

# Audio Watermark Robustness to Desynchronization via Beat Detection

Darko Kirovski and Hagai Attias

Microsoft Research  
One Microsoft Way, Redmond, WA, USA  
{darkok,hagaia}@microsoft.com

**Abstract.** Watermarks are hidden, imperceptible, and robust marks augmented into a host signal such as audio or video. Recent studies show that in the presence of an adversary, "blind" watermark detection within an attacked clip is an exceptionally difficult task. In this paper, we explore two technologies, beat detection and block redundant coding, to combat de-synchronization and watermark estimation as two attacks that have demonstrated superior effectiveness in preventing watermark detectors from reliably accomplishing their goal. As a result, we have achieved robustness of spread-spectrum watermarks augmented in audio clips to almost arbitrary constant time-warp, pitch-bending, and wow-and-flutter of up to 1%. The adversary can remove the watermark by subtracting an estimate of the watermark from the signal with an amplitude in excess of 6dB with respect to the host. Such an attack vector typically affects substantially the fidelity of the "pirated" recording.

## 1 Introduction

With the growth of the Internet, unauthorized copying and distribution of digital media has never been easier. As a result, the music industry claims a multi-billion dollar annual revenue loss due to piracy [20], which is likely to increase due to peer-to-peer file sharing Web communities. One source of hope for copyrighted content distribution on the Internet lies in technological advances that would provide ways of enforcing copyright. Traditional data protection methods such as scrambling or encryption cannot be used, since the content must be played back in the original form, at which point it can always be re-recorded and then freely distributed. A promising solution to this problem is marking the media signal with a secret, robust, and imperceptible watermark. The media player at the client side can detect this mark and consequently enforce a corresponding e-commerce policy.

Recent introduction of a content screening system that uses asymmetric direct sequence spread-spectrum WMs has significantly increased the value of WMs, because a single compromised detector (client player) in that system does not affect the security of the content [14]. In order to compromise the security of such a system without any traces, an adversary needs to break large number of players for a typical two-hour feature movie. Although the effectiveness of any content screening system requires global adoption of many standards, the industry is determined to carry out such a task [21].

**Watermarking Technologies.** Audio watermarking schemes rely on the imperfections of the human auditory system (HAS) [11]. Numerous data hiding techniques explore

the fact that the HAS is insensitive to small amplitude changes, either in the time [2] or frequency [5], [19], [23] domains, as well as insertion of low-amplitude time-domain echoes [7]. Information modulation is usually carried out using: direct sequence spread spectrum (SS) [24] or quantization index modulation (QIM) [4]. The main advantage of both SS and QIM is that WM detection does not require the original recording.

However, it is important to review the disadvantages that both technologies exhibit. First, the marked signal and the WM have to be perfectly synchronized at WM detection. Next, to achieve a sufficiently small error probability, WM length may need to be quite large, increasing detection complexity and delay. Finally, the most significant deficiency of both schemes is that by breaking a single player (debugging, reverse engineering, or the sensitivity attack [16]), one can extract the secret information (the SS sequence or the hidden quantizers in QIM) and recreate the original (in the case of SS) or create a new copy that induces the QIM detector to identify the attacked content as unmarked. While an effective mechanism for enabling asymmetric SS watermarking has been developed [14], an equivalent system for QIM does not exist to date, which renders QIM at this point relatively impractical for content screening.

### 1.1 Spread-Spectrum Watermarking of Audio via Beat Detection

One of the most effective attacks on almost any type of watermarking system is **de-synchronization**. In order to validate existence of a WM, the detector usually computes a certain statistical measure dependent upon the WM – however, the statistical measure is accurate only if the location of the detector is known with relatively high precision. A typical de-synchronization attack, such as StirMark [1], aims at rescaling the multimedia object with a variable scaling factor such that both the location and the size of the WM are changed as much as possible under the hi-fidelity requirement.

Kirovski and Malvar proposed block repetition coding of the WM chip combined with multi-test WM search as a remedy for this problem in audio [12]. Unfortunately, the deployed redundancy, while providing robustness to de-synchronization, opens doors to another attack: **watermark estimation** [15]. The more redundancy, the better the robustness, but also the more accurate the attacker’s WM estimate.

In this paper, we introduce beat detection as the key tool for enabling synchronicity between the WM detector and the location of the WM in an audio clip. In an exemplary watermarking system, we perform marking in several steps. First, we identify the average beat period in the clip. Then, we identify the location of each beat as accurately as possible. Next, we rescale the clip such that the length of each beat period is constant and equal to the average beat period rounded to the nearest multiple of a certain block of samples (typically, 1024 samples). The rescaled clip is marked with a SS sequence where each chip has an amplitude proportional to the variance of the host signal in its locality. The marked content is finally created by rescaling the marked clip back to its original tempo.

Assuming the adversary may have rescaled the content with a variable but slow-varying scaling factor, we detect the WM using a multi-test search. First, the same beat-scaling transform is applied to the clip as during embedding. Next, using the same multi-search process as described in [12], we perform matched filtering with the hidden secret in order to detect WM existence. Using beat detection as means of synchronization, we reduce the redundancy of the block repetition codes up to 4 times, while attaining the same robustness to variable time-warp. Another consequence of using beat detection as

a synchronization mechanism is that WMs can be placed at known positions in the clip (e.g. starting from a beat) which can speed up the search up to an order of magnitude compared to "blind" and exhaustive search [12].

In the remainder of the paper, we describe in detail the marking and detection procedures and we present a beat detector based on a variant of the expectation-maximization (EM) algorithm [6]. We investigate the robustness of such a technology with respect to WM estimation and de-synchronization both analytically and empirically. We show that if the variance of the time-warp attack within a single beat does not exceed a certain realistic limit value, the multi-search highly reliably synchronizes the detector with the location of the WM in the clip. The limit on the time-warp variance is relatively high, about 1%, knowing the strong impact of such an attack to sound fidelity. Finally, we show that, based on the deployed redundancy, the adversary can remove the WM by subtracting an estimate of the WM from the signal with an amplitude in excess of 6dB with respect to the host, a noise signal that has a strong impact on sound fidelity.

