Proceedings of *IEEE Multimedia Systems'99*, vol. 1, pp. 574–579, 7–11 June 1999, Florence, Italy.

Evaluation of copyright marking systems

Fabien A. P. Petitcolas and Ross J. Anderson

Abstract— **Hidden copyright marks have been proposed as a solution for solving the illegal copying and proof of ownership problems in the context of multimedia objects. Many systems have been proposed, but it is still difficult to have even a rough idea of their performances and hence to compare them. So we first describe some general attacks on audio and image marking systems. Then we propose a *benchmark* to compare image making software on a fair basis. This benchmark is based on a set of attacks that any system ought to survive.**

Keywords— **Digital watermarking, benchmark, attacks.**

A number of broad claims have been made about the 'robustness' of various digital watermarking or fingerprinting methods. Unfortunately the criteria and the sample pictures used to demonstrate it vary from one system to the other, and recent attacks [1], [2], [3], [4], [5] show that the robustness criteria used so far are often inadequate. JPEG compression, additive Gaussian noise, low pass filtering, rescaling, and cropping have been addressed in most of the literature but specific distortions such as rotation have often been ignored [6], [7]. In some cases the watermark is simply said to be 'robust against common signal processing algorithms and geometric distortions when used on some standard images'.

We formed the view that useful progress might come from trying to attack all these first generation schemes and from introducing a benchmark to compare their performances. In section I we describe two general attacks which reveal significant limitations of various image and audio marking systems on image an audio marking systems. We then explain the procedure we used for our benchmark and give some results in section II.

I. Two general attacks

A. Restoration as an attack on audio marking

Audio restoration techniques have been studied for several years and have proved to be very useful to remove localised degradations (clicks, scratches, crackles, etc.) from old recordings [8], [9]. After finding the local degradations, these methods basically ignore the bad samples and interpolate the signal using the neighbouring ones.

Based on this idea we suggest an attack on copyright marking systems for audio: the watermarked signal is simply reconstructed block by block using the original data.

The method assumes that the signal to be interpolated is the realisation of a stationary autoregressive (AR) process of finite order: from a given segment of data a set of AR parameters is estimated and then used to estimate the missing samples. Both estimations are based on a least-square minimisation problem.

Suppose that the recorded data x consists of N samples $x_1, \ldots, x_N$ and is the realisation of a stationary autoregressive process of order p, i.e.

$$x_n = \sum_{k=1}^{p} a_k x_{n-k} + e_n \qquad n = p+1, \ldots, N \tag{1}$$

where $\mathbf{e} = [e_{p+1}, \ldots, e_N]^T$ is the 'excitation' noise vector. We assume that a block of l consecutive samples starting at sample $m+1$ is missing. l, m and p are chosen such that $p \leq m < m + l \leq N - p$. There are methods to estimate the order of the AR process but, $p = 3l + 2$ gives good interpolation results in general [8].

The estimators for both a and x are chosen such that they minimise the quadratic error $E(\mathbf{a}, \mathbf{x}_u) = \mathbf{e}^T\mathbf{e}$ which is a function of the unknown AR parameters $\mathbf{a} = [a_1, \ldots, a_p]^T$ and the unknown samples $\mathbf{x}_u = [x_{m+1}, \ldots, x_{m+l}]^T$.

Minimisation of E is non-trivial since it involves non-linear fourth order unknown terms but a suboptimal solution to the above problem can be used.

First E is minimised with respect to $\mathbf{a}$ taking an arbitrary initial estimate for $\mathbf{x}_u$ (typically zero) in order to obtain an estimate $\hat{\mathbf{a}}$ of $\mathbf{a}$. If we note $\mathbf{x}_1 = [x_{p+1}, \ldots, x_N]^T$, then (1) can be written $\mathbf{e} = \mathbf{x}_1 - \mathbf{U}(\mathbf{x})\mathbf{a}$ where $\mathbf{U}(\mathbf{x})$ is a $(N - p) \times p$ matrix whose coefficients are $u_{i,j} = x_{p+i-j}$. Hence,

