

# Statistical Model and Error Analysis of a Proposed Audio Fingerprinting Algorithm

E.P. McCarthy, F. Balado, G.C.M. Silvestre, N.J. Hurley  
School of Computer Science and Informatics,  
University College Dublin, Ireland.

## ABSTRACT

In this paper we present a statistical analysis of a particular audio fingerprinting method proposed by Haitsma et al.<sup>1</sup> Due to the excellent robustness and synchronisation properties of this particular fingerprinting method, we would like to examine its performance for varying values of the parameters involved in the computation and ascertain its capabilities. For this reason, we pursue a statistical model of the fingerprint (also known as a hash, message digest or label). Initially we follow the work of a previous attempt made by Doets and Lagendijk<sup>2-4</sup> to obtain such a statistical model. By reformulating the representation of the fingerprint as a quadratic form, we present a model in which the parameters derived by Doets and Lagendijk may be obtained more easily. Furthermore, our model allows further insight into certain aspects of the behaviour of the fingerprinting algorithm not previously examined. Using our model, we then analyse the probability of error ( $P_e$ ) of the hash. We identify two particular error scenarios and obtain an expression for the probability of error in each case. We present three methods of varying accuracy to approximate  $P_e$  following Gaussian noise addition to the signal of interest. We then analyse the probability of error following desynchronisation of the signal at the input of the hashing system and provide an approximation to  $P_e$  for different parameters of the algorithm under varying degrees of desynchronisation.

**Keywords:** Audio fingerprint, statistical model, quadratic form, desynchronisation, error analysis.

## 1. INTRODUCTION

An audio fingerprint is a compact representation computed from the perceptually most relevant components of an audio recording.<sup>2</sup> As a result, fingerprint similarity should be representative of perceptual similarity between audio recordings, and less dependent on their similarity at bit-level. Any perceptually inaudible distortion of an original signal should not result in an altered fingerprint. The most important issues to be considered in fingerprint generation are the size, robustness and synchronisation properties of the fingerprint. The fingerprint should be as small as possible without loss of discriminatory power and robust to imperceptible distortion and desynchronisation of the original audio. Fingerprinting may be used in applications such as indexing of multimedia databases and authentication, where the fingerprint (also known as a hash, message digest or label) provides a compact representation which can be used to identify the data efficiently.

One particular method of audio fingerprinting which has proven to be remarkably robust is that proposed by Haitsma et al.<sup>1</sup> In this algorithm an audio signal is divided into overlapping frames and a *subfingerprint* computed for each frame. Obtaining these subfingerprints, as opposed to a complete hash for the whole audio signal, enables sections of the recording to be identified. It effectively deals with the issue of desynchronisation of the signal input to the hashing system with respect to the original. In the method, after performing the short term Fourier Transform on each frame, each resulting short term spectrum is divided into non-overlapping frequency bands and the energy of each band computed. A subfingerprint is then obtained by applying a simple slicer to the values of energy differences in both time and frequency. The final hash is a collection of those subfingerprints that correspond to the interval of the signal to be hashed. Due to the excellent robustness and synchronisation properties of this particular hashing method, we would like to examine its performance for varying values of the parameters involved in the computation and ascertain its capabilities. By optimising such parameters as the amount of overlap between frames, the window applied to each frame and the frame length, we aim to minimise the probability of error of the hash. To this end, we pursue a statistical model.

Previously, attempts have been made to obtain such a statistical model. A simple performance model was presented with the algorithm by Haitsma *et al.*,<sup>1</sup> under the assumption that the hash bits are independent and

identically distributed (i.i.d.). Given the overlap involved in the computation of the hash values, it is clear that this is not a realistic assumption. A more elaborate model of the same algorithm is proposed by Doets and Lagendijk,<sup>2-4</sup> for the case in which the signal to be hashed is uncorrelated Gaussian noise. The outcome of this model is the probability of fingerprint bit value conditioned to the fingerprint bit value of the previous frame corresponding to the same energy band. Before discussing further the model of Doets and Lagendijk, it should be noted that the algorithm of Haitsma et al<sup>1</sup> closely resembles the Welch method of spectrum estimation. In Welch’s method an audio signal is segmented into possibly overlapping frames, and the spectral estimate obtained by averaging a modified periodogram of each frame. The modified periodogram of one of these overlapping frames is obtained by applying the Fourier Transform to the windowed frame. The similarity between Welch’s method and the algorithm of Haitsma et al<sup>1</sup> is clearly taken into consideration in Doets and Lagendijk’s papers.<sup>2-4</sup> In these works, after obtaining the periodogram estimate of the power spectral density (p.s.d.) of a frame, the authors compute the energy difference between two successive frames per spectral sample. This is the initial step taken in obtaining an equivalent rearrangement of the hashing algorithm, which enables the authors to pursue a statistical model.

Here we will present an integrated model comprising all the components of the algorithm. Differently to,<sup>4</sup> where a continuous input signal is considered, we assume that the input  $\mathbf{x}$  to the hashing algorithm is already sampled. This approach is sufficient in principle if the sampling rate is high enough. Initially, we follow the work of Doets and Lagendijk in their convenient rearrangement of the hashing algorithm. We then follow a similar approach to that proposed in<sup>5</sup> for the modeling of overlapped periodograms, in which it is shown that the periodogram estimate can be rewritten using a quadratic form. We show below that this enables the parameters of the jointly Gaussian model of<sup>4</sup> to be obtained more easily. Also, the computation of the parameters required by the model of<sup>4</sup> turns out to be cumbersome when a window different from the rectangular one is used. Recall that the method in<sup>1</sup> employs a von Hann window. The model we propose can be easily applied to any kind of window.

