Insertion, deletion robust audio watermarking: a set theoretic, dynamic programming approach

Andrew Nadeau, Gaurav Sharma

ECE Dept, University of Rochester, Rochester NY, USA

ABSTRACT

Desynchronization vulnerabilities have limited audio watermarking's success in applications such as digital rights management (DRM). Our work extends (blind-detection) spread spectrum (SS) watermarking to withstand time scale desynchronization (insertion/deletions) by applying dynamic programming (DP). Detection uses short SS watermark blocks with a novel $O(N \log N)$ correlation algorithm that provides robustness to time shifts and the resulting offsets to the watermarking domain transform. To withstand insertion/deletion, DP techniques then search for sequences of blocks instead of detecting SS watermarks individually. This allows DP techniques to govern the tradeoff between long/short SS blocks for non-desynchronization/desynchronization robustness. However, high dimensional searches and short SS blocks both increase false detection rates. Consequently, we verify detections between multiple, simultaneously embedded watermarks. Embedding multiple watermarks while considering host interference, compression robustness, and perceptual degradation to the host audio is a complex problem, solved using a set theoretic embedding framework. Proposed techniques improve performance by multiple orders of magnitude compared with naive SS schemes. Results also demonstrate the tradeoff between non-desynchronization/desynchronization robustness.

1. INTRODUCTION

Robust watermarking is a complex problem which has inspired practical and theoretical work in fields ranging from signal processing and communications to information theory. In applications such as DRM, watermarking presents a solution to embed traceable signals in digital signals before distribution. To detect copyright infringement, watermarks must remain detectable after common operations such as MP3 compression. Inadvertent watermark desynchronization can also occur if portions of watermarked audio are used in musical mash-ups or crowdsourced into other users content, e.g. www.zooppa.com for advertising jingles. Additionally, adversaries may attempt to maliciously remove or desynchronize watermarks. Robustness to these attacks is critical for many watermarking applications. We differentiate non-desynchronization and desynchronization robustness by how well distortion fits an additive noise model. For example, MP3 compression adaptively quantizes audio in an adaptive time-frequency transform domain, but does not change the time scale of the original audio (assuming restoration of any MP3 downsampling). Because quantization in a transform domain is (signal dependent) additive noise, MP3 compression typifies non-desynchronizing distortion. Desynchronizing perturbations encompass a wider range of modifications to the watermarked signal, including cases where the original time scale of the audio is not preserved. This is challenging because many common concepts from communications apply only for additive distortion or assume prior synchronization.

Two current techniques for robust watermarking rely on exhaustive search for embedded synchronization marks, or specific invariant features of the original audio that survive desynchronization. Intrinsic features used previously include musical beats for robustness to TSM (time scale modification),[1] and distinct voiced segments in speech signals.[2] However these techniques rely on the presence of specially developed features and must consider the reliability of those features after desynchronization. Difficulty modeling these concerns partly explains why theoretical work has concentrated on exhaustive search methods.[3,4] The DP technique we propose in this paper builds on practical exhaustive search methods.[5,6] Specifically, the proposed method first uses correlation detectors for short SS watermark blocks followed by DP techniques to string together sequences of detections for greater reliability. Performed in a brute force fashion, this search over all possible insertion and

Further author information: (Send correspondence to A. Nadeau)
A. Nadeau: E-mail: andrew.nadeau@rochester.edu

Media Watermarking, Security, and Forensics 2013, edited by Adnan M. Alattar, Nasir D. Memon,
Chad D. Heitzenrater, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 8665, 866503
© 2013 SPIE-IS&T · CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2005462

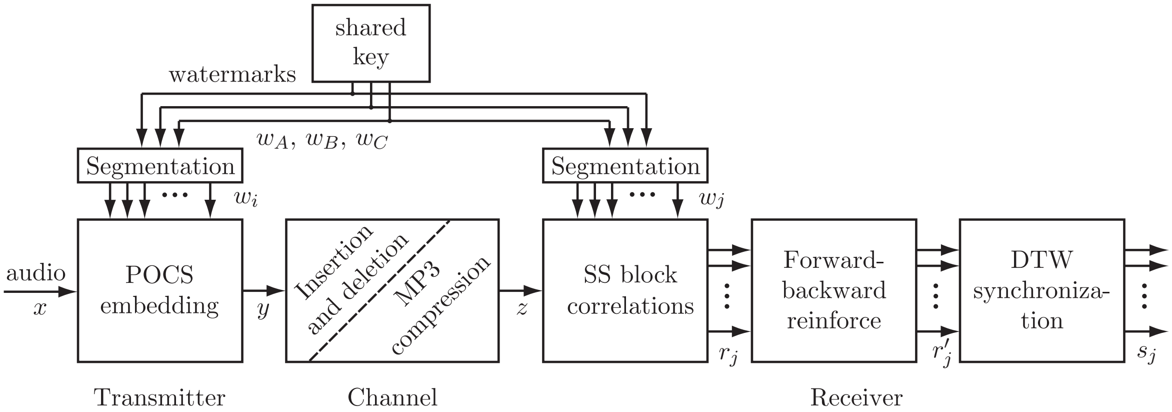


Figure 1. Transmitter, channel and receiver for the proposed framework. Audio signals are denoted x, y, and z. Watermark blocks segmented from the SS chip sequences w_A, w_B and w_C, are denoted w_i and w_j. r_j and r'_j are correlation signals, and s_j are *synchronization points* where each watermark block is detected.

deletion permutations is, however, computationally intractable and increases false positive errors; both are issues that typically hold back exhaustive search methods. The dynamic time warping (DTW) algorithm allows the proposed scheme to efficiently search over all possible combinations of insertions and deletions. This efficiency decreases the need to limit the search space, e.g. repeated synchronization codes, which can cause security vulnerabilities, such as watermark estimation and removal.[5] Additionally, our scheme reduces false positives by cross validating the overlapping structure of multiple embedded watermarks.

In addition to desynchronization, watermark robustness also considers non-desynchronizing perturbations and host interference. Our scheme balances the conflict between robustness to these operations and perceptual fidelity by embedding multiple watermarks using a set theoretic framework.[7] This framework determines a watermarked signal as a point in the high dimensional signal space that jointly meets the constraints of acceptable perceptual quality and detectability for each of the individual watermarks in the presence of MP3 compression. MP3 compression is used as an archetype for non-desynchronizing perturbations because of the large body of research aimed at encoding at the lowest possible rate (or equivalently, increasing the limit of MP3 quantization noise that can be introduced) while maintaining perceptual fidelity[8] a constraint that generally applies to malicious attacks.

