Robust Image-Adaptive Data Hiding: Modeling, Source Coding and Channel Coding*

K. Solanki, O. Dabeer, U. Madhow, B. S. Manjunath, and S. Chandrasekaran
Dept. of Electrical and Computer Engineering
University of California at Santa Barbara,
Santa Barbara, CA 93106
Email: {solanki, onkar, madhow, manj, shiv}@ece.ucsb.edu

Abstract

This paper provides a summary of our work over the past two years on robust, high-volume data hiding in images. We first present a basic framework for image-adaptive hiding, which allows selection of the coefficients in which to hide, and employs powerful "turbo-like" erasures and errors codes in a novel manner to prevent desynchronization of encoder and decoder due to selective embedding. This coding framework provides robustness against a variety of attacks, including compression, tampering and moderate resizing. Next, we provide a joint source-channel coding scheme for image-in-image hiding, in which the quality of the recovered signature image is better if the attack is milder. This is achieved by hybrid digital-analog hiding. Finally, we present preliminary results on hiding techniques that survive printing and scanning. The techniques are devised after experimental modeling of the print-scan channel.

1 Introduction

A common application of data hiding is the application of a digital watermark to multimedia content to be protected. Such a watermark must be robust against a variety of attacks that seek to destroy it without excessively degrading the content, but it does not need to carry a large amount of information. In contrast, the thrust of our work is to hide significantly larger volumes of data, while sacrificing some generality in terms of the kinds of attacks that the hidden data must survive. Thus, our approach follows classical communication-theoretic principles, in that we optimize our signal and code design for the hiding channel induced by certain specific attacks. Applications include steganography (covert communication), seamless upgrade of multimedia systems (by embedding data that newer equipment can read, without affecting the performance of old equipment), embedding metadata, and embedding authentication and other information on identification documents such as passports or driver's licenses.

In this progress report on our work over the past two years, we summarize some work detailed in prior publications [1, 2, 3, 4], and describe some preliminary results

*This research is supported in part by a grant from ONR # N00014-01-1-0380.

that have not appeared elsewhere. In accordance with information-theoretic guidelines [5, 6], we employ quantization index modulation [7] in the transform domain. In Section 2, we describe a hiding scheme originally designed to survive compression, but which ends up being robust to a variety of other attacks because of its use of powerful codes. The novel feature of this framework is the flexibility it provides in allowing selection of the coefficients in which to embed data. In Section 3, we build upon this digital hiding scheme to provide a "joint source-channel hiding" scheme for image-in-image hiding. This is a hybrid digital-analog technique in which the quality of the recovered hidden image is better if the attack is milder, a "graceful improvement" which is not possible using purely digital hiding. In Section 4, we consider the problem of surviving more drastic attacks, such as printing (digital-to-analog conversion) followed by scanning (analog-to-digital conversion). We present preliminary results on experimental modeling of the print-scan channel, and on the performance of an image-adaptive hiding scheme whose design is guided by the channel model. Our conclusions are presented in Section 5.

2 Image-adaptive Hiding via Selective Embedding in Coefficients

In order to robustly hide large volumes of data in images without causing significant perceptual degradation, hiding techniques must adapt to local characteristics within an image. Many prior quantization based blind data hiding schemes use global criteria regarding where to hide the data, such as statistical criteria independent of the image (e.g. embedding in low or mid-frequency bands), or criteria matched to a particular image (e.g. embedding in high-variance bands). These are consistent with information theoretic guidelines [6], which call for hiding in "channels" in which the host coefficients have high variance. This approach works when hiding a few bits of data, as in most watermarking applications. However, for hiding large volumes of data, hiding based on such global statistical criteria can lead to significant perceptual degradation.

Our selective embedding in coefficients (SEC) scheme is an image-adaptive technique in which data is embedded only in those discrete cosine transform (DCT) coefficients whose magnitude is greater than a predefined integer threshold. An 8×8 DCT of non-overlapping blocks is taken and the coefficients are divided by the JPEG quantization matrix at design quality factor. We embed in these coefficients only if its magnitude exceeds a positive integer threshold. As this threshold increases, fewer coefficients qualify for embedding, and hence less data can be hidden, which provides a tradeoff between hiding rate and perceptual quality. For thresholds ≥ 2, it becomes difficult for a human observer to distinguish between the original and composite image, while embedding reliably at fairly high rates. For example, in 512×512 peppers image, and threshold of 2, one can hide about 2800 bits such that it survives 0.4 bpp JPEG compression (QF=25) while maintaining complete perceptual transparency. The simplest SEC scheme is the zero-threshold SEC scheme, where the coefficients that are not quantized to zero, are used to embed information. High embedding rates are achieved using this scheme with very low perceptual degradation, which resembles JPEG compression.