$$E(\mathbf{a}, \mathbf{x}_u) = \mathbf{x}_1^T\mathbf{x}_1 + \mathbf{a}^T\mathbf{U}^T\mathbf{U}\mathbf{a} - 2\mathbf{a}^T\mathbf{U}^T\mathbf{x}_1 \tag{2}$$

which is minimised by setting

$$\frac{\partial E}{\partial a_k} = 0 \qquad k = 1, \ldots, p \tag{3}$$

From (2) and (3) we obtain a system of linear equations that can be used to compute $\hat{\mathbf{a}}$:

$$\mathbf{U}^T\mathbf{U}\hat{\mathbf{a}} = \mathbf{U}^T\mathbf{x}_1 \tag{4}$$

Then E is minimised with respect to $\mathbf{x}_u$ and using the value of $\hat{\mathbf{a}}$ found after the first minimisation. Equation (1) is written $\mathbf{e} = \mathbf{V}_k(\hat{\mathbf{a}})\mathbf{x}_k + \mathbf{V}_u(\hat{\mathbf{a}})\mathbf{x}_u$ where $\mathbf{x}_k = [x_1, \ldots, x_m, x_{m+l+1}, \ldots, x_N]^T$ is the vector of known samples. After minimisation (similar to the first step) the reconstructed block $\mathbf{x}_u$ is given by:

$$\mathbf{V}_u^T\mathbf{V}_u\mathbf{x}_u + \mathbf{V}_u\mathbf{V}_k\mathbf{x}_k = 0 \tag{5}$$

Both sets of linear equations (4) and (5) can be solved using singular value decomposition (SVD) method which handles large ill conditioned systems: typically $N = 1000$. These two steps can be iterated to get better results but it seems that one iteration is usually enough.

The authors are from the University of Cambridge Computer Laboratory, Pembroke Street, Cambridge CB2 3QG, UK. E-mail {fapp2, rja14}@cl.cam.ac.uk.

Part of this work was supported by Intel Corporation under the grant 'Robustness of Information Hiding Systems'.

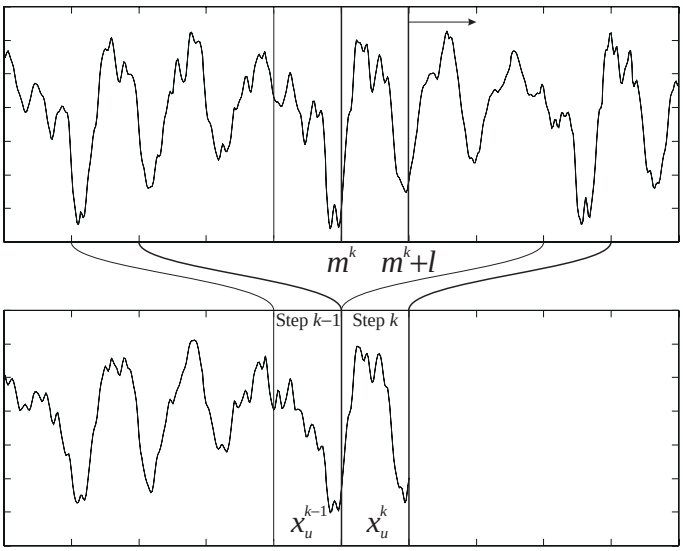

Fig. 1. One step of the algorithm.

The attack uses a watermarked audio signal as input and fully reconstructs it block by block. More precisely, it takes $N = 1000$ samples from the input and applies the restoration algorithm, described previously, to reconstruct a block of $l = 80$ samples, starting at sample $m = 460$ – hence the block is in the middle. This block is appended at the output as illustrated in figure. 1. Note that the beginning and the end of the input are not modified at all. Then a new set of N samples is extracted from the input – by shifting by l samples the sliding window – and a new reconstructed block is produced, and so on.