DTW based synchronization has been previously demonstrated for non-blind watermarks,[9,10] where the original unwatermarked audio signal is assumed to be available at the detector and synchronized with the received audio based on features of the audio signal, without reference to the watermark. Also in the non-blind detection category, Ref. 11 proposed embedding of watermarks via modulation of the time scale and detection by using DTW to align with the original audio. The method presented in this paper, in contrast with these prior methods, accomplishes synchronization for the blind-detection scenario by integrating DTW with blind SS detection.

2. INSERTION DELETION ROBUST WATERMARKING

2.1 Framework

To address robustness to both desynchronization and non-desynchronization perturbations, our watermarking scheme uses the framework shown in Fig. 1. The transmitter embeds three SS watermarks,w_A, w_B, and w_C, into the original audio x for subsequent cross-validation at the detector. w_A, w_B, and w_C are composed of pseudo random ±1 chips, and known to both the transmitter and receiver through a shared secret key. The watermarked audio, produced using the set theoretic framework, is denoted by y.

Because time-frequency resolution is critical to how humans perceive distortion in audio,[8] and consequently influences how MP3 compression distorts signals, watermark embedding occurs in a transform domain. Our method uses the same PQMF (polyphase quadrature modulated filterbank) as MP3 compression,[12] in order to leverage well-developed perceptual models as well as better estimate distortion caused by MP3 compression. The PQMF transform splits the audio into K equal bandwidth subband signals, each downsampled by a factor K from the original sampling frequency. Subband signals are distinguished by the subscript k that indexes subbands. The downsampled timescale is indexed by n. A near perfect-reconstruction inverse PQMF filterbank reconstructs the time domain watermarked audio after the watermark chips are embedded.

The next module, the channel, models desynchronization and non-desynchronizing operations. We choose insertion, deletion, and MP3 compression as typical challenges of both robustness classes mentioned. Insertion and deletions desynchronize detection by destroying the correspondence between where watermarks are embedded in y and the expected detection locations in z. Figure 5 shows one of the more severe sets of insertions and deletions randomly generated by our prototype channel. After potential desynchronization, the channel applies MP3 compression to the audio, and a decompressed version z, is passed on to the receiver.

The receiver is shown in Fig. 2. Classical detection of long SS watermarks is inappropriate when desynchronization is a concern. Consequently, before correlation, the receiver segments w_A, w_B, and w_C into short blocks w_j. Correlations are computed efficiently using an FFT method robust to time shifts. Forward backward reinforcement and DTW counteract decreasing interference rejection of shorter SS blocks by searching for the appropriate series of detection peaks in the correlation signals r_j, to locate the *synchronization points*, labeled as s_j in Fig. 2.

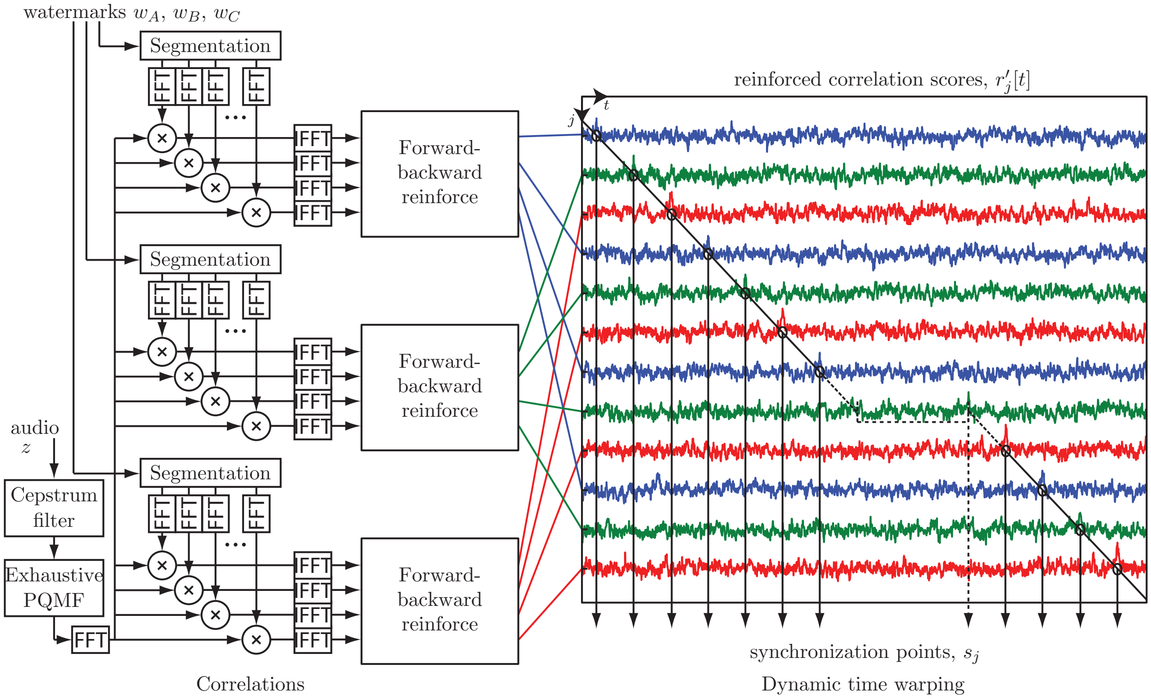

watermarks w_A, w_B, w_C
Segmentation
FFT
IFFT
Forward-backward reinforce
Segmentation
Forward-backward reinforce
audio z
Cepstrum filter
Exhaustive PQMF
Segmentation
FFT
Forward-backward reinforce
Correlations
reinforced correlation scores, $r'_j[t]$
t
j
synchronization points, s_j
Dynamic time warping

Figure 2. DP detection searches correlations (plotted) for a maximal sequence of detection peaks whose indices (circles) are the *synchronization points* s_j. A conceptual detection window bounded by minimum separation between block detections $t_{\min,\mathrm{DTW}}$, is shown (dashed lines). Synchronization and feasibility complications within the correlation calculations are discussed later and not depicted for simplicity.

DTW synchronization motivates our embedding constraints, forward backward reinforcement, and cross validation. To appreciate the need for these auxiliary steps, DTW is conceptualized as selecting the highest correlation peak occurring within a detection window for each watermark block, as in Fig. 2. Detection windows for each s_j are bounded by the minimum separation from the neighboring s_j of the optimal DTW sub-sequences in either direction. Errors occur when the correlation score for an embedded watermark block is not the maximum within its detection window. The set theoretic embedding methods account for this local maximum requirement by incorporating watermark SNR constraints into informed embedding. DTW detection also disregards local information. For example, a detection window for the last genuine watermark block present before an insertion will span from the previous block detection over the entire inserted segment to the detection of the following block. To favor expected locations at either end of an insertion where local information suggests a block would fit in sequence, forward backward reinforcement spreads high correlation scores to r_j of nearby blocks at the expected separation. However, reinforcing expected block spacing degrades cross validation which relies on spacing to verify genuine detections. To balance this tradeoff between false negatives/positives during detection/verification, forward-backward reinforcement is performed independently for the multiple watermarks while cross validation works across all watermarks as in Fig. 4. Validation is critical because DTW will always return a maximum sequence of detections even for unwatermarked z, as results in Fig. 5(b) show.