Table 1: Number of hidden information bits reliably recovered under various attacks using RA coded SEC scheme for 512×512 Lena image. 20 coefficients per block formed the candidate embedding band in all the cases.

Attack	JPEG compr. QF=25	AWGN 15 dB	0.8 bpp Wavelet compr.	20% Resizing bicubic interp.
# of bits	7447	6301	7447	6301
RA code rate	1/11	1/13	1/11	1/13

2.1 Coding Framework

While selective embedding reduces perceptual degradation, it can cause desynchronization of encoder and decoder. Distortion due to attack may cause an insertion (decoder guessing that there is hidden data where there is no data) or a deletion (decoder guessing that there is no data where there was data hidden). Any embedding scheme that employs local adaptive criterion can, in principle, incur this problem.

To solve this problem, we introduce the concept of erasures at the encoder. The bit stream to be hidden is coded, using a low rate code, assuming that all host coefficients that meet the global criteria (i.e., lie in a *candidate embedding band*) will actually be employed for hiding. A code symbol is erased at the encoder if the local perceptual criterion for the coefficient is not met. Since we code over entire space of coefficients that lie in a designated low-frequency band, long codewords can be constructed to achieve very good correction ability.

Any turbo-like code that operates close to Shannon limit for the erasures channel, while possessing a reasonable error-correcting capability, could be used with the SEC scheme. We use repeat-accumulate (RA) codes [8] in our experiments because of their simplicity and near-capacity performance for the erasure channels. A rate 1/q RA encoder involves q-fold repetition, pseudorandom interleaving and accumulation of the resultant bit-stream. Decoding is performed iteratively using the sum-product algorithm [9].

The coding framework, initially designed to counter insertion/deletion, also provides robustness to the hidden data against various attacks such as additive white Gaussian noise (AWGN), image resizing, wavelet based compression (JPEG 2000), and image tampering. Table 1 shows the number of bits hidden for 512×512 Lena image under these attacks along with the RA code rate used. Readers are referred to [2] and [3] for more extensive discussion.

2.2 Image Tampering

The preceding coding framework can also deal with *image tampering* wherein a part of image is replaced by some other image data, either locally or globally. In order to survive tampering, the code rate used is further lowered to survive the errors caused due to the replacement of the image data. The code rate is a design parameter shared by encoder and decoder: if a tampering attack is anticipated, then a low enough code rate should be chosen beforehand.

Once the hidden bitstream is decoded, localization of the tampered area can be done by reconstructing the originally encoded bitstream (using the same RA code parameters) and determining the error locations. If the host image has undergone tampering, then most of the errors would be concentrated at the locations where the tampering was done. Such an ability to robustly decode the bitstream and then localize the tampered area

(a) 20 % of 512×512 Lena image tampered

(b) 512×512 Lena image tampered globally

(c) Localization of tampered area at the decoder for the globally tampered image

Figure 1: Local and global image tampering and localization of the tampered area.

is potentially useful in medical and forensic applications to detect whether a malicious attacker has tampered with "evidence".

Figure 1(a) shows an example attacked image where 20% of the image is tampered. The 5,280 hidden bits are correctly decoded in the presence of this attack. Figure 1(b) shows globally tampered image and again, the 6,301 hidden bits are recovered successfully. Figure 1 (c) shows the localization results for this globally tampered image.

3 Joint Source-Channel Hiding

The purely digital hiding scheme in the previous section exhibits the cliff effect common to all coded digital communication systems: if the actual attack is more severe than the design level, then there is a catastrophic failure, but if the actual attack is milder, no benefit in performance is gained. When the hidden data has a specific interpretation, better performance may be obtained using joint source-channel hiding. We illustrate this via the example of image-in-image hiding. Since the attack level is seldom known *a priori,* our design goal is to provide graceful improvement in the quality of the recovered hidden image as the attack becomes milder. While joint source-channel coding for attaining similar objectives have been studied for the Gaussian channel, to the best of our knowledge, this approach has not been previously employed in data hiding. We now summarize our work on hybrid digital analog (joint source-channel) hiding reported in [4]: the digital scheme in Section 2 is employed to send digital data describing the hidden image, while the analog residue is sent using a new method. Focusing on JPEG attacks, graceful improvement in recovered image quality is demonstrated as the level of JPEG compression decreases.