Note that for each restoration the original neighbouring samples are used to avoid a drift of the whole process and hence a catastrophic quality loss. Error-free restoration is theoretically possible in certain cases. But contrary to the usual applications of audio restoration we do not want the restoration to be error free. Input and output would be the same and this would not be an attack against the watermark. One can adjust the quality of the output by diminishing the number l of 'unknown' samples. In fact it has been shown that when l increases, the error variance per sample approaches the signal variance.

Other and better interpolation algorithms are available, but the least square AR interpolation technique, we briefly summarised, gives satisfactory results if the blocks are relatively small, up to 80 samples. We tried the attack against BlueSpike's method [10], which seems to be one of the contenders for the International Federation of the Phonographic Industry (IFPI) call for proposal. We used well known samples including *castanets*, *svega*, *clarinet* and *schubert*. After watermarking and then reconstruction, which did not introduce noticeable effects, the detector could not find the digital watermark in any of these.

Although we used it only against BlueSpike's method, this attack is quite general and could also be used against image marking too. Indeed, similar algorithms for image reconstruction are given in [11].

B. Random geometric distortion attack

For the case of images, there are better attacks. After evaluating some image watermarking software, it became clear to us that although most schemes could survive basic manipulations – that is, manipulations that can be done easily with standard tools, such as rotation, resampling, resizing and lossy compression – they would not cope with combinations of them or with random geometric distortions. This motivated the design of StirMark [3].

StirMark is a generic tool for basic robustness testing of image watermarking algorithms and has been freely available since November 1997.[1] It applies a minor unnoticeable geometric distortion: the image is slightly stretched, sheared, shifted, bent and rotated by an unnoticeable random amount. A slight random low frequency deviation, which is greatest at the centre of the picture, is applied to each pixel. A higher frequency displacement of the form $\lambda \sin(\omega_x x) \sin(\omega_y y) + n(x, y)$ – where $n(x, y)$ is a random number – is also added. Finally a transfer function that introduces a small and smoothly distributed error into all sample values is applied. This emulates the small nonlinear analogue/digital converter imperfections typically found in scanners and display devices. Resampling uses the approximating quadratic B-spline algorithm [12]. An example of these distortions is given in figure 2.

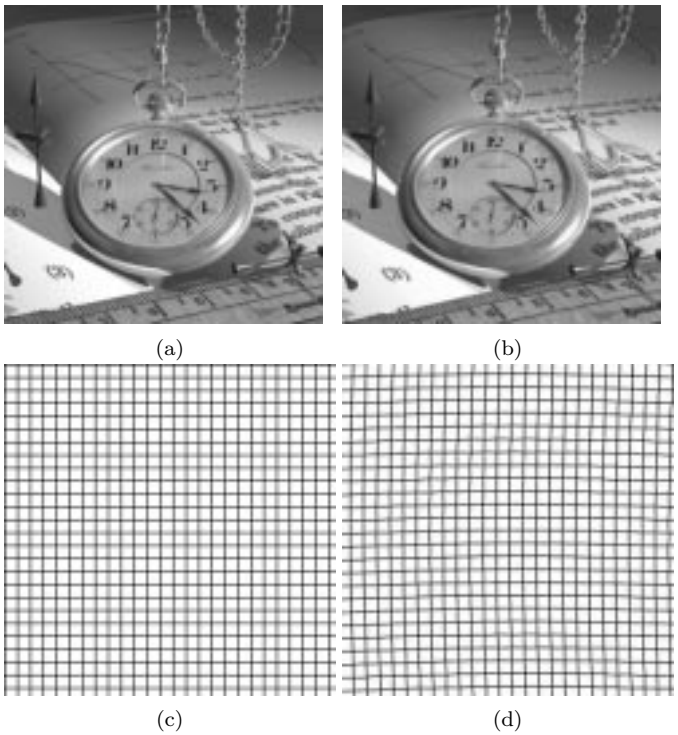

Fig. 2. When applied to images, the distortions introduced by StirMark are almost unnoticeable: *watch* before (a) and after (b) StirMark with default parameters. For comparison, the same distortions have been applied to a grid (c & d). Both images have the same size: 256×256 pixels. Copyright image courtesy of Kevin Odhner (jko@home.com).