2.2 Set Theoretic Embedding

Set theoretic embedding uses convex sets of audio signals, $\mathcal{S}_{\text{imperceptible}}$ and $\mathcal{S}_{\text{detectable},i}$, to enforce perceptual transparency and watermark robustness, respectively. The following section defines these sets and corresponding projections onto convex sets (POCS)[7] that iteratively produce a watermarked signal, y from the intersection.

Projections onto $\mathcal{S}_{\text{imperceptible}}$ enforce watermark imperceptibility using a perceptual model[13] applied to x. The model calculates thresholds, $\beta_k[n]$, for each PQMF sample below which distortion is inaudible. These thresholds define the convex set of perceptually acceptable signals, $\mathcal{S}_{\text{imperceptible}}$, as a high dimensional box around the audio subbands x_k, which y_k should not breach:

$$\mathcal{S}_{\text{imperceptible}} \equiv \{y : |y_k[n] - x_k[n]| \leq \beta_k[n]\}\,. \tag{1}$$

Projections onto this box enforce hard limits on embedding distortion to produce:

$$y'_k[n] = \begin{cases} x_k[n] + \beta_k[n], & \text{if} \quad y_k[n] > x_k[n] + \beta_k[n], \\ x_k[n] - \beta_k[n], & \text{if} \quad y_k[n] < x_k[n] - \beta_k[n], \\ y_k[n], & \text{otherwise.} \end{cases} \tag{2}$$

The remaining convex sets, $\mathcal{S}_{\text{detectable},i}$ segment w_A, w_B, and w_C into I blocks indexed by i to insure watermarks are detectable throughout a watermarked z. Detectability depends on SNR between genuine correlation peaks and nearby false positives within detection windows. Correlation* for each watermark block w_i, are

$$r_i[n] = \frac{1}{\sqrt{BK}} \sum_{k=1}^{K} \sum_{m=1}^{B} w_{i,k}[m] \cdot \text{ceps}\left\{z_k[n+m]\right\}. \tag{3}$$

B is the number of chips in $w_{i,k}$, in each PQMF subband, and ceps$\{\cdot\}$ denotes adaptive cepstrum filtering used to reduce host interference before correlation.[5] Cepstrum filtering removes a few strong low cepstral coefficients which approximately corresponds to and an adaptive whitening filter. Each $w_{i,k}$ is embedded at sample $n_{i,\text{true}}$ such that $r_{i,\text{true}} = r_i\left[n_{i,\text{true}}\right]$ is the genuine detection peak. Forward-backward reinforcement is neglected during embedding because it also depends on SNR such that if $r_{i,\text{true}}$ is a local maximum the appropriate neighboring correlations will be reinforced. $\mathcal{S}_{\text{detectable},i}$ enforces robustness by considering distributions of r_i calculated from z. $r_{i,\text{true}}$ and nearby false positive r_i are modeled as uncorrelated†, normally distributed random variables with means: $\text{E}\left\{r_{i,\text{true}}\right\}$ and 0, and variances: $\sigma^2_{i,\text{true}}$ and $\sigma^2_{i,\text{false}}$ respectively. The *signal* amplitude term for the SNR is the expectation:

$$\text{E}\left\{r_{i,\text{true}}\right\} = \frac{1}{\sqrt{BK}} \sum_{k=1}^{K} \sum_{m=1}^{B} w_{i,k}[m] \cdot \widehat{\text{ceps}}\left\{y_k[n_{i,\text{true}}+m]\right\}, \tag{4}$$

*Correlations in (3) neglect possible delay in z that can offset the transform producing z_k. Therefore, (3) only applies at the transmitter where the time scale is fixed.

†Forward backward reinforcement introduces correlation in the r'_j in Fig. 2 but the random ±1 nature of $w_{i,k}$ results in no correlation between consecutive r_i in (3).

where $\widehat{\text{ceps}}\{\cdot\}$ approximates $\text{ceps}\{\cdot\}$ by fixing the adaptive filter response of $\text{ceps}\{x_k[n]\}$ and assuming subsequent changes on x will have little effect. Neglecting dependence between host interference and MP3 distortion, variances $\sigma^2_{i,\text{false}}$ and $\sigma^2_{i,\text{true}}$ sum to produce the *noise* standard deviation for the SNR, $\sigma_i = \sqrt{\sigma^2_{i,\text{false}} + \sigma^2_{i,\text{true}}}$. For informed embedding, host interference only contributes to $\sigma^2_{i,\text{false}}$, and is approximated,

$$\sigma^2_{\text{host},i} = \frac{1}{BK}\sum_{k=1}^{K}\sum_{m=1}^{B}\widehat{\text{ceps}}\left\{x_k\left[n_{i,\text{true}}+m\right]\right\}^2 . \tag{5}$$

Distortion from MP3 compression adds variance throughout r_i, contributing to both $\sigma^2_{i,\text{false}}$ and $\sigma^2_{i,\text{true}}$. The additional variance is estimated as

$$\sigma^2_{\text{MP3},i} = \frac{1}{BK}\sum_{k=1}^{K}\sum_{m=1}^{B}\sum_{\ell=1}^{\frac{N}{K}}\left|T_k\left[\ell, n_{i,\text{true}}+m\right]\right|^2 \cdot \frac{\left(\widehat{\text{ceps}}\cdot\Delta_k[\ell]\right)^2}{12}, \tag{6}$$

using MP3 quantization step sizes $\Delta_k[\ell]$, taken from an MP3 file of x, with ℓ indexing the MDCT coefficients from each subband. $T_k\left[\ell, n\right]$ represents the additional modulated discrete cosine transform (MDCT) that MP3 applies after the PQMF and before quantization. Because $\widehat{\text{ceps}}\{\cdot\}$ is implemented in the MDCT domain, gains can be applied directly to Δ_k corresponding to the abuse of notation in (6). Combining $\sigma^2_{i,\text{false}}$ and $\sigma^2_{i,\text{true}}$, the *noise* standard deviation for the SNR becomes

$$\sigma_i = \sqrt{\left(\sigma^2_{\text{host},i} + \sigma^2_{\text{MP3},i}\right) + \sigma^2_{\text{MP3},i}}. \tag{7}$$