## 1.2 Applications of Watermarking Technologies

While it seems that WMs as defined can provide powerful copyright protection tools [10], it turns out that all of the four most important copyright protection applications actually do not need classic WMs<sup>1</sup>.

CONTENT SCREENING – assumes that media players detect WMs before playing the content and if the WM is present and the user does not have the license to play the content the media player refuses to play the media. Clearly, by the definition of both classic SS and QIM watermarking, the player needs to store the secret (SS sequence or the QIM quantizers) in order to detect the WM. The adversary in such case does not target the robustness of the WM but aims to reverse engineer or debug the media player in order to extract the hidden secret, i.e. the root of system security. A single broken client breaks the security of the entire system in such a scenario. Thus, some form of public-key watermarking is required, where the adversary cannot break the security of the entire system by breaking a single client as the detection key should not reveal the hidden secret.

PROOF OF AUTHORSHIP – assumes that the author of the content is distributing only a marked version of her recording, where the mark serves as a statistically undeniable proof of creation. In this scenario, both the original and the secret are securely stored only with the author and detection is performed potentially only in court. "Blind" detection is really not a requirement here, because in court, the author uses both the original and the hidden secret to demonstrate authorship.

TRACING ROOTS OF PIRACY – is usually a goal of media studios, who create several copies of the original content each marked with a distinct mark (fingerprint) and distribute these copies to their clients. If a "pirated" copy is found, the marks are used to trace that copy to the client-"pirate". Again, "blind detection" is not an issue just as in the previous case because the media studios use both the original copy as well as the fingerprint data only internally or in court. The main system requirement is collusion resistance, i.e. the number of copies that a clique of malicious users needs to create a clean copy or a copy which points to a client not in the clique [3]!

---

<sup>1</sup> Imperceptible hidden marks that are robustly and reliably detected in the presence of the adversary in a "blind" manner, i.e. without having access to the original recording.

TRACING UNLICENSED BROADCAST – refers to automated monitoring of the content played by a broadcasting station (radio, TV, e-radio, etc.) against a database of licenses. The system creates a proof of an unlicensed broadcast and sends it to the copyright owners. Commonly, two technologies are mentioned in this context: non-robust WMs or content identifiers (hashes) [8]. WM robustness in the most general sense, is not a system requirement as it is expected that due to legal penalties related to ”willful infringement”, the broadcaster is not likely to tamper with the WMs.

In this paper, we refer to content screening as the main target of our watermarking technology. We assume that the dual watermarking and fingerprinting system [14] is deployed as a system-level solution, which poses a requirement of robustness of SS WMs in the traditional sense.

## 2 Fundamentals of Spread-Spectrum Watermarking

Beat detection can be potentially attached to any audio watermarking system: e.g. SS or QIM. However, for the sake of applicability to content screening and in the light of using the dual watermarking and fingerprinting system, in this paper, we restrict our work exclusively to SS. In this section, we review the fundamentals of SS data hiding.

The media signal to be watermarked  $x \in \mathbb{R}^N$  can be modeled as a random vector, where each element  $x_i \in x$  is a normal independent identically distributed (i.i.d.) random variable with standard deviation  $\sigma_x$ , i.e.  $x_i \sim \mathcal{N}(0, \sigma_x)$ .<sup>2</sup> Signal  $x$  actually represents a collection of blocks of samples from an appropriate invertible transformation on the original audio signal [5], [23], [24]. Modeling  $x$  with a gaussian is relatively accurate because at detection time samples with redundant WM information are averaged – regardless of the pdf of a single sample, due to the Central Limit Theorem their sum quickly takes the looks of a gaussian. A *watermark* is defined as a direct SS sequence  $w$ , which is a vector pseudo-randomly generated in  $w \in \{\pm 1\}^N$ . Each element  $w_i$  is usually called a “chip”. WM chips are generated such that they are mutually independent with respect to the original recording  $x$ . The marked signal  $y$  is created by vector addition  $y = x + \delta w$ , where  $\delta$  is the WM amplitude. Signal variance  $\sigma_x^2$  directly impacts the security of the scheme: the higher the variance, the more securely information can be hidden in the signal. Similarly, higher  $\delta$  yields more reliable detection, less security, and potential WM audibility.

Let  $p \cdot q$  denote the normalized inner product of vectors  $p$  and  $q$ , i.e.  $p \cdot q \equiv N^{-1} \sum p_i q_i$ , with  $p^2 \equiv p \cdot p$ . For example, for  $w$  as defined above, we have  $w^2 = 1$ . A WM  $w$  is detected by correlating (matched filtering) a given signal vector  $z$  with  $w$ :

$$C(z, w) = z \cdot w = E[z \cdot w] + \mathcal{N}(0, \sigma_x / \sqrt{N}). \quad (1)$$

Under no malicious attacks or other signal modifications, if the signal  $z$  has been marked, then  $E[z \cdot w] = \delta$ , else  $E[z \cdot w] = 0$ . The detector decides that a WM is present if  $C(z, w) > \theta$ , where  $\theta$  is a detection threshold that controls the tradeoff between the probabilities of false positive and false negative decisions. We recall from modulation and detection theory that under the condition that  $x$  and  $w$  are i.i.d. signals, such a detector is optimal [25]. Finally, the probability  $P_{FA}$  that the detection decision is a false alarm is quantified as:

<sup>2</sup>  $\mathcal{N}(a, b)$  denotes a Gaussian with  $a$ -mean and  $b^2$ -variance.

$$P_{FA} = \Pr[C(z, w) \geq \theta | (z = x)] = \frac{1}{2} \operatorname{erfc} \left( \frac{\theta \sqrt{N}}{\sigma_x \sqrt{2}} \right), \quad (2)$$

and the probability  $P_{MD}$  that the detection decision is a misdetection, equals:

$$P_{MD} = \Pr[C(z, w) \leq \theta | (z = x + w)] = \frac{1}{2} \operatorname{erfc} \left( \frac{(E[z \cdot w] - \theta) \sqrt{N}}{\sigma_x \sqrt{2}} \right). \quad (3)$$

Due to the de-synchronization and estimation attack, straightforward application of the above mentioned SS WM principles does not provide reliability nor robustness. In the following sections, we outline the deficiencies of the basic SS WM paradigm and provide solutions for improved SS WM robustness and detection reliability.