Using our model, we analyse the probability of error ( $P_e$ ) of the hash. We identify two particular error scenarios and obtain an expression for the probability of error in each case. We present three methods of varying accuracy to approximate  $P_e$  following Gaussian noise addition to the signal of interest. We then analyse the probability of error following desynchronisation of the signal at the input of the hashing system and provide an approximation to  $P_e$  for different parameters of the algorithm under varying degrees of desynchronisation.

Initially, we will obtain the statistical model of an arbitrary band coefficient before the slicer with threshold at zero (i.e.  $ED(n, m)$  in Figure 1). Notice that, differently to,<sup>4</sup> we pursue a probabilistic model of the “soft values” before quantization, which allows for a more accurate analysis.

## 2. STATISTICAL MODEL OF ONE COEFFICIENT BEFORE QUANTIZATION

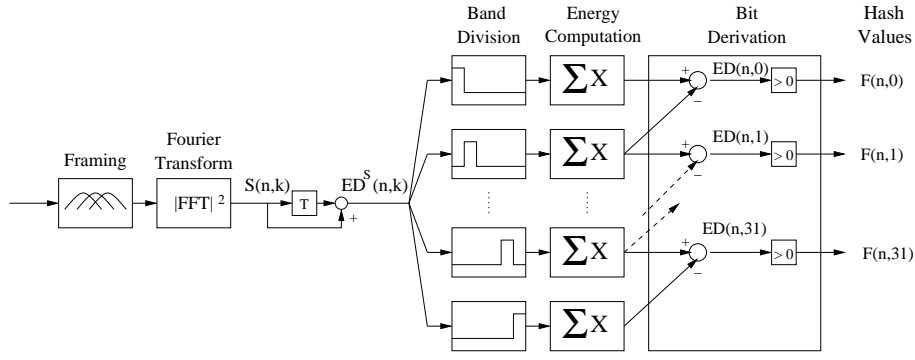
Firstly we introduce some notation, following that of<sup>4</sup> as much as possible. Lower case bold face will be used to indicate that a letter represents a vector, while matrices are represented by upper case letters.  $\mathbf{x}$  denotes the framed signal to be hashed,  $L$  the number of samples in a frame and  $\Delta L$  the number of non-overlapping samples between frames. The arbitrary window used in the computation will be denoted by the  $L$ -length vector  $\mathbf{w}$ . The  $L$ -length column vector formed by the elements of  $\mathbf{x}$  used in the computations corresponding to the  $n$ -th frame is given by

$$\mathbf{x}_n \triangleq (x[n \cdot \Delta L], x[n \cdot \Delta L + 1], \dots, x[n \cdot \Delta L + L - 1])^T. \quad (1)$$

The random variable (r.v.) we wish to model, i.e. a single soft hash value, is denoted by  $ED(n, m)$  and is computed as follows:

$$ED(n, m) = [E(n, m) - E(n, m + 1)] - [E(n - 1, m) - E(n - 1, m + 1)], \quad (2)$$

where  $E(n, m)$  denotes the energy of band  $m$  of frame  $n$ . As previously mentioned, we follow here the convenient equivalent rearrangement of the Phillips algorithm described in<sup>4</sup> and illustrated in Figure 1.



**Figure 1.** Hashing algorithm for one frame, rearranged as in.<sup>4</sup>

The periodogram estimator of the power spectrum of the windowed signal at the  $n$ -th frame is given by

$$S(n, k) = \frac{1}{LU} \left| \sum_{i=0}^{L-1} x_n[i] w[i] e^{-j2\pi i \frac{k}{L}} \right|^2, \quad (3)$$

with  $U = \frac{1}{L} \sum_{i=0}^{L-1} w^2[i]$  an energy normalization factor due to the windowing. For a rectangular window  $U = 1$ , while a von Hann window yields  $U = \frac{3}{8}$ . Now following a similar approach to that of,<sup>5</sup> the expression (3) can be rewritten in matricial form as

$$S(n, k) = \mathbf{x}_n^T \mathbf{M}_k \mathbf{x}_n, \quad (4)$$

where the  $L \times L$  matrix  $\mathbf{M}_k$  is defined as

$$\mathbf{M}_k \triangleq \frac{1}{LU} \Omega \mathbf{N}_k \Omega, \quad (5)$$

with  $(\mathbf{N}_k)_{n,m} \triangleq \cos(2\pi(n-m)k/L)$ , and with  $\mathbf{w}$  conveniently rearranged in the  $L \times L$  diagonal matrix  $\Omega \triangleq \text{diag}(\mathbf{w})$ . The imaginary parts of the exponentials are not included in the definition of  $\mathbf{N}_k$  as they cancel out anyway in the positive semidefinite quadratic form (4). Notice that  $\mathbf{M}_k$  is symmetric.