The resulting SNR constraint for each w_i is

$$\begin{aligned}\mathcal{S}_{\text{detectable},i} &\equiv \left\{y : 20\log_{10}\frac{\mathrm{E}\left\{r_{i,\text{true}}\right\}}{\sigma_i} \geq \text{SNR}\right\}\\ &\equiv \left\{y : \mathrm{E}\left\{r_{i,\text{true}}\right\} \geq \tau_i\right\},\end{aligned} \tag{8}$$

where $\tau_i = 10^{\frac{\text{SNR}}{20}}\sigma_i$ is an adaptive threshold that adjusts to varying interference for each $w_{i,k}$, and $\mathrm{E}\left\{r_{i,\text{true}}\right\}$ is given in (4). The resulting detectability projections are found by simplifying (4) to a dot product[‡] and substituting into (8). This reveals that each $\mathcal{S}_{\text{detectable},i}$ bounds $\widehat{\text{ceps}}\left\{y_k\right\}$ away from the origin to the far side of a hyperplane, $\frac{1}{B}w^{\mathrm{T}}_{i,k}\cdot\widehat{\text{ceps}}\left\{y_k\right\} = \tau_i$. Using $\widehat{\text{ceps}}^{-1}\{\cdot\}$ for the inverse frequency response of $\widehat{\text{ceps}}\{\cdot\}$, the resulting projections for each $\mathcal{S}_{\text{detectable},i}$ are

$$y'_k = \begin{cases}\widehat{\text{ceps}}^{-1}\left\{\widehat{\text{ceps}}\left\{y_k\right\} + w_{i,k}\cdot\left(\tau_i - \mathrm{E}\left\{r_{i,\text{true}}\right\}\right)\right\}, & \text{if } \mathrm{E}\left\{r_{i,\text{true}}\right\} < \tau_i\\ y_k, & \text{otherwise.}\end{cases} \tag{9}$$

To produce y from the initial audio signal x, each POCS iteration first applies the series of projections, (9) for all watermark blocks not meeting detectability constraints. Each iteration then applies (2) to enforce perceptual quality and passes the resulting audio signal to the next iteration.

2.3 Robust Transform Domain Correlations

At the receiver, calculating each correlation $r_j[t]$, requires exhaustive search over PQMF transform analysis window offsets to preserve detection peaks. Our technique uses an auxiliary FFT domain to reduce complexity. This technique consolidates calculations between both watermark blocks and subbands, and leverages sparsity of band limited subband signals. Notation uses $X[\nu] = \mathcal{F}_N\left\{x[t]\right\}$ and $x[t] = \mathcal{F}^{-1}_N\left\{X[\nu]\right\}$ for a N point FFT and its inverse; ν indexes discrete frequency; W^* denotes the complex conjugate of W; $*$ represents convolution; and $\delta_N\left\{\nu\right\}$ is a unit impulse train defined as 1 for $\nu \pmod{N} = 0$, and 0 otherwise. Because we leave PQMF subband oversampled, t from the original audio signal indexes time rather than n.

[‡] Dot product in (9) assumes $w_{i,k}$ and $\widehat{\text{ceps}}\left\{y_k\right\}$ are rearranged into column vectors

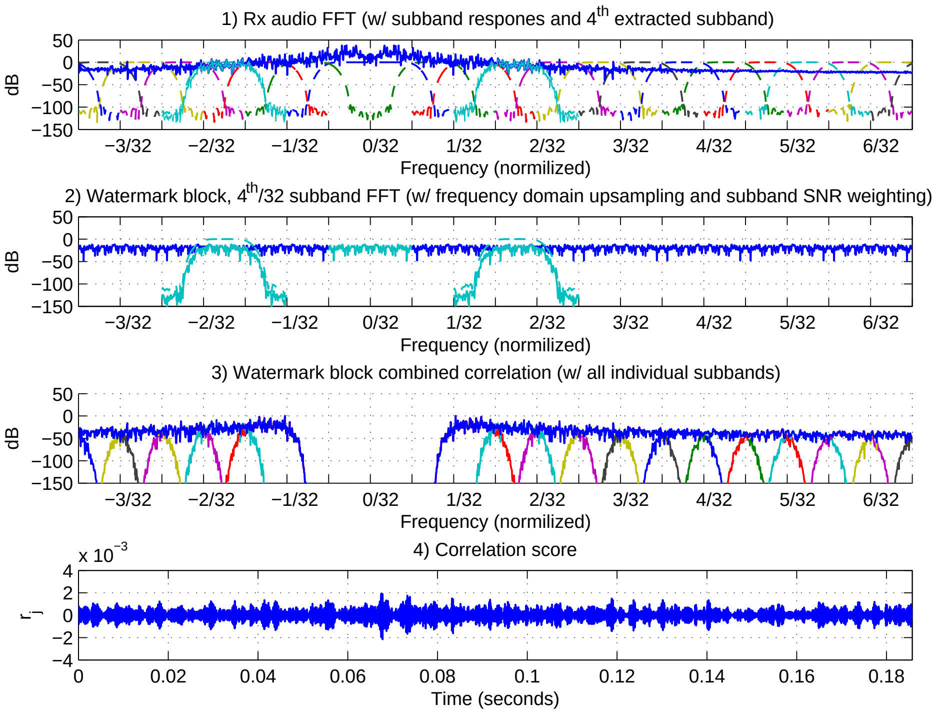


Figure 3. Correlations between watermark blocks, w_j and z are performed in the FFT domain. Plot 1 shows the 4th subband extracted from the FFT spectrum of z in (10). Plot 2 shows the frequency domain upsampling of watermark subbands in (11). Plots 3 and 4 show the correlation score calculation in (12). Cepstrum filtering is omitted.

The first operation shown in Fig. 3 extracts the band limited PQMF subband spectra,

$$Z_k[\nu] = \mathcal{F}_N\left\{\text{ceps}\left\{z[t]\right\}\right\} \cdot H_{\text{PQMF},k}[\nu]. \tag{10}$$

from z, according to frequency responses $H_{\text{PQMF},k}[\nu]$, of the K PQMF subbands. Generalizing to other embedding transforms is a matter of updating the frequency responses, $H_{\text{PQMF},k}[\nu]$. The FFT and multiplication with K responses of $\frac{N}{K}$ points each, is $O(N \log N)$.

In the second operation in Fig. 3, all subbands of $w_{j,k}[n]$ are transformed to the FFT domain before correlation. The baseband spectra produced by FFT of the watermark block subbands are shifted up to the respective frequencies of the PQMF subbands they were embedded in. Shifting also replicates the spectra to account for aliasing of embedded power outside nominal subband frequencies. Wiener detection requires weighting these shifted spectra by expected watermark power.[§] These operations produce spectra for each watermark block subband:

$$W_{j,k}[\nu] = \left(\mathcal{F}_{\frac{N}{K}}\left\{w_{j,k}[n]\right\} * \delta_{\frac{N}{K}}\left\{\nu + \tfrac{N}{K}(k+1) \pmod 2\right\}\right) \cdot \left|H_{\text{PQMF},k}[\nu]\right|. \tag{11}$$

The frequency inversion when $(k+1) \pmod 2 \neq 0$ is required by the PQMF. Complexity for all $\frac{N}{K}$ point FFTs in the K subbands and weighting (neglecting out-of-band power) is $O\left(N \log \frac{N}{K}\right)$ for each watermark block.