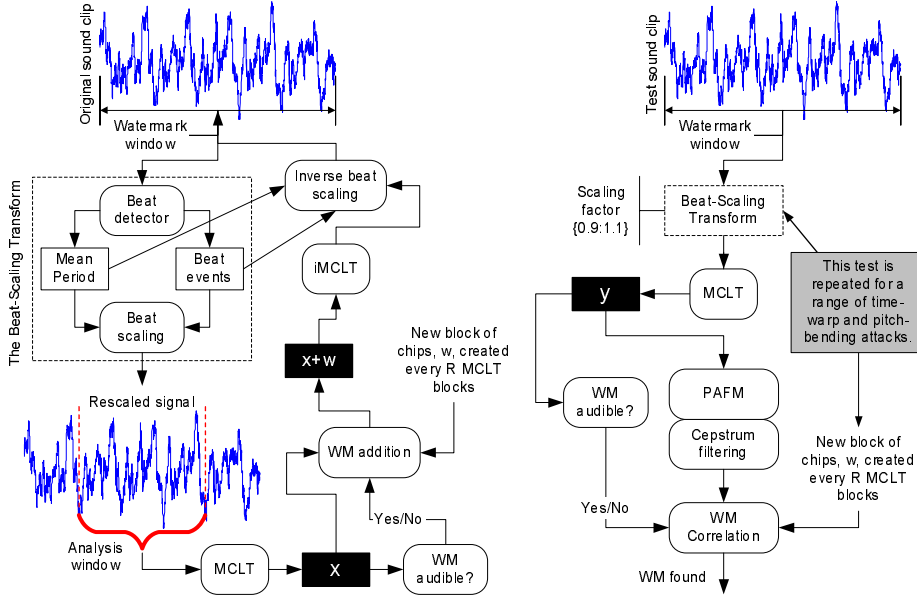
### 3 Hiding Spread-Spectrum Sequences in Audio Signals

In the developed watermarking system, vector  $x$  is composed of magnitudes of several frames of a modulated complex lapped transform (MCLT) [17] in the dB scale. The MCLT is a 2x oversampled DFT filter bank, used in conjunction with analysis and synthesis windows that provide perfect reconstruction. After addition of the WM, we generate the time-domain marked audio signal by combining the vector  $y = x + \delta w$  with the original phase of  $x$ , and passing these modified frames to the inverse MCLT. Typically, WM amplitude  $\delta$  is set to a fixed value in the range  $\{0.5-2.5\}$ dB. For example, our technology has passed the "golden ears test" for  $\delta = 1.5$ dB and a benchmark suite consisting of pop, rock, jazz, classical, instrument solo, and vocals musical pieces. For the typical 44.1kHz sampling, we use a length-2048 MCLT. Only the coefficients within 200-2kHz are marked and only the audible magnitudes in the same sub-band are considered during detection. Sub-band selection aims at minimizing carrier noise effects as well as sensitivity to downsampling and compression. In addition, we use several additional mechanisms (already detailed in [12], [13]) to cope with the problems inherent in SS watermarking.

**Psycho-Acoustic Masking.** (PAFM) The WM detector must correlate only the audible frequency magnitudes with the WM [23], because the inaudible portion of the frequency domain is significantly more susceptible to attack noise. Consequently, the attacker can remove the entire inaudible portion of the spectrum and reduce the proof of authorship, as correlation of silence and any WM equals zero. Such an attack can be effective because the inaudible portion often dominates the frequency spectrum of an audio signal [19]. In order to quantify the audibility of a particular frequency magnitude, we use a simple PAFM model [18] and a modified correlation test that addresses a consequence of using PAFM described in [12].

**Cepstrum Filtering.** The variance  $\sigma_x^2$  of the original signal directly affects the carrier noise in Eqn.1. Audio clips with large energy fluctuations or with strong harmonics are especially bound to produce large  $\sigma_x$ . Thus, we use cepstrum filtering, a nonlinear

processing step to reduce the carrier noise by filtering out the low-frequency components of the signal cepstrum [12]. Cepstrum filtering preserves the WM because it exists in the higher frequencies of the cepstrum due to its randomness. As a result, cepstrum filtering usually halves  $\sigma_x$  – thus, in order to attain the performance of a detector that uses cepstrum filtering, a traditional detector must integrate almost four times more magnitude points.



**Fig. 1.** Block diagram of the WM embedding (left) and detection (right) procedures.

**Improved Watermark Imperceptiveness.** SS WMs can be audible when embedded in the MCLT domain even at low magnitudes (e.g.  $\delta < 1\text{dB}$ ). This can happen in MCLT blocks where certain part of the block (up to 10ms) is quiet whereas the remainder of the MCLT block is rich in audio energy. Since the SS sequence spreads over the entire MCLT block, it can cause audible noise in its quiet portion. To alleviate this problem, we detect blocks with dynamic content where a SS WM may be audible if added. The blocks are identified according to a certain empirically determined criteria. WMs are not embedded nor detected in such blocks. Fortunately, such blocks do not occur often in audio content; in our benchmark set we identified up to 5% of MCLT blocks per WM as potential hazard for audibility.