With (4) in hand, we will see that it is straightforward to obtain a quite compact expression for  $\text{ED}(n, m)$ . Firstly, the energy difference between two successive frames per spectral sample is given by

$$\text{ED}^s(n, k) = S(n, k) - S(n-1, k) \quad (6)$$

Then by summing over the samples in a particular frequency band we may obtain the energy difference between two successive frames for that band:

$$\sum_{k \in B_m} \text{ED}^s(n, k) = E(n, m) - E(n-1, m) \quad (7)$$

where  $B_m$  is the set of integers indexing the samples in frequency band  $m$ . Finally we obtain

$$\begin{aligned} \text{ED}(n, m) &\triangleq \sum_{k \in B_m} \text{ED}^s(n, k) - \sum_{k \in B_{m+1}} \text{ED}^s(n, k) \\ &= \sum_{v=n-1}^n \sum_{w=m}^{m+1} \sum_{k \in B_w} (-1)^{n-v+m-w} S(v, k). \end{aligned} \quad (8)$$

Using (4), we can divide (8) in two summands as

$$\text{ED}(n, m) = \mathbf{x}_n^T \mathbf{P}^m \mathbf{x}_n - \mathbf{x}_{n-1}^T \mathbf{P}^m \mathbf{x}_{n-1}, \quad (9)$$

with

$$\mathbf{P}^m \triangleq \sum_{d=m}^{m+1} \sum_{k \in B_d} (-1)^{m-d} \mathbf{M}_k. \quad (10)$$

We may further simplify (9) recalling from (1) that  $\mathbf{x}_n$  and  $\mathbf{x}_{n-1}$  share  $L - \Delta L$  components. To that end, we define the extended vector

$$\mathbf{y}_n \triangleq (x[(n-1) \cdot \Delta L], \dots, x[n \cdot \Delta L + L - 1])^T, \quad (11)$$

with length  $M \triangleq L + \Delta L$ , and the  $M \times M$  matrix

$$\mathbf{Q}^m \triangleq \begin{bmatrix} -\mathbf{P}^m & & & \\ & \vdots & & \\ & & \ddots & \\ & & & \mathbf{P}^m \end{bmatrix}, \quad (12)$$

which is the matrix formed by adding  $-\mathbf{P}^m$  at the position  $(1, 1)$  with  $\mathbf{P}^m$  at the position  $(\Delta L + 1, \Delta L + 1)$ . Using definitions (11) and (12), we can finally write (9) as

$$\text{ED}(n, m) = \mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n. \quad (13)$$

If  $\mathbf{y}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ , this is a quadratic form in a Gaussian vector. This r.v. may be expressed as a weighted sum of  $\chi^2$  distributions,<sup>6</sup> whose moment generating function (MGF) is

$$\Phi(t) = E\{\exp(-tx)\} = \prod_{i=1}^D (1 - t2\lambda_i)^{\frac{\nu_i}{2}}, \quad (14)$$

with  $\lambda_i$  the  $D$  distinct eigenvalues of  $\mathbf{R} \cdot \mathbf{Q}^m$  with multiplicities  $\nu_i$ .

We show next how the expressions derived above prove extremely useful in the derivation of certain parameters of the jointly Gaussian model.

### 3. DEALING WITH DEPENDENCIES

#### 3.1. Intraframe behaviour

In<sup>4</sup> the intraframe dependencies are not considered, i.e., given a frame  $n$ , the possible dependencies among the r.v.'s  $\text{ED}(n, m)$  corresponding to the bands  $m = 0, 1, \dots, 31$ . The relationship between these r.v.'s needs careful consideration, for, as it can be observed in (13), the random vector which generates all these random variables is exactly the same.

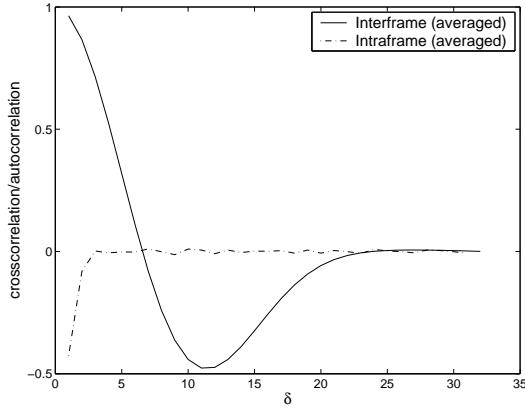
Rigorously, any two of the quadratic forms in the same Gaussian vector (13) are independently distributed iff<sup>7</sup>

$$\mathbf{Q}^m \cdot \mathbf{R} \cdot \mathbf{Q}^{m'} = \mathcal{O}, \quad (15)$$

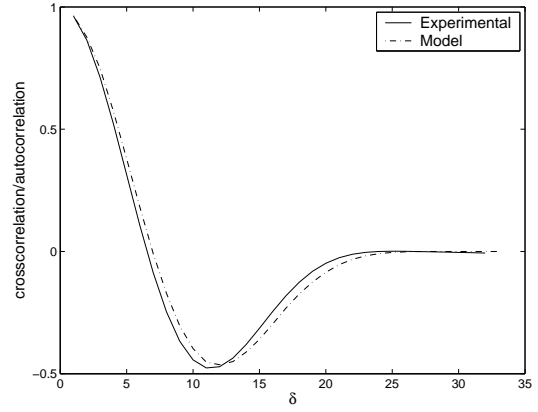
for all  $m \neq m'$ , and with  $\mathcal{O}$  the null matrix. We may readily check (15), and assume independence when the matrix obtained is sufficiently "close" to the null one. To this end, a matricial norm could be used. In practice, it is convenient to check if the random variables are uncorrelated. For zero-mean Gaussian  $\mathbf{y}_n$  the correlation (covariance) between the quadratic forms  $\text{ED}(n, m)$  and  $\text{ED}(n, m')$  is given by (see for instance<sup>7</sup>)

$$\text{Cov}\{\text{ED}(n, m), \text{ED}(n, m')\} = 2 \text{tr}(\mathbf{R} \cdot \mathbf{Q}^m \cdot \mathbf{R} \cdot \mathbf{Q}^{m'}), \quad (16)$$

where  $\text{tr}(\cdot)$  is the trace of a matrix. Notice from replacing (15) in (16) that independence implies in this case uncorrelation, although the reverse may not be true. We will assume approximate intraframe independence if (16) is small.