FFT domain correlations, shown third in Fig. 3, are calculated by multiplication:

$$r_j[t] = \mathcal{F}_N^{-1}\left\{\sum_{k=1}^{K} W_{j,k}^{*}[\nu] \cdot Z_k[\nu]\right\}. \tag{12}$$

[§] multiplying by $|H_{\text{PQMF},k}[\nu]|$ accounts for the factor $H_{\text{PQMF},k}[\nu]$ already present in $Z_k[\nu]$ to result in overall weighting by the PSD of each embedded watermark subband. This assumes the interference PSD is white after cepstrum filtering.

Summing the K band limited subband correlations in the FFT domain is equivalent to summing in the time domain, but saves computations both from the summation and IFFT. Overall complexity is dominated by the $O(N \log N)$ FFT and IFFT used to extract subband spectra and produce each time domain correlation, $r_j[t]$.

2.4 Dynamic Programming Detection

Figure 2 gives an overview of the detection process, showing how DP searches through correlations to find the maximum sequence of detections. Alignments resulting from these detection sequences are shown in Fig. 5 for both watermarked and unmarked z.

The first step after FFT domain correlations generate r_j, is forward backward reinforcement. Reinforcement builds detection peaks using local knowledge disregarded by DTW. Two passes move through the watermark blocks similar to how forward and backward probabilities are tabulated when training hidden Markov models. Maximum previous and subsequent sequences of correlation peaks are recursively calculated as:

$$r_{\text{forw}}[j,t] = r_j[t] + D\left(\max_{t_{\min} \le t' \le t_{\max}} r_{\text{forw}}[j-3, t-t']\right) \tag{13}$$

and,

$$r_{\text{back}}[j,t] = r_j[t] + D\left(\max_{t_{\min} \le t' \le t_{\max}} r_{\text{back}}[j+3, t+t']\right) \tag{14}$$

respectively. Each correlation score is reinforced according to these previous and subsequent block sequence correlations by combining r_{forw} and r_{back}, producing

$$r'_j[t] = r_j[t] + D\left(\max_{t_{\min} \le t' \le t_{\max}} r_{\text{forw}}[j-3, t-t'] + \max_{t_{\min} \le t' \le t_{\max}} r_{\text{back}}[j+3, t+t']\right). \tag{15}$$

The limits $t_{\min}$ and $t_{\max}$ bound the range of correlation spacings reinforced. This allows robustness to some tolerance in blocklength. The decay factor $D < 1$ limits the timescale that reinforcement works over. If reinforcement persists for too long, errors are likely in cases where insertions or deletions disrupt the assumed spacing of detections.

To find s_j in r'_j, detection uses DTW for robustness to insertions and deletions. DTW searches for optimal alignment between a test and template signal by maximizing a total similarity measure. In the proposed scheme, the template signal is the sequence of watermark blocks, w_j, and the test signal is the received audio, z. The similarity measure between each w_j and $z[t]$ is the reinforced correlation $r'_j[t]$. DTW's first pass through the watermark blocks constructs a matrix, Q, of maximum total correlation scores. Each $Q[j,t]$ holds the maximum total score resulting from optimally aligning watermark blocks 1 to j of the received audio signal up to time t. To allow dynamic programming, Q is filled recursively:

$$Q[j,t] = \max \begin{cases} Q[j, t-1] + r'_{\text{insert}}[t] & \text{(Insertion)} \\ Q[j-1, t] + r'_{\text{delete}}[t] & \text{(Deletion)} \\ Q[j-1, t-t_{\min,\text{DTW}}] + r'_j[t] & \text{(Detection).} \end{cases} \tag{16}$$

1. **Insertion** of 1 sample of un-watermarked audio.
2. **Deletion** of audio where watermark block j was embedded.
3. **Detection** of watermark block j at sample t in z.

The **detection** case of (16) makes DTW synchronization especially useful for SS watermarks: adding a current r'_j to the total correlation of previously aligned blocks $Q[j-1, t-t_{\min}]$, is equivalent to improving detection SNR through increasing SS chip sequence length without decreasing synchronization robustness. $t_{\min,\text{DTW}}$ is the min number of samples between detections of consecutive watermark block detections. Dummy terms r'_{insert} and r'_{delete} bias the detection process against declaring weak correlation scores as detections similar to how adjusting a MAP threshold can balance false positives and false negative errors.

After the first pass, the last element in Q will hold the maximum correlation score resulting from the optimal alignment of the entire sequence of watermark blocks. DTW then backtracks through the optimum alignment in Q starting from the last element to find the times in the received audio where each watermark block was detected or deleted. These times are the synchronization points, s_j, given to the cross validation stage.

The last step at the receiver, cross validation, addresses the high false positive rate typical of advanced search techniques such as DTW. Cross validation relies on the multiple watermarks embedded with the set theoretic framework. Because watermarks overlap in both time and frequency, relative positions of nearby watermark blocks are not disrupted by insertions or deletions. Validation shown in Fig. 4 requires at least 3 s_j in sequence must match expected spacing. Because forward backward reinforcement only favors expected separation within watermarks (every third s_j) it is unlikely that false positive are validated.

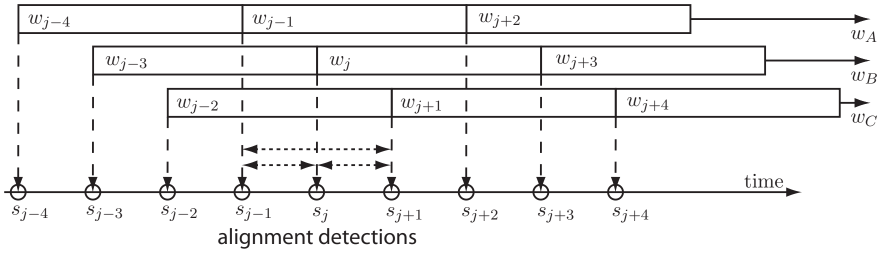


Figure 4. The overlapping watermark blocks (segments $w_{j-3,\ldots,j+4}$) allow cross validation of the DTW synchronization points (circles). To validate a sequence of detections the 3 separations shown (dotted lines) must match expectations.