**Putting It Altogether.** A block diagram that illustrates how the enlisted technologies, jointly with the beat-scaling transform presented in the next Section, are linked into a cohesive system for audio marking is presented in Figure 1. Reference implementation of our data hiding technology on an x86 platform requires 32 KB of memory for code and 100 KB for the data buffer. The data buffer stores averaged MCLT blocks of 12.1

seconds of audio (for a WM length of 11 seconds). Real-time WM detection under these circumstances requires about 15 MIPS. WM encoding is an order of magnitude faster, with smaller memory footprints.

## 4 Preventing De-Synchronization via Beat Detection

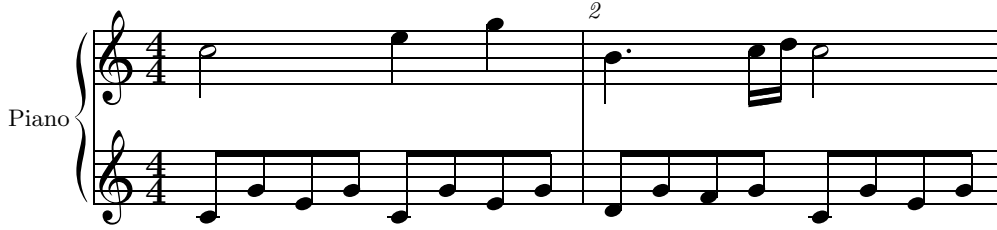
The correlation metrics from Eqns. 1, 2, and 3 are reliable only if the majority of detection chips are aligned with those used in marking. Thus, an adversary can attempt to de-synchronize the correlation by fluctuating time- or frequency-axis scaling within the loose bounds of acceptable sound quality. To prevent from such attacks, we use beat detection as a time-synchronization mechanism and block repetition coding of WM chips combined with a multi-test search (as presented in [12]) to provide robustness to variable pitch-bending.

**The time-warp and pitch-bending attack model.** It is important to define the degree of freedom for time- and frequency-scaling that preserves the relative fidelity of the attacked recording with respect to the original. The HAS is much more tolerable to constant scaling rather than wow-and-flutter. Hence, we adopt the following tolerance levels:  $\gamma_T \leq 0.1$  for constant time-scaling,  $\gamma_F \leq 0.05$  for constant frequency-scaling, and  $\gamma_V \leq 0.01$  for the scaling variance (wow-and-flutter) along both time and frequency. An additional requirement is that the scaling factor does not change more than  $\gamma_V$  within a single detected beat. This requirement is probably the least constraining as variable scaling with fast dynamics is commonly highly intolerable. Finally, note that similar tolerance levels have been adopted within the SDMI call for proposals for music screening technologies [21].

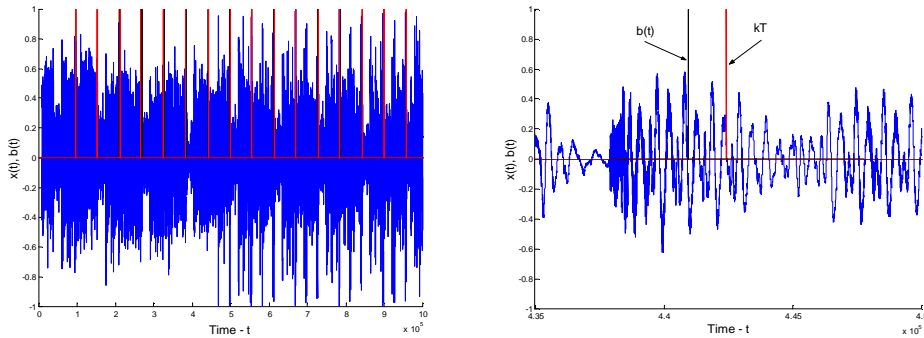
### 4.1 The Beat-Scaling Transform

One of the most robust events in music is its beat. For example, while one can easily remove and/or add instrument solos and voice with potentially perceptible but still not unpleasant effect, the repetitiveness and relatively small variance of the periodicity of the music beat must be preserved in almost any attack if the adversary aims at creating a marketable content. An overwhelming portion of popular and in particular classical music, can be, in general, rigorously defined across almost all cultures, even ancient, using a notation as illustrated in Figure 2. Rarely, music content experiences intentional and significant rhythm speed-up and even when it happens it does not last more than several seconds. In summary, although music has been authored in diverse ways, primarily with different harmonic scales, one characteristic that is predominant for music is the rigorous constraint for accurate rhythm.

We use this characteristic of music to create a transform that aims at enabling synchronicity between the WM detector and mark’s location in an audio clip. For a given audio time-domain signal  $x(t) \in \mathbb{R}^N$ , the beat-scaling transform (BST) initially computes two parameters of clip’s rhythm: (i) the average period between two beats  $\bar{T}$  and (ii) the actual beat pattern – a binary vector  $b(t) \in \{0, 1\}^N$ , where  $b(t) = 1$  denotes a start of a beat or else  $b(t) = 0$ . Details of the EM-based beat detector that we have developed are presented in Section 4.2. Lets denote the sorted list of indices of the vector  $b(t)$  for which  $b(t) = 1$ , as  $t_i, i = 1 \dots K$  where  $K$  is the number of beats in the clip.



**Fig. 2.** First two bars of the sonata in C-major K545 by Mozart. The pulse, or pattern of regular accents, of a musical piece can be broken into individual pulses, or beats. In rhythmic notation, notes are assigned time values by their relation to these beats. The grouping of beats in a piece of music establishes the music's meter. Meter is identified by the time signature, a fractional symbol in which the numerator specifies the number of beats per bar, and the denominator specifies the relative note value assigned to one beat. A time signature of 4/4 indicates four beats per measure and the fourth note is given a value of one beat.



**Fig. 3.** An example of the basic rescaling entities:  $x(t)$  - audio signal,  $b(t)$  - beat events denoted as pulses, and a periodic pulse at  $kT$  samples where  $k \in \mathbb{N}$ .