**Figure 2.** Intraframe dependency: the correlation between  $ED(n, m)$  and  $ED(n, m - \delta)$ , averaged over  $n$ ; and interframe dependency: the correlation between  $ED(n, m)$  and  $ED(n - \delta, m)$ , averaged over  $m$ , under i.i.d. Gaussian input.



**Figure 3.** Interframe dependence using a von Hann window under i.i.d. Gaussian input.

### 3.2. Interframe behaviour

We will assume from now on intraframe independence and we will verify that this is reasonable. As regards the interframe dependencies, notice that the vectors  $\mathbf{y}_n$  overlap for several consecutive frame indices  $n$  depending on  $\Delta L$ . Then, for a fixed band  $m$  the consecutive r.v.'s  $ED(n, m)$  are strongly dependent as the corresponding quadratic forms share the associated matrix  $\mathbf{Q}^m$ . Actually, this dependence is an inherent side effect of the good synchronisation properties of the algorithm.

#### 3.2.1. Rederivation of parameters for Gaussian model

As mentioned previously, our expression for  $ED(n, m)$  as a quadratic form (13), enables us to easily compute the parameters of the model proposed in,<sup>2</sup> where two consecutive interframe values of  $ED(n, m)$  are roughly modeled as jointly Gaussian on the basis of empirical observations. Although we have seen that the exact model is given by (14), the Gaussian assumption is reasonably accurate and easier to handle analytically. The performance of the system may be then analysed by obtaining the p.d.f. of  $ED(n, m)$  conditioned to the previous value of  $ED(n - 1, m)$ , which constitutes a Markov chain.

In this case the model is completely specified by the covariance of  $ED(n, m)$  and  $ED(n - 1, m)$ , and by the corresponding expectations and variances. These quantities are obtained in<sup>4</sup> through a cumbersome process. They may be obtained in a simpler way with the model that we have proposed, which moreover can be easily applied to any kind of window.

First, we must obtain the expectation and variance of the variables (13), which are computed as

$$\begin{aligned} E\{ED(n - 1, m)\} &= E\{ED(n, m)\} \\ &= \text{tr}(\mathbf{R} \cdot \mathbf{Q}^m), \end{aligned} \quad (17)$$

and

$$\begin{aligned} \text{Var}\{ED(n - 1, m)\} &= \text{Var}\{ED(n, m)\} \\ &= 2 \text{tr}((\mathbf{R} \cdot \mathbf{Q}^m)^2). \end{aligned} \quad (18)$$

Notice that the input Gaussian process is required to be stationary for (17) and (18) to hold. Also, in<sup>2</sup> the expectation of the joint Gaussian is not computed, although it is correctly assumed to be zero.

For obtaining the covariance we will need some convenient definitions similar to those already made for writing  $\text{ED}(n, m)$  as a quadratic form. Now we wish to rewrite  $\text{ED}(n, m)$  and  $\text{ED}(n-1, m)$  as quadratic forms in the same Gaussian vector. To this end, let us define

$$\tilde{\mathbf{y}}_n \triangleq (x[(n-2) \cdot \Delta L], \dots, x[n \cdot \Delta L + L - 1])^T, \quad (19)$$

with length  $\tilde{M} \triangleq M + \Delta L$ , and the two  $\tilde{M} \times \tilde{M}$  matrices

$$\tilde{\mathbf{Q}}^{n-1, m} \triangleq \begin{bmatrix} \mathbf{Q}^m & \vdots \\ \vdots & \mathbf{Q}^m \end{bmatrix}, \quad \tilde{\mathbf{Q}}^{n, m} \triangleq \begin{bmatrix} \vdots & \mathbf{Q}^m \\ \mathbf{Q}^m & \vdots \end{bmatrix}, \quad (20)$$

obtained by placing  $\mathbf{Q}^m$  at  $(1, 1)$  and  $(\Delta L + 1, \Delta L + 1)$  respectively. Clearly, with these definitions we can write  $\text{ED}(i, m) = \tilde{\mathbf{y}}_n^T \tilde{\mathbf{Q}}^{i, m} \tilde{\mathbf{y}}_n$  for  $i \in \{n-1, n\}$ . Then, the covariance sought is just (cf. (16))

$$C_{n, m} \triangleq \text{Cov}\{\text{ED}(n, m), \text{ED}(n-1, m)\} = 2 \text{tr}(\tilde{\mathbf{R}} \cdot \tilde{\mathbf{Q}}^{n, m} \cdot \tilde{\mathbf{R}} \cdot \tilde{\mathbf{Q}}^{n-1, m}), \quad (21)$$

with  $\tilde{\mathbf{R}}$  the covariance matrix of the Gaussian vector (19). Notice that the computation of (17), (18) and (21) is simple, only requiring the construction of the matrices involved. Fig. 2 illustrates the dependence of a hash value on previous frames, computed empirically. The solid line illustrates the correlation, for a fixed  $m$ , between the hash value for frame  $n$  and previous frames  $n - \delta$ ,  $\delta = 1, \dots, 31$ , averaged over  $m$ . The dashed line illustrates the correlation, for a fixed band  $m$ , between the hash value for band  $m$  and previous bands  $m - \delta$ ,  $\delta = 1, \dots, 31$ , averaged over  $n$ . Fig. 3 illustrates the consistency of the model with the empirical behaviour of the hashing algorithm.