3. RESULTS

To demonstrate set theoretic embedding and DP detection, watermarking trials use a small but diverse set of audio clips ripped from original CDs to 16 bit PCM (pulse code modulated) .wav files at a sampling rate of 44.1kHz. Trials explore performance tradeoffs over ranges of parameters instead of utilizing individually optimized values for each operating point. We compare relative performance results against a baseline "naive" SS scheme that directly uses the block correlations in Fig. 1 in a threshold detector for obtaining receiver operating characteristics.

For each clip, 30 independent trials are performed. Each trial embeds 3 simultaneous SS watermarks in the audio. Embedding uses a $K = 32$ band PQMF, but watermark chips in the lowest 2 subbands are set to 0 to avoid host interference. Embedding and detection also remove content above 16 kHz to avoid using frequency bands commonly discarded by compression. Set theoretic embedding constraints use distortion thresholds calculated by Ref. 13 for perceptibility constraints, and estimate MP3 distortion using quantization step sizes taken from an MP3 of the original clip produced using *LAME version 3.98.4*[14] with bitrate 128 kbps and the *-h* high quality flag set. Adaptive cepstrum filtering at the receiver and its estimate at the transmitter are applied in the MP3 MDCT domain which gives short time spectra with 576 frequency lines using all *long blocks*. As in Ref. 5, cepstrums are taken by DCT of the short time spectra dB magnitude. The DCT uses a boxcar window that ignores frequencies above 16 kHz, again to avoid compression cutoff frequencies. Removing the lowest 3 cepstrum coefficients is found to balance whitening host interference with preserving the watermark signals for detection. Embedding blocks, w_i use $B = 240$ chips per block within each subband, and target SNR = 20 dB for robustness[¶]. POCS was limited to 200 iterations with the perceptibility projection last to give that constraint precedence in situations where convergence was not achieved. While watermarked audio sounded transparent to our untrained ears, adherence to the perceptual model[13] provides a more reproducible measure of audio quality. Parameters for insertion deletion channel are set so lengths of preserved, inserted, and deleted segments are exponentially distributed with expected lengths of 100,000 samples, 20,000 samples, and 20,000 samples, respectively. After each preserved segment there is a $\frac{1}{3}$ chance of deleting a segment of watermarked audio, $\frac{1}{3}$ chance of inserting a segment of unwatermarked audio (randomly taken from the original clip), and $\frac{1}{3}$

[¶] To improve embedding performance in the absence of convergence a slack term was added heuristically the robustness projections, increasing τ by a factor of 1.5 to project y_k into the interior of the set.

chance of both. Compression in the channel also uses the *LAME* MP3 encoder[14] with *-h* flag and bitrate of 128 kbps. Detection for all 30 trials was run 18 times, testing 6 detection blocklengths at 3 different warping tolerances for each watermarked audio clip. Robustness to warping was not tested because correlations were found to only withstand delay, not time/frequency warping. Figure 7 shows average block error rates over the 30 trials. Detection typically ran 10X slower than real time[‖], but less naive DP implementations and removing overhead due to MATLAB implementation hold significant speedup potential.

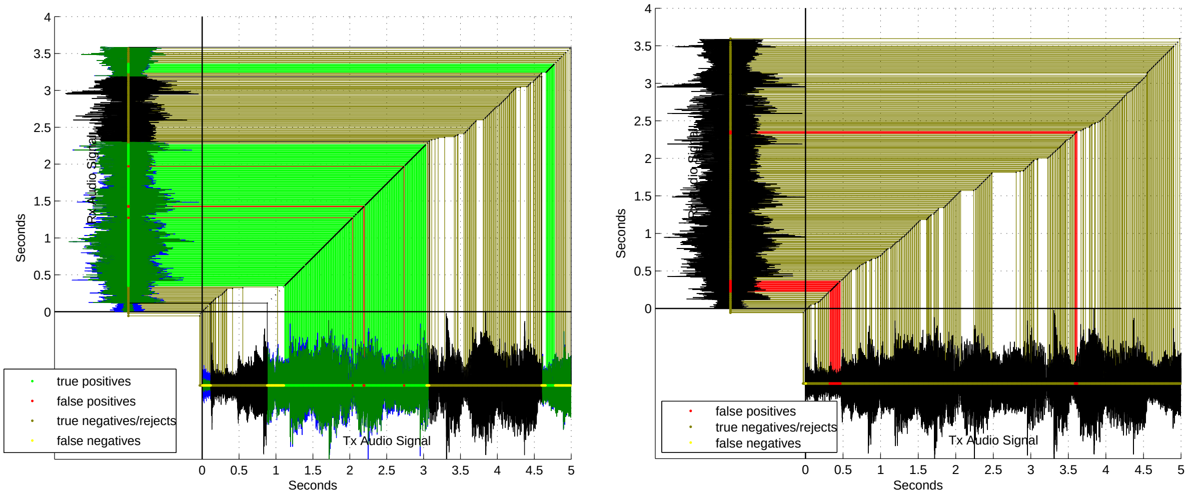
Results in Fig. 7 show the proposed DP detection scheme presents a significant improvement over typical SS detection in the presence of insertions and deletions. To insure the reference naive SS scheme gives a fair comparison many steps are shared from the proposed scheme, such that any improvement seen can be attributed to the cross validated DP methods. The naive SS implementation embeds a single sequence of SS chips and detects watermark blocks directly from correlation scores: no DP or cross validation are used after FFT domain correlations. To account for the cost of cross validating multiple watermarks the single naive SS watermark is embedded with the same overall PQMF subband distortion allowance used for the multiple watermarks:

$$y_{\text{naive},k} = x_k + w_{A,k}\beta_k. \tag{17}$$

After w_A is embedded in y_{naive}, insertions, deletions, and MP3 compression follow identical procedures as the DP scheme. The naive scheme receiver uses identical cepstrum filtering and FFT domain correlation procedures as the DP scheme. Fixing the detection block length between the two schemes provides equivalent watermark localization and robustness. However, once correlation scores are generated, the naive SS detector simply records the distributions of correlation peaks where watermark blocks are embedded and the false positive correlations elsewhere. Using these distributions, given in Fig. 6, thresholds can be applied to produce the naive ROC curves in Fig. 7.