Next, the content between any two beat events  $x(t)\{t|t_i \leq t < t_{i+1}\}$  is linearly time-warped to  $x(t')$  such that the length of each beat period  $T'$  is constant in the time-warped domain and equal to the average beat period rounded to the nearest multiple of a certain block length  $\vartheta$  (typically,  $\vartheta \in \{512, 1024\}$ ), i.e.  $T' = \lceil \bar{T}/\vartheta \rceil$ . Thus, the distance of an original sample  $x(t = T_0)$  from its preceding beat at  $x(t = t_i)$  changes from  $T_0 - t_i$  to  $(T_0 - t_i)(t_{i+1} - t_i)/T'$  in the new time-warped domain.

Rescaling can be done in many ways: a simple and fast solution is a linear and weighted local interpolation – another more precise solution is to use an anti-aliasing FIR filter with, for example, a Kaiser analysis window. Note that the original clip may have fluctuations in the beat period; nevertheless, the BST flattens this period in the resulting rescaled clip. The inverse BST (iBST) is defined as scaling back the time-warped domain  $t'$  to the original time-domain  $t$ . In order to perform the iBST, the original scale factors for each period between two beat events must be memorized. Figure 3 illustrates the vectors  $x(t)$ ,  $b(t)$ , and the average beat.

## 4.2 Beat Detection

In this section, we describe an EM-based algorithm for beat detection, which we used in one implementation of the BST. Beat detection is performed in three steps. First, the mean period of the beat is estimated from data using a statistical modeling approach. Second, the mean onset of the beat is estimated. Third, the actual onset of the beat for each beat period is estimated. An important comment is that we have applied this algorithm to 12-second music windows with non-significant rhythm changes. Other cases can be trivially reduced to the problem being solved here using traditional segmentation algorithms.

**Mean beat period.** Let  $u_m$  denote the signal energy at frame  $m$ . To compute  $u_m$ , we consider the signal waveform in the time domain, and apply a window function at equally spaced time points indexed by  $m = 1, \dots, M$ .  $u_m$  is the mean squared value of the windowed signal.

We model the beat by assuming that  $u_m$  is approximately periodic in  $m$ , with beat period  $\tau$ . To estimate  $\tau$  we use the following model,

$$u_m = au_{m-\tau} + v_m, \quad (4)$$

where  $v_m$  is i.i.d. Gaussian noise with mean zero and variance  $\sigma^2$ . Hence we have a probabilistic model where  $u_m$  are the observed variables,  $\tau$  is a hidden variable, and  $a$ ,  $\sigma$  are parameters:

$$p(\{u_m\} | \tau) = \prod_m \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(u_m - au_{m-\tau})^2 / 2\sigma^2}. \quad (5)$$

To complete the definition of our model, we must specify a prior distribution  $p(\tau)$  over the beat period. We use a flat distribution  $p(\tau) = \text{const.}$ .

We now use an EM algorithm to estimate the period  $\tau$  and the model parameters. As usual with EM, this is an iterative algorithm, where the E-step updates the sufficient statistics, and the M-step updates the parameter estimates. In our case, the sufficient statistics is the full posterior distribution over the beat period conditioned on the data. It is computed via Bayes' rule,

$$p(\tau | \{u_m\}) = \frac{1}{z} p(\{u_m\} | \tau) p(\tau), \quad (6)$$

where  $z$  is a normalization constant. It can be shown to equal the data distribution,  $z = p(\{u_m\})$ , but since it is independent of  $\tau$  it does not need to be actually computed. This posterior can be computed efficiently for any value of  $\tau$  by observing that its logarithm is the autocorrelation of  $u_m$ ,

$$\log p(\tau | \{u_m\}) = \frac{1}{\sigma^2} \sum_m u_m u_{m-\tau} + \text{const.} \quad (7)$$

and using FFT. The resulting complexity of the E-step is  $\mathcal{O}(M \log M)$ .

The M-step update rules are derived by minimizing the complete data log-likelihood  $E \log p(\{u_m\} | \tau) p(\tau)$ , where the operator  $E$  performs averaging over  $\tau$  w.r.t. the posterior (6). We obtain

$$a = \sum_m u_m E u_{m-\tau} / \sum_m u_m^2, \quad \sigma^2 = \frac{1}{M} \sum_m E (u_m - a u_{m-\tau})^2. \quad (8)$$

As in the E-step, the computations involved in (8) can be performed efficiently using FFT.

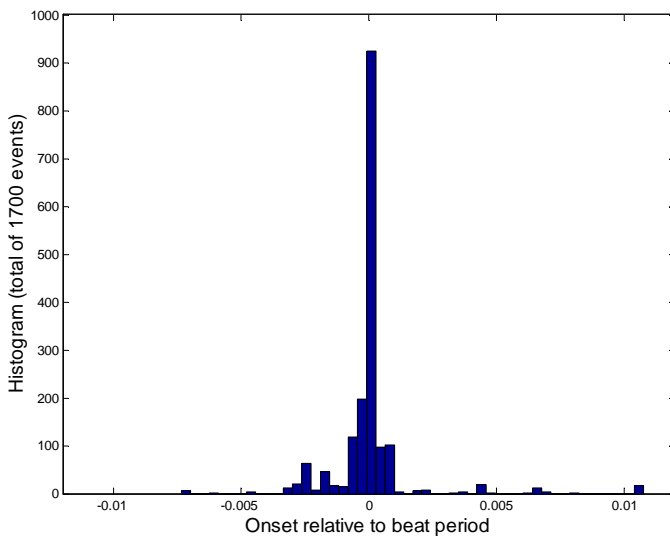
Finally, the beat period is obtained using a MAP estimate,

$$\hat{\tau} = \arg \max_{\tau} p(\tau | \{u_m\}) . \quad (9)$$

Experimentally, the posterior over  $\tau$  turns out to be quite narrow. Below we use  $\tau$  to refer to  $\hat{\tau}$ .