#### 4. PROBABILITY OF ERROR

As discussed in the introduction, the signal presented to the algorithm may differ from the corresponding one indexed in the database. In this section we use the model obtained in the previous section, to examine the probability of error ( $P_e$ ) of the hash of a distorted input signal. We pursue an expression which may enable optimisation of certain parameters of the hashing algorithm.

We identify two general situations which result in a probability of error: a) the signal itself may be distorted, intentionally or not, by noise addition or operations such as compression; b) error may be introduced by desynchronisation of the signal, i.e. the frame boundaries of a section of the signal presented to the algorithm may not match exactly those for which the corresponding stored fingerprints are computed. The probability of error which results from the addition of Gaussian distributed noise is examined in Sect. 4.1, while that which results from desynchronisation of the signal is examined in Sect. 4.2.

##### 4.1. Noise Addition

In this section we examine the probability of error of a single hash value,  $\text{ED}(n, m)$ , following the addition of zero mean Gaussian distributed noise  $\mathbf{g} = \mathcal{N}(0, \sigma_g^2 I)$  to the corresponding extended vector  $\mathbf{y}_n$ . In the case where no noise is added, the hash value is given by (13). Following noise addition, the hash value, denoted  $\text{ED}'(n, m)$ , is given by

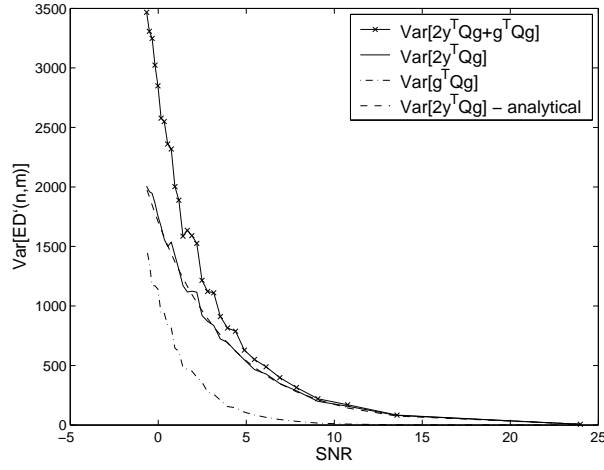
$$\begin{aligned} \text{ED}'(n, m) &= (\mathbf{y}_n + \mathbf{g})^T \mathbf{Q}^m (\mathbf{y}_n + \mathbf{g}) \\ &= \mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n + 2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g} + \mathbf{g}^T \mathbf{Q}^m \mathbf{g}. \end{aligned} \quad (22)$$

In our analysis of the probability of error we require an expression for the mean and variance of  $\text{ED}'(n, m)$ . The mean of the hash value is unaffected by the addition of zero mean noise as

$$\text{E}[\text{ED}'(n, m)] = \text{E}[\mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n + 2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g} + \mathbf{g}^T \mathbf{Q}^m \mathbf{g}] = \text{E}[\mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n] = \text{E}[\text{ED}(n, m)] \quad (23)$$

which may be computed easily from (17). The variance however is affected and may not be computed as in (18). In the case of low distortion, which is of interest to us, where the signal to noise ratio (SNR) is high, we may drop the term  $\mathbf{g}^T \mathbf{Q}^m \mathbf{g}$  yielding the following approximation to  $\text{ED}'(n, m)$

$$\text{ED}'(n, m) \approx \mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n + 2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g}. \quad (24)$$



**Figure 4.** Variance of the noisy hash value  $\text{ED}'(n, m)$  given by analytical and empirical expressions.

Then, following (24), the variance may be approximated as

$$\text{Var}[\text{ED}'(n, m)] \approx \text{Var}[\mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n + 2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g}] \approx \text{Var}[2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g}]. \quad (25)$$

Figure 4 illustrates the variance of (22), i.e. the variance of the exact value of  $\text{ED}(n, m)$ , the variance of  $\mathbf{g}^T \mathbf{Q}^m \mathbf{g}$  which is discarded in the approximation of (24), and finally the variance of the approximation, i.e.  $\text{Var}[2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g}]$ , where all are computed empirically. We see from Figure 4 that (25) is a reasonable approximation at high values of SNR.

The approximation (25) may be computed analytically. Letting  $T \triangleq 2\mathbf{y}_n^T \mathbf{Q}^m \mathbf{g}$  and  $\mathbf{b} \triangleq \mathbf{Q}^m \mathbf{y}_n$ , we may write  $T = \sum b_i g_i$ . As  $g_i$  are independent and identically Gaussian distributed, we have that, given  $\mathbf{y}_n$ ,  $T$  is a normal random distribution with mean 0 and variance  $4\|\mathbf{Q}^m \mathbf{y}_n\|^2 \sigma_g^2$ . Averaging the variance over  $\mathbf{y}$ , the distribution of  $T$  can be approximated by  $\mathcal{N}(0, \sigma_T^2)$ , where  $\sigma_T^2$  is given by

$$\sigma_T^2 = 4\|\mathbf{Q}^m\|^2 \sigma_g^2 \sigma_y^2, \quad (26)$$

where  $\|\mathbf{Q}^m\|$  is the Frobenius norm of matrix  $\mathbf{Q}^m$ ,  $\|\mathbf{Q}^m\| \triangleq \sqrt{\sum_{i,j} |q_{i,j}^m|^2}$ . This analytic approximation is also illustrated in Fig. 4 and shown to be a reasonable approximation to the variance of the noisy hash value  $\text{Var}[\text{ED}'(n, m)]$  at high SNR values. In the following, we use for simplicity the notations  $R \triangleq (\mathbf{y}_n + \mathbf{g})^T \mathbf{Q}^m (\mathbf{y}_n + \mathbf{g})$ ,  $S \triangleq \mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n$ , and  $U \triangleq \mathbf{g}^T \mathbf{Q}^m \mathbf{g}$ .