As in Fig. 7, the tradeoff between robustness and accuracy is governed by DP detection through the choice of block length and a warping tolerance parameters. Block length B, and warping tolerance γ, determine the

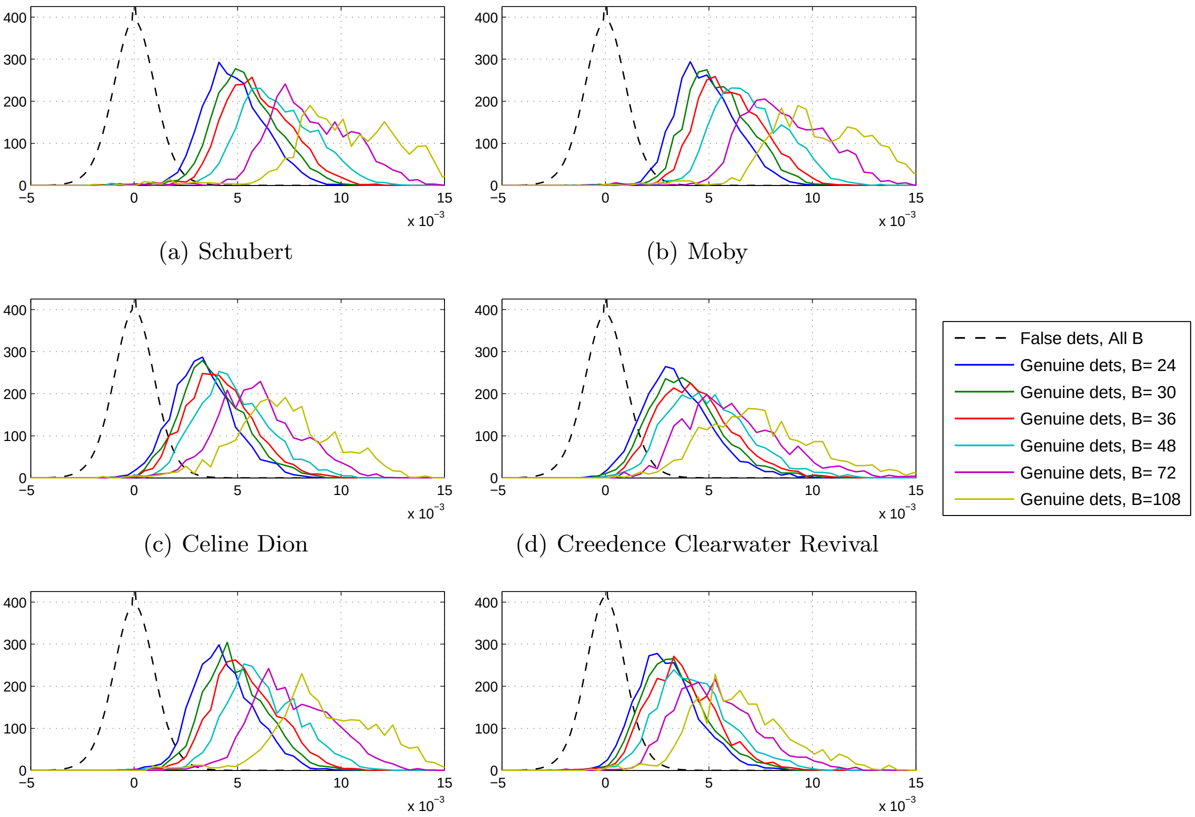
[‖] Benchmark run on 2.13GHz Core2Duo desktop machine without parallelization.



(a) Watermarked

(b) Unwatermarked

Figure 5. Receiver aligns s_j in the Rx signal (y-axis), to w_j from the Tx signal (x-axis). Waveforms shown for reference only. **Detection is blind**. Plots 5(a), 5(b) show detection response to watermarked/unmarked audio. Insertion/deletions are distinguished (Black). Detections are either correctly aligned (green true positives), lost by deletion or rejected during validation (brown true negatives), or misaligned and incorrectly validated (red false positives). Missed block detections are shown in the Tx waveform (yellow false negatives). Detection shown for *Jewel* clip using blocklenth 36 and warping tolerance 1%.



(a) Schubert

(b) Moby

(c) Celine Dion

(d) Creedence Clearwater Revival

(f) Creedence Clearwater Revival

(g) Jewel

Figure 6. Naive SS correlation score histograms colored by watermark detection block length. Naive watermarks are embedded using the same overall perceptual thresholds from the set theoretic embedding framework, but only a single SS watermark sequence is embedded. After correlation calculations (exhaustive search of PQMF offsets), Naive detection does not use forward-backward reinforcement, DTW detection or cross validation.

forward backward reinforcement limits and minimum DTW separation:

$$t_{\min} = KB(1-\gamma), \tag{18}$$

$$t_{\max} = KB(1+\gamma), \tag{19}$$

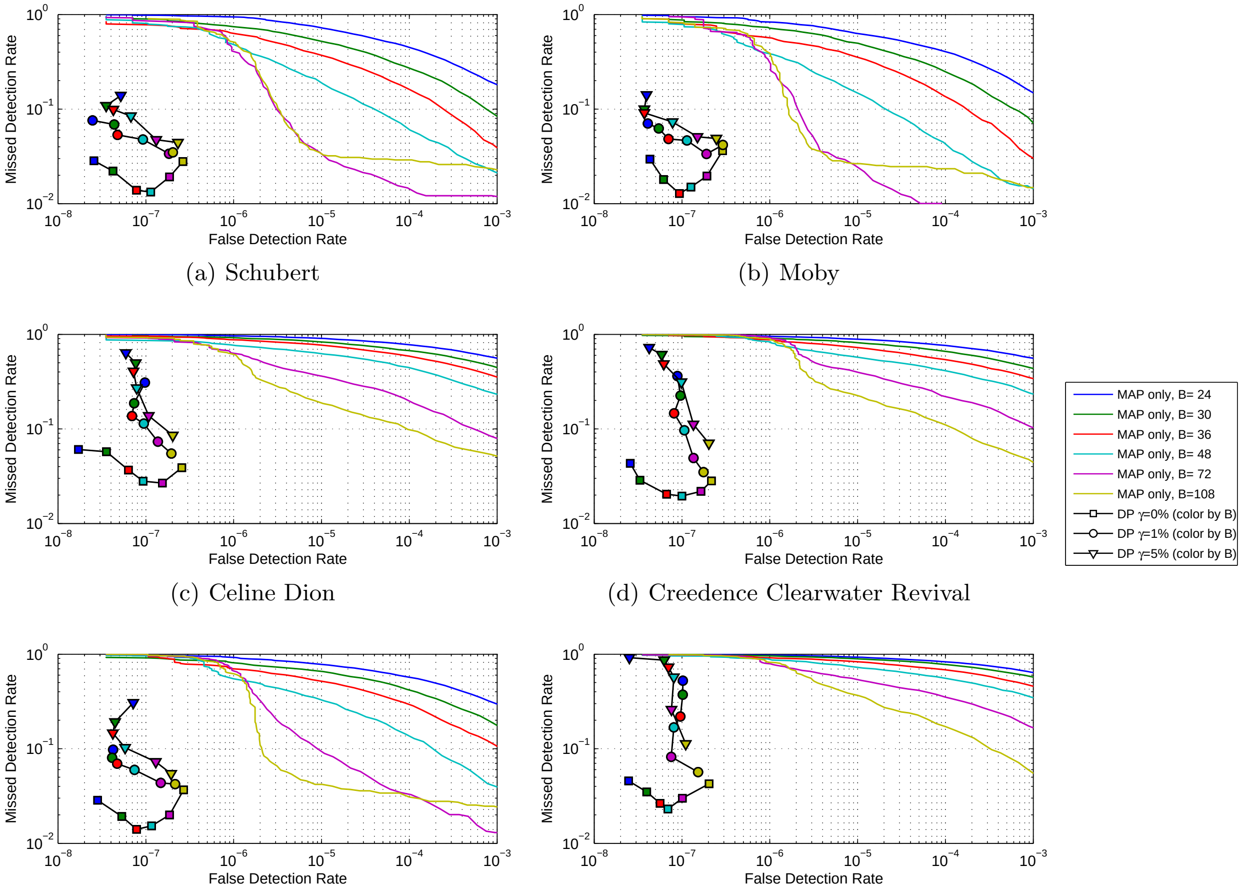
$$t_{\min,\mathrm{DTW}} = K\frac{B}{3}(1-\gamma). \tag{20}$$