**Mean beat onset.** To compute this quantity, we divide  $u_m$  into consecutive non-overlapping sequences of length  $\tau$ . Denote sequence  $i$  by  $(u_1^i, \dots, u_{\tau}^i)$ , where  $u_n^i = u_{(i-1)\tau+n}$  and  $n = 1, \dots, \tau$ . We then average over those sequences. Denote the average sequence by  $(\bar{u}_1, \dots, \bar{u}_{\tau})$ , then the mean onset  $\bar{l}$  is its maximum,

$$\bar{l} = \arg \max_{1 \leq n \leq \tau} \bar{u}_n . \quad (10)$$



**Fig. 4.** Histogram of the relative onset of the detected beats in a typical pop-song with respect to additive white gaussian noise conceived as  $\mathcal{N}(0, \sigma_x * 0.1)$ . More than 99% of beats were detected with sub-percent scaling accuracy.

**Actual beat onset.** This quantity is computed for each of the  $\tau$ -long sequences above. We assume that the onset time  $l$  for a given sequence may deviate from the mean onset  $\bar{l}$  by as much as 10% of the beat period. Hence, we search for  $l_i$ , the beat onset time for sequence  $i$ , within the corresponding interval,

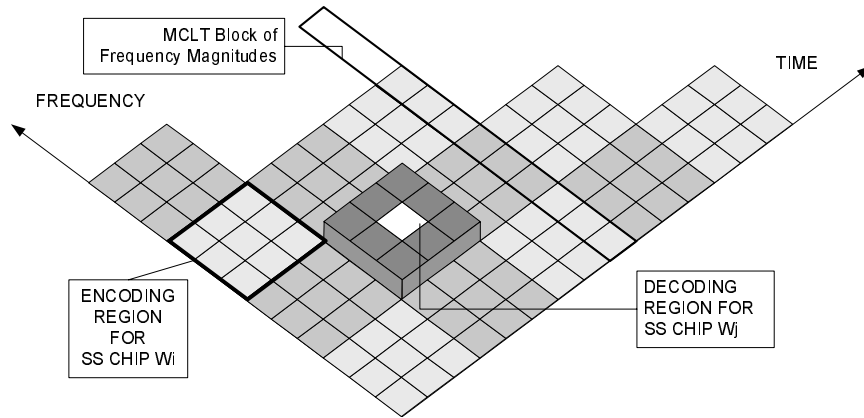
$$l_i = \arg \max_{\bar{l} - \tau/10 \leq n \leq \bar{l} + \tau/10} u_n^i . \quad (11)$$

The onset times  $l_i$  are converted back to the time domain and form the output beat signal. One characteristic of the deployed beat detector is robustness with respect to noise and de-synchronization – the beats are retrieved within 1% scaling accuracy. An example of its performance is presented in Figure 4.

### 4.3 Marking Audio with the BST

The overview of the WM embedding process using the BST is illustrated in Figure 1. A distinct and independently detectable mark is hidden into each 12 seconds of audio. The 12-sec window is first rescaled using the BST. This transformation rescales the rhythm of the clip to a fixed period. We add the WM in the time-warped domain using block repetition codes as described in [12]. To create the final marked content, we perform the inverse BST with the scaling factors inverse to the ones induced during the forward BST. Depending on the actual scaling algorithm, the noise due to rescaling is negligible with respect to the noise induced by the imperceptible WM. The key differences with respect to the work done in [12] are:

- **The redundancy deployed without a beat detector must synchronize the watermark detector with respect to its location in the clip throughout the entire length of the watermark, while in the opposite case it must provide robustness to de-synchronization independently and exclusively within two consecutive beat events.** This results in approximately 4 times less redundancy along the time domain.
- **Watermarks can be placed at well defined positions;** for example, at a beat event. Clearly, instead of searching through the entire audio clip in exhaustive fashion as in [12], the detector can focus its search only at the beginning of each beat period which results in an order of magnitude faster WM detection.



**Fig. 5.** An example of block repetition coding along the time and frequency domain of an audio clip. Each block is encoded with the same bit, whereas the detector integrates only the center locations of each region.

The block repetition coding of the WM is performed by processing the time-warped signal  $x(t')$  starting from a certain beat event with overlapping MCLT windows as follows. We represent a SS sequence as a matrix of chips  $W = \{w_{ij}\}, i = 1..N_F, j = 1..N_T$ , where  $N_F$  is the number of chips per MCLT block and  $N_T$  is the number of blocks of  $N_F$  chips per WM. Within a single MCLT block, each chip  $w_{ij}$  is spread over a

sub-band of  $F_i$  consecutive MCLT coefficients. Chips embedded in a single MCLT block are then replicated along the time axis within consecutive  $T_j$  MCLT blocks. An example of how redundancies are generated is illustrated in Figure 5 (with fixed parameters  $F_i = 3, T_j = 3$  for all  $i$  and  $j$ ). Widths of the encoding regions  $F_i, i = 1..N_F$  are computed using a geometric progression [12]. Within a region of  $F_i T_j$  samples watermarked with the same chip  $w_{ij}$ , only the center  $\eta_F \eta_T$  samples are integrated in Eqn.1 where  $\eta_F < N_F$  and  $\eta_T < N_T$ . It is straightforward to prove that such generation of encoding and decoding regions guarantees that regardless of a limited wow-and-flutter, the correlation test is performed in perfect synchronization.