The general lesson from this attack is that given a target

[1] <http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/>

marking scheme, one can invent a distortion (or a combination of distortions) that will prevent detection of the watermark while leaving the perceptual value of the previously watermarked object undiminished. We are not limited in this process to the distortions produced by common analogue equipment, or usually applied by end users with common image processing software. Moreover, the quality requirements of pirates are often lower than those of content owners who have to decide how much quality degradation to tolerate in return for extra protection offered by embedding a stronger signal.

II. A benchmark

Digital watermarking remains a largely untested field and very few authors have published extensive tests on their systems (e.g., [13]). A benchmark is needed to highlight promising areas of research by showing which techniques work better than others but also to be able to compare quickly new algorithms which appear in the literature. Until now most papers have used their own series of tests, their own pictures and their own methodology. So comparison is impossible without re-implementing the method and trying separately. But then, the implementation might be very different, and probably weaker, than the one of the original authors. With a common benchmark authors would just need to provide a brief table of results and other researchers would then have a fairly good idea of the performances of the proposed scheme and may make more thorough evaluation if interested.

For this benchmark we consider the watermarking process for embedding and recovering as a black box. For instance, some systems employ synchronisation templates or transformation invariants to survive some geometrical transformations. These templates help to detect specific geometrical transformation. An inverse transformation is applied to the image and the watermark is extracted from the modified image. The combination of the template and the embedding/extracting process form the watermarking algorithm as a whole, so they should be evaluated together.

A. General procedure

For each image in a determined set, we used the following procedure for our tests.

1. Embed a mark with the strongest strength which does not introduce annoying effects. In other words, embed the mark such that the quality of the output, for a given quality metric, is greater than a given minima.
2. Apply a set of given distortions.
3. For each distorted image try to extract the watermark using a success/failure approach, that is to consider the extraction successful if and only if the payload is fully recovered without error.

Note that if the watermarking tools has a command line interface which allows to modify the parameters of the embedding and to check the error rate after extraction of the watermark, this procedure can be easily automated using Unix shell or Perl scripts.

The general framework we just described still has some unknowns. First, the number of bits that can be hidden. Ideally one should try to embed at least a 70-bit watermark so one can use numbering systems such as the one mentioned by in [14]. Unfortunately this is not always possible, especially when testing commercial off-the-shelf software but this does not always matter: there are some cases where some systems hide more bits than others and still survive more attacks for the same 'quality' measure of the watermarked images.

Secondly, the metric to measure the quality of the watermarked images. Many metrics based on human perceptual models have been proposed in the literature and several marking algorithms use one of them. It is not clear yet whether the choice of such a metric would introduce a significant bias in the experiments. Indeed a watermarking algorithm using the a particular perceptual model might give better results than others for tests using a quality metric based on the same model. For our experiments we simply used the $PSNR$ as quality metric and applied the strongest watermark strength such that the $PSNR$ of the marked image is greater than 38 dB. The $PSNR$ does not take into account any property of the human visual system and is probably the worse case metric for the systems we tested. This explains, for instance, the poor results we got when using *baboon* as test image.

This is also one of the reasons why it is important to test an image watermarking software on many different images and for fair comparison the same set of sample images should always be used. Pictures can be interesting from the signal processing point of view: textured/smooth areas, size, synthetic, with straight edges, sharp, blur, brightness/contrast, etc. A general benchmark should use a broad range of contents and types of images, but one can also imagine benchmarks intended for medical or computer generated images. For our tests we used some of the 'classical' images such as *lena* or *baboon* but also new pictures given for research purpose by some photographers we contacted.

The last unknown is the set of attacks. In the next section, we review the attacks used for the proposed benchmark.

B. Attacks used in the benchmark

In addition to the random geometric distortions described section I-B, StirMark can also perform a default series of tests which serve as the basis for the benchmark: given a watermarked image, StirMark will apply these transformations with various parameters. Then the output images can be tested with watermark detection or extraction programs. So the full process can be automated.