#### 4.1.1. Method I

As the values of  $\text{ED}(n, m)$  are quantized by a simple slicer with threshold at zero to obtain the final hash, for a given  $\mathbf{y}_n$ , assuming without loss of generality  $S < 0$ , we may write the probability of error as

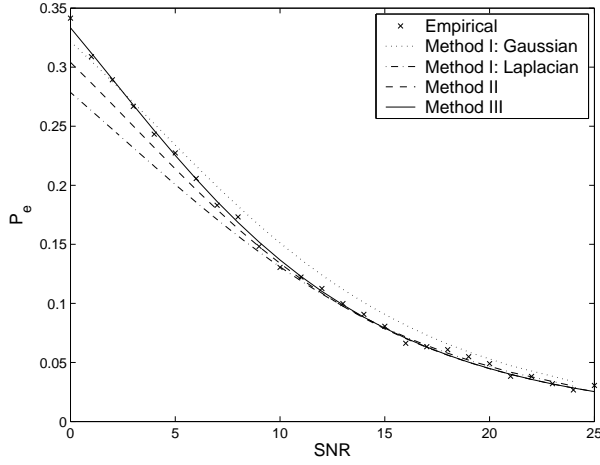
$$P_e = \text{Pr}[R > 0]. \quad (27)$$

Approximating the tails of the distribution of  $\text{ED}(n, m)$  by a normal distribution, we may apply the Gaussian error integral or  $\mathcal{Q}$ -function given by  $\mathcal{Q}(x) \triangleq \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp(-\frac{z^2}{2}) dz$ . So, assuming  $R$  and  $S$  are normally distributed and independent, where  $S$  has zero mean, we may approximate (27) as

$$P_e \simeq \mathcal{Q}\left(\frac{-\mu_R}{\sigma_R}\right), \quad (28)$$

where  $\mu_R = S$  and  $\sigma_R^2$  is approximated by  $\sigma_T^2$ , and given by (26). The resulting probability of error is given by

$$P_e \simeq \mathcal{Q}\left(\frac{|\mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n|}{2\|\mathbf{Q}^m\| \sigma_g \sigma_y}\right). \quad (29)$$



**Figure 5.**  $P_e$  under i.i.d. Gaussian noise addition.

Figure 5 illustrates the probability of error, computed from (29) and averaged over  $S$ .

Observing the empirical distribution, a Laplacian model seems more accurate. Using this model we obtain

$$P_e \simeq \frac{1}{2} \exp\left(-\sqrt{2} \frac{|\mu_R|}{\sigma_T}\right) = \frac{1}{2} \exp\left(-\frac{|\mathbf{y}_n^T \mathbf{Q}^m \mathbf{y}_n|}{\sqrt{2} \|\mathbf{Q}^m\| \sigma_g \sigma_y}\right). \quad (30)$$

From Figure 5 we see that the values for the probability of error obtained using the Laplacian p.d.f., again averaged over  $S$ , are closer to the empirical values than those obtained using the Gaussian  $Q$ -function.

#### 4.1.2. Method II

A more accurate analysis of the probability of error is now given. Again, as the values of  $\text{ED}(n, m)$  are quantized by a slicer with threshold at zero,  $P_e$  may be written as

$$P_e = \Pr[S + T + U > 0 | S < 0] + \Pr[S + T + U < 0 | S > 0]. \quad (31)$$

The variances of  $S$  and  $U$  are given by

$$\sigma_S^2 = 2 \|\mathbf{Q}^m\|^2 \sigma_y^4 \quad (32)$$

and

$$\sigma_U^2 = 2 \|\mathbf{Q}^m\|^2 \sigma_g^4, \quad (33)$$

respectively. The variance of  $T$  is given by (26).

Using the same approximation as in (24) we may write

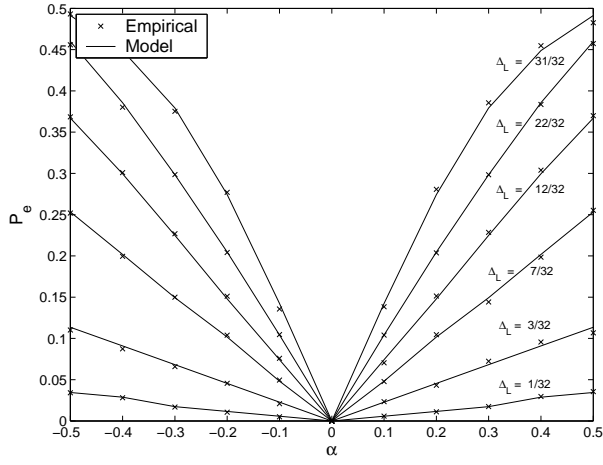
$$P_e \simeq P(S + T > 0 | S < 0) + P(S + T < 0 | S > 0) \quad (34)$$