#### 4.4 Detecting Watermarks using the BST

Detection of WMs embedded as described in the previous Subsection is performed using a multi-test search which exhaustively searches the solution space within the adopted degrees of attack freedom (see Section 4). The interaction of DSP functions involved in the detection is presented in Figure 1. The algorithm for detection is outlined using the following pseudo-code:

1	pointer = $t_1$ (for def. of $t_i$ see Section 4.1).
2	load buffer with $L$ samples of the time-domain signal $x(t)$ starting from pointer
3	<b>for</b> time.scaling = $-\gamma_T$ to $+\gamma_T$ step $\gamma_V/2$
4	$x(t') = \text{scale}(\text{BST}(\text{buffer}), \text{time.scaling})$
4	<b>for</b> frequency.scaling = $-\gamma_F$ to $+\gamma_F$ step $\gamma_V/2$
5	correlate $MCLT(x(t'))$ with $w$ scaled according to time.scaling and frequency.scaling
6	<b>if</b> ( $w$ found in buffer) <b>then</b> pointer = position of next WM
7	<b>else</b> pointer = next beat $t_i$
8	<b>goto</b> [2]

The search algorithm initially loads a buffer of time-domain samples of the input audio clip  $x(t)$  starting from the first detected beat at time  $t_1$ . The length of the buffer equals  $L$ , i.e. the length of the WM. Next, for each scaling test point in the time domain,  $\tau = \text{time.scaling}$ , the content of the buffer is scaled, first using BST and then, an additional linear scaling occurs with a constant scaling factor equal to  $\tau$ . The resulting time-warped content  $x(t')$  is then converted to the MCLT domain and the frequency magnitudes of this domain are correlated with different scalings of the searched WM. The scalings are such that they create a grid over  $\{\tau, -\gamma_F.. \gamma_F\}$  with  $\gamma_V/2$  minimal distance between test-points. Due to the block repetition encoding of WM chips, each test at  $\{\tau, F\}$  can detect a WM if the actual scaling of the clip is within the  $\{\tau - \gamma_V/2.. \tau + \gamma_V/2, F - \gamma_V/2.. F + \gamma_V/2\}$  region. The test that yields the greatest correlation is compared to the detection threshold to determine WM presence. If WM is found, the entire buffer is reloaded with new time-domain coefficients starting from the first detected beat after the WM. Otherwise, the content of buffer is shifted to the next beat event  $t_i$  and the procedure is repeated.

In a typical implementation, for  $\gamma_V = 0.02$ , in order to cover  $\gamma_T = 0.1$  and  $\gamma_F = 0.05$ , the WM detector computes 105 different correlation tests. Note that the main incentive for providing such a mechanism to enable synchronization is the fact that, within the length of the WM, the adversary really cannot move away from the selected constant time

and frequency scaling more than  $\gamma_V/2$ ; such a change would induce intolerable sound quality. If the attacker is within the assumed attack bounds, the described mechanism enables the detector to conclude whether there is a WM or not in the audio clip based on the SS statistics from Eqn.1 and regardless of the presence of the attack.

#### 4.5 Empirical Robustness

We have tested our proposed watermarking technology using a composition of common sound editing tools and malicious attacks, including all tests defined by the Secure Digital Music Initiative (SDMI) industry committee [21]. In particular, we have addressed various time-warp and pitch-bending attacks with superimposed variance using high- and low-quality warping tools – all of them unsuccessful. We tested the system against a benchmark suite of eighty 15-sec audio clips, which included: jazz, classical, voice, pop, instrument solos (accordion, piano, guitar, sax, etc.), and rock. In that dataset, there were no errors, and we estimated the error probability to be well below  $10^{-6}$ . Error probabilities decrease exponentially fast with the increase of WM length, so it is relatively easy to design a system for error probabilities below  $10^{-9}$ , for example. An analysis of the security of embedded WMs is presented in the next Section.

### 5 Effect of the BST on the Watermark Estimation Attack

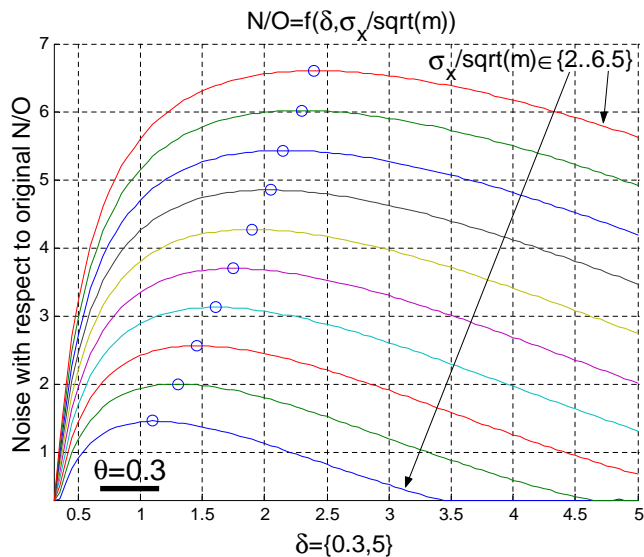
In this section, we evaluate the security of our watermarking mechanisms with respect to the estimation attack. Thus, in this section we explore the fundamental limits in chip replication with respect to WM security and apply these results to the developed audio WM mechanisms. In order to simplify the formal description of block repetition codes in our audio WM codec, we define the following WM setup. The marked signal  $y$  is created by adding the WM with certain magnitude  $\delta$  to the original:  $y = x + \delta w$ ,  $w \in \{-1\}^m, \{1\}^m$ . Vectors  $y$  and  $x$  have  $N = m \times n$  samples, whereas  $w$  has  $n$  chips, each of them replicated successively  $m$  times. The WM detector correlates the averages of the central  $m_o < m$  elements of each region marked with the same chip.