The list of attacks actually implemented into StirMark is not exhaustive but includes most of the simple operation that users are likely to perform and also random geometric distortions. Since most artists will first apply filtering or some slight geometric transformation (e.g., rotation) and then save the image in a compressed format it makes sense to test robustness of watermarking system to geometric

transformations or filtering followed by compression. So, most experiments have been done with and without JPEG compression using 90 as quality factor.[2]

- **Low pass filtering** – This includes the following linear and non-linear filters:
  - Gaussian (blur) $\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$;
  - 3×3 median filter;
  - Simple sharpening $\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$;
  - Frequency mode Laplacian removal attack [5].
- **Colour quantisation** to 256 colours.
- **JPEG compression** – The test uses the following quality factors: 90, 85, 80, 75, 60, 50, 25, 15 and 10. Although images compressed with a very low quality factor do not have much commercial value, some marking systems do survive them. Hence using a broad scale of compression parameters gives more accurate comparison.
- **Scaling** – As we noticed earlier, scaling happens for instance, when a high resolution digital image is used for electronic applications such as Web publishing. We used uniform scaling – that is the same factor is applied to the width and height of the picture – with the following factors: 0.5, 0.75, 0.90, 1.1, 1.5, and 2.0. Each transformation is done with and without JPEG compression (quality factor 90%).
- **Cropping** – In some cases, infringers are just interested by the 'central' part of the copyrighted material, moreover more and more Web sites take advantage of image segmentation, which is the basis of the 'Mosaic' attack [3]. This is of course an extreme case of cropping. StirMark crops images (with and without JPEG compression 90) by removing 1, 2, 5, 10, 15, 20, 25, 50, or 75% of the border.
- **Rotation** – Small angle rotations, often in combination with cropping, do not usually change the commercial value of the image but can make the watermark un-detectable. Rotations are used to realign horizontal features of an image and it is certainly the first modification applied to an image after it has been scanned. For benchmarking we propose to crop the rotated image so that there is no need to add a fixed color border to it. We used the following angles: -2, -1, -0.5, 0.5, 1, 2, 5, 10, 15, 30, 45, and 90 degrees, with and without scaling – to keep original size after rotation – and cropping and with and without JPEG compression 90.
- **Other simple geometric transformation** – These simple transformations are more likely to be used by wilful infringers and include: 1 and 10% *shearing* in the X and Y directions (with and without JPEG compression 90), *removal* of 1, 5 or 10 lines and columns (with and without JPEG compression 90) and *horizontal flip*, since many images can be flipped without loosing any value. Although resilience to flipping is usually straightforward to implement, not all systems do survive it (see table I).
- **Random geometric distortions** – See section I-B.

In the next versions of StirMark, we plan to add other possible attacks such as: histogram stretching, histogram equalisation, gamma correction, restoration techniques (see section I-A), noise addition or even other specific attacks such as the one proposed by Maes [2] or by Langelaar et al. [4].

[2]We used the implementation of the Independent JPEG Group.

C. Some results

Table I shows early results based on a subset of the transformation described previously and without using any quality measurement, just the naked eye and default software parameters.[3] Although comparison should be done with great care, the table confirms what is currently achieved in term of robustness and what needs further research.

For the results summarised in table II, we followed exactly[4] the procedure detailed previously. The images used for the test were *lena*, *baboon*, *fishing boat*, *bear*, *skyline arch* and *watch*[5]. We will keep adding new results with other images but after four images we noticed that the average results were stable. Detailed results, including strength of the watermarks and $PSNR$ of the watermarked images, are at `<http://www.cl.cam.ac.uk/~fapp2/watermarking/benchmark/>`.

Two general remarks apply to these tests. First, we did not take into account the computation time which is also an important parameters, especially for the extraction process. Second, some of the tools we have tested have already been improved. For instance the method of the University of Geneva now also addresses shearing using log-log maps [15]; this was not the case for the version we tested.

III. Conclusion

Our general attack on image marking algorithms suggests that the real problem is not so much inserting the marks as recognising them afterwards. Thus progress may come not just from devising new marking schemes, but in developing ways to recognise marks subjected to distortion. Only few papers deal with this problem [15], [16].