The factor K accounts for upsampling due to exhaustive search of PQMF transform offsets; the factor of 3 accounts for the overlap between blocks in DTW detection. As mentioned, decreasing SS blocklength increases robustness, but at the cost of increasing error rates. This trend can be seen in Fig. 7 as the naive SS implementations use shorter watermark blocks and performance suffers. However, as block length varies for DP detection there is generally a tradeoff between false positive and negative errors rather than a strict loss of performance. This contradiction between how blocklength effects the naive and DP schemes results from the DP scheme's internal tradeoff between detection and validation. This trade-off depends on the tendency of DP detection to return long regularly spaced sequences of detections that will be validated whether or not they are genuine detections and is governed by the remaining parameters: forward backward decay D, and DTW dummy terms r'_{insert} and r'_{delete}. The following definitions were empirically determined to work well for a blocklength of 36 and used for all detection trials:

$$D = .8, \tag{21}$$

$$r'_{\text{insert}}[t] = \tfrac{1}{t_{\min,\text{DTW}}}\sigma_{r'}[t], \qquad r'_{\text{delete}}[t] = \tfrac{1}{4}\sigma_{r'}[t], \tag{22}$$

$$\sigma_{r'}[t] = \sqrt{\frac{1}{J}\sum_{j=1}^{J} r'[j,t]^2}.$$



(a) Schubert

(b) Moby

(c) Celine Dion

(d) Creedence Clearwater Revival

(f) Creedence Clearwater Revival

(g) Jewel

Figure 7. Block error rate in the presence of insertion and deletions shows DP detection significantly improves naive SS detection. The 6 audio clips show varying watermark detection performance. Error rates are averaged over 30 trials at each point. Missed detection rate considers all blocks of which two thirds or less of the block duration has been deleted.

4. CONCLUSION

This paper presented an audio watermarking scheme robust to desynchronization (insertion/deletion) and non-desynchronizing distortion (MP3 compression) using a powerful yet efficient DP search as part of blind detection. Two typical challenges of advanced searches for watermark detection: feasibility and false positive detections, are addressed respectively by DTW methods and cross validation of multiple watermarks. Insertion and deletion robustness is an important problem because these operations can desynchronize watermark detection, a class of attacks many watermarking schemes struggle with. The proposed scheme is motivated by considering watermark localization and its impact on the tradeoff between desynchronization and non-desynchronization robustness. For example, SS techniques provide the ability to reduce non-desynchronizing interference in proportion to blocklength. However, shorter blocks with greater localization, are less likely to be split apart by insertions or deletions, and more robust to desynchronization. DP detection is shown to govern this robustness tradeoff. A second dimension, the tradeoff between false positive/negative error rate is controlled by the proposed cross validation and reinforcement techniques. These detection methods balance characteristics of embedded watermarks between those used to search for optimal detections, preventing missed detections, and those reserved for validation to prevent false positives.

REFERENCES

[1] Kirovski, D. and Attias, H., "Audio watermark robustness to desynchronization via beat detection," in [*Information Hiding*], Petitcolas, F., ed., *Lecture Notes in Computer Science* **2578**, 160–176, Springer Berlin / Heidelberg (2003).

[2] Coumou, D. J. and Sharma, G., "Insertion, deletion codes with feature-based embedding: A new paradigm for watermark synchronization with applications to speech watermarking," *IEEE Transactions on Information Forensics and Security* **3**, 153–165 (June 2008).

[3] Barni, M., "Effectiveness of exhaustive search and template matching against watermark desynchronization," *IEEE Signal Processing Letters* **12**, 158–161 (Feb. 2005).

[4] Merhav, N., "An information-theoretic view of watermark embedding-detection and geometric attacks," in [*WaCha*], (June 2005). Barcelona, Spain.

[5] Kirovski, D. and Malvar, H. S., "Spread-spectrum watermarking of audio signals," *Signal Processing* **51**, 1020–1033 (Apr. 2003).

[6] Wu, S., Huang, J., Huang, D., and Shi, Y., "Efficiently self-synchronized audio watermarking for assured audio data transmission," *Broadcasting, IEEE Transactions on* **51**, 69 – 76 (Mar. 2005).

[7] Altun, O., Sharma, G., Celik, M., and Bocko, M., "A set theoretic framework for watermarking and its application to semifragile tamper detection," *IEEE Transactions on Information Forensics and Security* **1**, 479–492 (Dec. 2006).

[8] Painter, T. and Spanias, A., "Perceptual coding of digital audio," *Proceedings of the IEEE* **88**, 451–513 (Apr. 2000).

[9] Xu, C., Lim, Y., and Feng, D. D., "Recovering modified watermarked audiobased on dynamic time-warping technique," in [*Digital Image Computing Techniques and Applications*], (2002).

[10] Kim, T.-H., Lee, J., and Shin, S. Y., "Robust motion watermarking based on multiresolution analysis," *Computer Graphics Forum* **19**(3), 189–198 (2000).

[11] Foote, J., Adcock, J., and Girgensohn, A., "Time base modulation: a new approach to watermarking audio," in [*Multimedia and Expo, 2003. ICME '03. Proceedings. 2003 International Conference on*], **1**, I – 221–4 vol.1 (July 2003).

[12] Pan, D., "A tutorial on MPEG/audio compression," *IEEE Multimedia* **2**, 60–74 (1995).

[13] Petitcolas, F. A. P., "MPEG psychoacoustic model 1 for MATLAB," (Aug. 2003).

[14] "LAME MP3 encoder," (2012).