26 Jan. 2000, The Society for Imaging Science and Technology (IS&T) and the International Society for Optical Engineering (SPIE), SPIE, ISBN 0-8194-3589-9.
[18] I. S. Moskowitz, ed., *Information hiding: fourth international workshop (IH'2001)*, vol. 2137 of *Lecture Notes in Computer Science*, Pittsburgh, Pennsylvania, U.S.A., 2001, Springer-Verlag, Berlin, Germany, ISBN 3-540-42733-3.

[3]Or marked with a different watermark.

Darko Kirovski received his Ph.D. degree from the Computer Science Department at the University of California in Los Angeles in January 2001. He joined Microsoft Research as a researcher in April 2000. His research interests include: secure systems, software delivery, multimedia processing and applications, intellectual property protection, and embedded system design. He has been awarded the 1998-2000 Microsoft Graduate Research Fellowship, the 1999-2000 ACM/IEEE Design Automation Conference Graduate Scholarship, and the 2002 ACM Outstanding PhD dissertation in Electronic Design Automation Award. He has received the Best Paper Award at the ACM Multimedia 2002. His work is presented in more than 70 journal and conference papers and patent filings.

Fabien A.P. Petitcolas received his PhD on information hiding in 1999 from the University of Cambridge, England. He then joined Microsoft Research as a researcher for thee years. Recently he joined the University Relations team as manager for UK and Ireland. He is the editor of the first book on information hiding. In 2002 he chaired the fifth international workshop on information hiding.