We also proposed a benchmark that can be used to judge the robustness of image watermarking systems. For this benchmark, we have selected a set of attacks that are likely to be applied by users and that introduce an acceptable amount of degradation. Its main purpose is to give a short overview of the performances of watermarking algorithms and provide a base to compare them.

Acknowledgments

Some of the ideas presented here were clarified by discussion with Gabriela Csurka, Frédéric Deguillaume, Jean-Luc Dugelay, David Hilton, Shelby Pereira, Burt Perry and Thierry Pun. The first author is grateful to Intel Corporation for financial support under the grant 'Robustness of Information Hiding Systems'. Special thanks to the Computer Vision Group of the University of Geneva, the Digi-

[3]For this evaluation, we used the images available at `<http://ltssg3.epfl.ch:1248/kutter/watermarking/database.html>`

[4]Except for Signum SureSign for which we could not choose the strength of the embedding and used default parameters.

[5]These images are available at `<http://www.cl.cam.ac.uk/~fapp2/watermarking/benchmark/image_database.html>`

	Digimarc 1.51	**SureSign 3.0 Demo**	**EikonaMark 3.01**	**Giovanni 1.1.0.2**	**SysCoP 1.0R1**
Filtering (3×3 median, Gaussian)	100	100	100	60	80
Scaling (0.5, 0.75, 0.9, 1.1, 1.5, 2)	70	100	0	63	0
Cropping (1, 2, 5, 10, 15, 20, 25, 50 %)	100	100	0	15	0
Rotation (-2, -1, -0.5, 0.5, 1, 2)	82	58	0	10	0
JPEG (90, 85, 80, 75, 60, 50, 25, 15, 10, 5)	56	72	90	12	58
GIF Conversion	100	100	100	60	80
Horizontal flip	100	100	0	0	0
StirMark 1.0	80	80	0	0	0
StirMark 2.2	0	0	0	0	0

TABLE I

Early robustness tests (August 1998) for various digital watermarking products. Values are percentage of survival to attack. For each product 5 test images (baboon, benz, girl, glasses, and lena) have been used and for each image 42 transformations have been applied using StirMark 2. Each image has been watermarked using the best parameters that do not give obvious and annoying distortions. Although comparison should be done with great care (not all systems have the same applications, some systems are public other semi-private, etc.), the table confirms what is currently achieved and what needs further research.

	Digimarc	**Unige**	**SureSign**
Signal enhancement			
Gaussian	100	100	100
Median	100	100	100
Sharpening	100	100	100
FMLR	100	67	100
Compression			
JPEG	65	52	87
GIF/Colour quantisation	100	100	100
Scaling			
Without JPEG 90	81	81	97
With JPEG 90	72	81	83
Cropping			
Without JPEG 90	100	81	94
With JPEG 90	98	72	91
Shearing			
X	50	13	42
Y	50	4	42
Rotation			
Auto-crop	95	74	37
Auto-scale	97	77	51
Other geometric trans.			
Col. & line removal	100	69	89
Horizontal flip	100	100	100
Random geometric dist.	17	0	0

TABLE II

Summary of the results for the benchmark presented in this paper. We tested the following piece of software: Digimarc's Batch Embedding Tool 1.00.13, Digimarc's ReadMarc 1.5.8, the watermarking tool of the University of Geneva (version 15 January 1999) and Signum Technologies' SureSign Server 1.94. The partition in the table means that the conditions of the experiments were slightly different for SureSign as explained in the body of this paper.

marc Corporation and the Signum Technologies Limited for providing software needed for this evaluation.

References

[1] J.-P. M. G. Linnartz and M. van Dijk, "Analysis of the sensitivity attack against electronic watermarks in images." In Aucsmith [20], pp. 258–272, ISBN 3-540-65386-4.

[2] M. Maes, "Twin peaks: The histogram attack on fixed depth image watermarks." In Aucsmith [20], pp. 290–305, ISBN 3-540-65386-4.