As  $S$  is a quadratic form in central Gaussian variables, its distribution may be expressed as a weighted sum of  $\chi^2$  distributions.<sup>6</sup> In our analysis however, we assume  $S$  and  $T$  are normally distributed and independent. Then, we can obtain the probability of error (34) as

$$P_e = \int_{-\infty}^0 f_S(s) ds + \int_0^{\infty} \mathcal{Q}\left(\frac{s}{\sigma_T}\right) f_S(s) ds - \int_{-\infty}^0 \mathcal{Q}\left(\frac{s}{\sigma_T}\right) f_S(s) ds. \quad (35)$$

As the mean of  $S$  is zero,  $f_S(s)$  is symmetrically distributed about zero and we may write

$$\begin{aligned} P_e &= 2 \int_0^{\infty} \mathcal{Q}\left(\frac{s}{\sigma_T}\right) f_S(s) ds \\ &= \frac{\arctan\left(\frac{\sqrt{2}\sigma_a}{\sigma_y}\right)}{\pi}. \end{aligned} \quad (36)$$



**Figure 6.**  $P_e$  vs desynchronisation level  $\alpha$ .

While (36) is independent of  $\|Q^m\|$  we see that it matches closely empirical data. (36) represents the probability of error for a given input signal under Gaussian noise addition. The same expression for  $P_e$ , applied to general input signals and noise, is given by Doets and Lagendijk through personal communication.

### 4.1.3. Method III

We may refine the approximation of Sect. 4.1.2 by assuming  $S$ ,  $T$  and  $U$  are normally distributed, uncorrelated and therefore independent. As the autocorrelation between  $T$  and  $U$  is zero, we have that  $\sigma_{T+U}^2 = \sigma_T^2 + \sigma_U^2$ . We may write using (31)

$$P_e = \int_{-\infty}^0 f_S(s) ds + \int_0^{\infty} \mathcal{Q}\left(\frac{s}{\sqrt{\sigma_T^2 + \sigma_U^2}}\right) f_S(s) ds - \int_{-\infty}^0 \mathcal{Q}\left(\frac{s}{\sqrt{\sigma_T^2 + \sigma_U^2}}\right) f_S(s) ds, \quad (37)$$

where  $\sigma_T$  and  $\sigma_U$  are given by (26) and (33) respectively. Again assuming the mean of  $S$  is zero we may write

$$\begin{aligned} P_e &= 2 \int_0^{\infty} \mathcal{Q}\left(\frac{s}{\sqrt{\sigma_T^2 + \sigma_U^2}}\right) f_Z\left(\frac{s}{\sigma_S}\right) ds \\ &= \frac{1}{\pi} \arctan\left(\frac{\sigma_g}{\sigma_y} \sqrt{2 + \left(\frac{\sigma_g}{\sigma_y}\right)^2}\right). \end{aligned} \quad (38)$$

From Figure 5 we see that this refined approximation is accurate at all levels of SNR. In this method we have not ignored  $g^T Q^m g$  and thus, unlike the method of Section 4.1.2, we achieve accuracy for low values of SNR.

## 4.2. Desynchronisation

If a frame of  $\mathbf{x}$  is taken at random, it is unlikely that its frame indices will match exactly those of any particular frame for which a subfingerprint has been computed and stored in a database. Due to this desynchronisation, errors may occur when comparing this subfingerprint to those in the database. In this section we pursue an expression for the resulting probability of error.

Consider a frame whose first sample is  $x[n\Delta L + k]$  where  $-\frac{\Delta L}{2} < k < \frac{\Delta L}{2}$ , i.e. a frame that is desynchronised from frame  $n$  by  $k$  samples. Then the vector  $\mathbf{y}_{n,k}$  used in the computation of the corresponding hash value is defined as

$$\mathbf{y}_{n,k} \triangleq (x[(n-1)\Delta L + k], x[(n-1)\Delta L + k + 1], \dots, x[n\Delta L + L + k - 1])^T. \quad (39)$$

and the corresponding hash value, denoted  $\text{ED}(n, m, k)$ , is given by

$$\text{ED}(n, m, k) = \mathbf{y}_{n,k}^T \mathbf{Q}^m \mathbf{y}_{n,k}. \quad (40)$$

Repeating the procedure in Section 2 and 3, we may write  $\text{ED}(n, m)$  and  $\text{ED}(n, m, k)$  as quadratic forms in the same Gaussian vector by defining an extended vector

$$\mathbf{z}_{n,k} \triangleq (x[(n-1)\Delta L], x[(n-1)\Delta L + 1], \dots, x[n\Delta L + L + k - 1])^T \quad (41)$$

of length  $L + \Delta L + k$ , and two  $L + \Delta L + k \times L + \Delta L + k$  matrices

$$\hat{\mathbf{Q}}^m \triangleq \begin{bmatrix} \mathbf{Q}^m & \vdots \\ \dots & \dots \end{bmatrix}, \quad \hat{\mathbf{Q}}_k^m \triangleq \begin{bmatrix} \dots & \dots \\ \vdots & \mathbf{Q}^m \end{bmatrix}, \quad (42)$$

obtained by placing  $\mathbf{Q}^m$  at  $(1, 1)$  and  $(k+1, k+1)$  respectively. We can then write

$$\text{ED}(n, m) = \mathbf{z}_{n,k}^T \hat{\mathbf{Q}}^m \mathbf{z}_{n,k}. \quad (43)$$

and

$$\text{ED}(n, m, k) = \mathbf{z}_{n,k}^T \hat{\mathbf{Q}}_k^m \mathbf{z}_{n,k}. \quad (44)$$