**Theorem 1.** *Given a set of  $m$  samples of  $x$ , marked with the same chip  $w_i$  such that  $y_{(i-1)m+j} = x_{(i-1)m+j} + \delta w_i$ ,  $1 \leq j \leq m$ , the optimal estimate  $v_i$  of the hidden WM chip  $w_i$  is given as:*

$$v_i = \text{sign} \left( \sum_{j=1}^m (x_{(i-1)m+j} + \delta w_i) \right). \quad (12)$$

See Lemma 1 in [14] for proof. Note that  $v \in \{\pm 1\}^N$ . We construct the estimation attack by subtracting an amplified WM estimate  $\alpha v$  from the marked content  $y$  as:  $z = y - \alpha v$ . The goal of the adversary is to induce  $\alpha$  such that the expected correlation value drops below the detection threshold. Since the maximal value of the amplification factor  $\alpha$  depends solely on the imperceptiveness of the attack, we have constructed two mechanisms that: (a) maximize  $\alpha$  with respect to the WM amplitude  $\delta$  and (b) perform an optimal *undo* of the estimation attack and therefore force the adversary to superimpose additional noise to disable the *undo* operator (both techniques are detailed

in [15]). Since WM synchronization via beat detection requires a factor of 4 fewer redundant chips ( $m$ ), the impact on the amount of noise that the adversary adds with respect to the original (N/O) recording  $x$  can be observed from Figure 6 which depicts the dependency of N/O with respect to  $\delta$  for realistic values of  $\frac{\sigma_x}{\sqrt{m}} \in \{2 \dots 6.5\}$ . Optimal values  $\delta(\sigma_x)$ , which result in maximal N/O, are depicted using the  $\{\circ\}$  symbol. Note that a reduction in  $m$  for a factor of four, almost doubles the amount of noise that the adversary needs to add to remove the WM with respect to a scheme that does not rely on beat detection.



**Fig. 6.** Diagram of the dependency of  $N/O \equiv E[|z_i - x_i|]$  with respect to  $\delta$  for given  $\frac{\sigma_x}{\sqrt{m}} \in \{2 \dots 6.5\}$  and a detection threshold  $\theta = 0.3$ . Details on how these curves are derived, can be found in [15].

## 6 Conclusions

In this paper, we introduce beat detection as a crucial technology along with block redundant coding that combats the two most effective attacks on watermarking systems: de-synchronization and watermark estimation. As a result, we have achieved robustness of spread-spectrum watermarks augmented in audio clips to almost arbitrary constant time-warp and pitch-bending and wow-and-flutter of up to 1%. The adversary can remove the watermark by subtracting an estimate of the watermark from the signal with an amplitude in excess of 6dB with respect to the host. Such an attack vector typically affects substantially the fidelity of the "pirated" recording.

## References

1. Anderson, R.J., Petitcolas, F.A.P.: On the limits of steganography. Journal on Selected Areas in Communications, vol.16, pp.474-481, IEEE (1998).

2. Bassia, P., Pitas, I.: Robust audio watermarking in the time domain. EUSIPCO, vol.1. Rodos, Greece, IEEE (1998).
3. Boneh D., Shaw J.: Collusion secure fingerprinting for digital data. Transactions on Information Theory, vol.44, pp.1897–1905, IEEE (1998).
4. Chen, B., Wornell, G.W.: Digital watermarking and Information embedding using dither modulation. Workshop on Multimedia Signal Processing, Redondo Beach, CA, IEEE (1998).
5. Cox, I.J., Kilian, J., Leighton, T., Shamoon, T.: A secure, robust watermark for multimedia. Information Hiding Workshop, Cambridge, UK, (1996).
6. Dempster A.P., Laird N.M., Rubin D.B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society, vol.39, no.1, pp.1–38, (1977).
7. Gruhl, D., Lu, A., Bender, W.: Echo hiding. Information Hiding Workshop, Cambridge, UK, (1996).
8. Haitsma J.A., Kalker T., Oostveen J.: Robust Audio Hashing for Content Identification. International Workshop on Content Based Multimedia and Indexing, Brescia, Italy, 2001.
9. Hartung, F., Su, J.K., Girod, B.: Spread spectrum watermarking: malicious attacks and counter-attacks. Security and Watermarking of Multimedia Contents, San Jose, CA, SPIE (1999).
10. Jessop P.: The Business Case for Audio Watermarking. IEEE International Conference on Acoustics, Speech and Signal Processing, vol.4, pp.2077–2080, Phoenix, AZ, (1999).
11. Katzenbeisser S., Petitcolas, F.A.P., (eds.): Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, Boston (2000).
12. Kirovski D., Malvar H.: Robust Covert Communication over a Public Audio Channel Using Spread Spectrum. Information Hiding Workshop, Pittsburgh, PA, (2001).
13. Kirovski D., Malvar H.: Robust Spread-Spectrum Audio Watermarking. IEEE International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, UT, IEEE (2001).
14. Kirovski D., Malvar H., Yacobi Y.: A Dual Watermarking and Fingerprinting System. Microsoft Research Technical Report, (2001).
15. Kirovski D., Malvar H.: Embedding and Detecting Spread Spectrum Watermarks under The Estimation Attack. International Conference on Acoustics, Speech, and Signal Processing, Orlando, FL, IEEE (2002).
16. Linnartz, J.P., van Dijk, M.: Analysis of the sensitivity attack against electronic watermarks in images. Information Hiding Workshop, Portland, OR, (1998).
17. Malvar H.: A modulated complex lapped transform and its application to audio processing. International Conference on Acoustics, Speech, and Signal Processing, Phoenix, AZ, IEEE (1999).
18. Malvar, H.S.: Auditory masking in audio compression. Greennebaum, K. (ed.): Audio Anecdotes. Kluwer, New York, (2001).
19. Neubauer, C., Herre, J.: Digital watermarking and its influence on audio quality. 105th Convention, San Francisco, CA. Audio Engineering Society (1998).
20. Recording Industry Association of America. See <http://www.riaa.org>.
21. Secure Digital Music Initiative. See <http://www.sdmi.org>.
22. Su, J.K., Girod, B.: Power-spectrum condition for energy-efficient watermarking. International Conference on Image Processing, Yokohama, Japan, IEEE (1999).
23. Swanson, M.D., Zhu, B., Tewfik, A.H., Boney, L.: Robust audio watermarking using perceptual masking. Signal Processing, vol.66, pp.337–355, (1998).
24. Szepanski, W.: A signal theoretic method for creating forgery-proof documents for automatic verification. In: Carnahan Conf. on Crime Countermeasures, Lexington, KY, pp.101–109, (1979).
25. van Trees, H.L.: Detection, Estimation, and Modulation Theory. Part I, New York: John Wiley and Sons, (1968).