[3] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems." In Aucsmith [20], pp. 218–238, ISBN 3-540-65386-4.

[4] G. C. Langelaar, R. L. Lagendijk, and J. Biemond, "Removing spatial spread spectrum watermarks by non-linear filtering." In *9th European Signal Processing Conference (EUSIPCO'98)*, pp. 2281–2284, Island of Rhodes, Greece, 8–11 Sep. 1998, ISBN 960-7620-05-4.

[5] R. Barnett and D. E. Pearson, "Frequency mode L.R. attack operator for digitally watermarked images." *Electronics Letters*, vol. 34, no. 19, pp. 1837–1839, Sep. 1998.

[6] J. J. K. Ó Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking." *Signal Processing*, vol. 66, no. 3, pp. 303–317, May 1998, ISSN 0165-1684, European Association for Signal Processing (EURASIP).

[7] M. Kutter, "Watermarking resisting to translation, rotation, and scaling." In *Proceedings of SPIE International Symposium on Voice, Video, and Data Communications*, vol. 3528, pp. 423–431, Boston, U.S.A., Nov. 1998.

[8] A. Janssen, R. N. Veldhuis, and L. B. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes." *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 317–330, Apr. 1986.

[9] S. V. Vaseghi, *Algorithms for restoration of archived gramophone recording*. Ph.D. thesis, Emmanuel College, University of Cambridge, England, Feb. 1988.

[10] "Giovanni audio marking software." Blue Spike company, `<http://www.bluespike.com/>`, May 1998.

[11] R. Veldhuis, *Restoration of lost samples in digital signals*. International Series in Acoustics, Speech and Signal Processing, Hertfordshire, England: Prentice Hall, 1990.

[12] N. A. Dodgson, "Quadratic interpolation for image resampling." *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1322–1326, Sep. 1997, ISSN 1057-7149.

[13] G. W. Braudaway, "Results of attacks on a claimed robust digital image watermark." In van Renesse [18], ISBN 0-8194-2556-7.

[14] J.-F. Delaigle, "Common functional model." Deliverable AC019-UCL-TEL-DR-P-D12-b1, CEC, 29 Mar. 1996, tracing Author's Rights by Labelling Image Services and Monitoring Access Project.

[15] S. Pereira, J. J. K. O'Ruanaidh, F. Deguillaume, G. Csurka, and T. Pun, "Template based recovery of fourier-based watermarks using log-polar and log-log maps." In *International Conference on Multimedia Systems (ICMS'99)*, IEEE, Firenze, 7–1 Jun. 1999, to appear.

[16] F. Hartung, J. K. Su, and B. Girod, "Spread spectrum watermarking: Malicious attacks and counterattacks." In Wong and Delp [19], pp. 147–158, ISBN 0-8194-3128-1.

[17] J. Dittmann, P. Wohlmacher, P. Horster, and R. Steinmetz, eds., *Multimedia and Security – Workshop at ACM Multimedia'98*, vol. 41 of *GMD Report*, Bristol, United Kingdom, Sep. 1998, ACM, GMD – Forschungszentrum Informationstechnik GmbH, Darmstadt, Germany.

[18] R. L. van Renesse, ed., *Optical Security and Counterfeit Deterrence Techniques II*, vol. 3314, San Jose, California, U.S.A., 28–30 Jan. 1998, The Society for Imaging Science and Technology (IS&T) and the International Society for Optical Engineering (SPIE), SPIE, ISBN 0-8194-2556-7.

[19] P. W. Wong and E. J. Delp, eds., *Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, California, U.S.A., 25–27 Jan. 1999, The Society for Imaging Science and Technology (IS&T) and the International Society for Optical Engineering (SPIE), SPIE, ISBN 0-8194-3128-1.

[20] D. Aucsmith, ed., *Information Hiding: Second International Workshop*, vol. 1525 of *Lecture Notes in Computer Science*, Portland, Oregon, U.S.A., 1998, Springer-Verlag, Berlin, Germany, ISBN 3-540-65386-4.