Letting  $S \triangleq \text{ED}(n, m)$  (as in Section 4.1) and  $V \triangleq \text{ED}(n, m, k)$ , we can now write the probability of error resulting from a desynchronisation of  $k$  samples as

$$P_e = P(S > 0 | V < 0) + P(S < 0 | V > 0). \quad (45)$$

$S$  and  $V$  are clearly correlated and we assume that they are jointly distributed as a bivariate normal distribution. Under this assumption we may write

$$\begin{aligned} P_e &= \int_0^\infty \left( \int_{-\infty}^0 f_{S,V}(s, v) ds \right) dv + \int_{-\infty}^0 \left( \int_0^\infty f_{S,V}(s, v) ds \right) dv \\ &= 2 \int_0^\infty \int_{-\infty}^0 f_{S,V}(s, v) ds dv, \end{aligned} \quad (46)$$

where

$$f_{S,V}(s, v) = \frac{1}{2\pi\sigma_S\sigma_V\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2\sqrt{1-\rho^2}} \left( \frac{s^2}{\sigma_S^2} - \frac{2\rho sv}{\sigma_S\sigma_V} + \frac{v^2}{\sigma_V^2} \right)\right), \quad (47)$$

with the correlation coefficient  $\rho$  given by

$$\rho = \frac{2\text{tr}(\hat{\mathbf{Q}}^m \hat{\mathbf{Q}}_k^m)}{2\sqrt{\text{tr}(\hat{\mathbf{Q}}^m)\text{tr}(\hat{\mathbf{Q}}_k^m)}}. \quad (48)$$

The variances required are just  $\sigma_S^2 = 2\|\hat{\mathbf{Q}}^m\|^2\sigma_y^2$  and  $\sigma_V^2 = 2\|\hat{\mathbf{Q}}_k^m\|^2\sigma_y^2$ . Evaluating (46) we obtain the following approximation to the probability of error

$$P_e = \frac{\arccos(\rho)}{\pi}. \quad (49)$$

In Fig. 6 the probability of error is shown for a range of desynchronisations where  $k = \alpha\Delta L$ ,  $-1/2 < \alpha < 1/2$ , for a range of values of  $\Delta L$ . We see that (49) provides a good approximation to  $P_e$  at all levels of desynchronisation. Fig. 6 illustrates the relationship between the probability of error and the level of overlap, i.e. when the number of non-overlapping samples  $\Delta L$  is large, the hash size is small and less robust to desynchronisation, resulting in a higher  $P_e$ . By integrating over the distribution of  $k$ , the average probability of error may be computed.

## 5. CONCLUSION

In this paper we have presented an analysis of the statistical behaviour of a particularly robust audio fingerprinting scheme.<sup>1</sup> We initially follow the work of Doets and Lagendijk<sup>4</sup> in obtaining an equivalent rearrangement of the fingerprinting algorithm and then show that by introducing the representation of a modified periodogram as a quadratic form, the parameters of the model of Doets and Lagendijk<sup>4</sup> can be very easily computed ((17),(18),(21)) regardless of the type of windowing function employed by the algorithm. A further advantage of our representation of the hash value as a quadratic form is that we have been able to examine the intraframe dependencies (16) of the hash as well as the interframe dependencies.

Using this model we then presented an analysis of the probability of error of the hash under both Gaussian noise addition and desynchronisation of the signal. In the case of noise addition, we have presented three approximations of  $P_e$ , where the third and most accurate is shown to provide a good approximation at all levels of SNR. In the case of desynchronisation, we have given an expression for the probability of error which provides a good approximation at all levels of desynchronisation for all values of overlap.

Although the model has been developed with the method of Haitsma et al<sup>1</sup> in mind, it may find application in other similar methods. Particularly, it is applicable whenever the generation of the hash involves energy measures obtained by linear overlaps of linear transforms. A linear transform leads to a quadratic form in terms of energy measurement and the overlap can be easily taken into account following the steps undertaken in our proposal. This is the case, for example, in the audio fingerprinting method proposed by Mihcak and Venkatesan.<sup>8</sup>

## Acknowledgement

This work is kindly supported by Enterprise Ireland Advanced Technologies Research Program, research grant ATRP2002/230 and by the European Commission through the IST Programme under Contract IST-2002-507609 SIMILAR.

## REFERENCES

1. J. Haitsma, T. Kalker, and J. Oostven, "Robust audio hashing for content identification," in *Procs. of the International Workshop on Content-Based Multimedia Indexing*, (Brescia, Italy), 2001.
2. P. Doets and R. Lagendijk, "Stochastic model of a robust audio fingerprinting system," in *5th International Symposium on Music Information Retrieval (ISMIR)*, (Barcelona, Spain), October 2004.
3. P. Doets and R. Lagendijk, "Theoretical modelling of a robust audio fingerprinting system," in *Procs. of the Fourth IEEE Benelux Signal Processing Symposium*, pp. 101–104, (Hilvarenbeek, The Netherlands.), April 2004.
4. P. Doets, "Modelling a robust audio fingerprinting system," in *Technical Report, Delft University of Technology*, June 2004.
5. P. Johnson and D. Long, "The probability density of spectral estimates based on modified periodogram averages," *IEEE Transactions on Signal Processing* **47**, pp. 1255–1261, May 1999.
6. J. Imhof, "Computing the distribution of quadratic forms in normal variables," *Biometrika* **48**, pp. 419–426, December 1961.
7. S. Searle, "Linear models for unbalanced data," *John Wiley and Sons*, 1987.
8. M. Mihcak and R. Venkatesan, "A perceptual audio hashing algorithm: A tool for robust audio identification and information hiding," in *Procs. of the 4th Information Hiding Workshop*, (Pittsburgh, USA), 2001.