3.1 Hiding Analog Information

To hide an analog number m into a host sample h, we first quantize the host h using a quantizer of step size Δ, and then replace the residue with the source m, which has been companded or scaled to lie in the interval $(0, \Delta)$. Let us consider an example where

$\Delta = 1$ and the host symbol is, say, 6.235. We want to send a source symbol whose value is 0.729 (a real number $\in (0, \Delta)$) through the hiding channel. The encoder first determines that the host symbol lies between 6 and 7 (an interval $(n\Delta, (n+1)\Delta)$), then it sends the source symbol directly within that interval, i.e., it just sends 6.729. In practice, we use a hiding strategy that always *measures* the message m from an even reconstruction point of the host. This is done to avoid catastrophic error when a hidden coefficient switches to a different integer interval as a result of attack. Thus, the symbol y to be sent for hiding a message m into a host symbol h is given by,

$$\begin{aligned} y &= \Delta(\lfloor h/\Delta \rfloor) + m, \text{ if } \lfloor h/\Delta \rfloor \text{ is even,} \\ &= \Delta(\lfloor h/\Delta \rfloor + 1) - m, \text{ if } \lfloor h/\Delta \rfloor \text{ is odd.} \end{aligned} \tag{1}$$

Here, $\lfloor \cdot \rfloor$ denotes the `floor` operation (defined as the largest integer smaller than or equal to the given number).

We consider varying levels of JPEG attacks assuming that the attack level is known only to the decoder, but not to the encoder. The JPEG compression performs uniform quantization of the DCT coefficients of 8×8 blocks of the image. We derive the MMSE decoder for the above hiding scheme under uniform quantization attack, when the reconstruction points of the attack quantizer are known to the decoder [4].

3.2 Image-in-Image Hiding Implementation

The encoding process can be divided into the following parts.

Processing the signature image: This step involves separating the signature image into digital and analog parts. The image is compressed using JPEG to generate a bitstream, which constitutes the digital part. The analog part is obtained by computing the residual errors of pre-selected DCT coefficients after the quantization based on design *signature* quantization matrix. Note that, the design quality factor, and the number of analog residues chosen to send, are predetermined at the design stage.

Allocating the channels: Here, we allocate the host coefficients (i.e., channel) for the digital and analog parts respectively. A few low frequency coefficients (other than the DC coefficient) of each 8×8 host block are reserved for the analog channel. Remaining low and/or mid frequency coefficients are dedicated to the digital channel. Thus the decoder would know where to look for analog and digital data respectively.

Hiding the digital part: The digital bitstream is hidden into its allocated channel using the RA-coded SEC scheme (Section 2, and [2, 3]).

Hiding the analog part: The analog residues of selected low frequency coefficients are sent through its allocated channel using the scheme discussed in the previous Section. Since the residue always lies in $[0, \Delta_{sig})$, where Δ_{sig} is the design signature quantizer, we simply scale it to lie in $[0, 1)$.

The analog and the digital parts are decoded separately and then combined to give an estimate of the sent signature image. An example of hiding a 128×128 image into a 512×512 image at a design QF of 25 is presented here. Figure 2 shows the recovered signature images when the host image undergoes JPEG compression at varying levels, starting from the worst case QF of 25. The signature image is JPEG compressed at QF = 10 to form the digital part and the residues of 16 low frequency coefficients make up the analog part. We use one coefficient from each 8×8 host block for transmitting the analog data. 34 coefficients constitute the *digital channel*. Note the improvement in decoded quality as the attack becomes milder.

(a) attk. QF = 25 (93.5% compr.), MSE = 0.0286

(b) attk. QF = 45 (88.7% compr.), MSE = 0.0193

(c) attk. QF = 65 (85.0% compr.), MSE = 0.0119

(d) attk. QF = 85 (75.8% compr.), MSE = 0.0043

(e) No attack

Figure 2: Hiding a 128×128 peppers image into a 512×512 harbour image (not shown here). The signature images received after various levels of JPEG compression are shown along with the corresponding observed MSE per coefficient.

4 Print-Scan Resilient Data Hiding

With the use of powerful coding framework, the technique presented in Section 2 is robust against attacks such as compression, image tampering, and image resizing. However, due to its block based hiding mechanism, this method is not robust to print-scan operation, filtering, or geometric transformations. In this section we propose a scheme based on a global image transform that survives some of these attacks, especially the print-scan process.

Many print-scan resilient watermarking methods have been proposed, that embed a watermark into an image, which can be detected after the image is printed and scanned (e.g., [10],[11],[12]). Lin et al [10] propose a model for the print-scan process by considering pixel value and geometric distortions separately. They propose a watermarking method based on log polar map of discrete Fourier transform (DFT) magnitudes (i.e., the Fourier-Mellin transform). Technique proposed in [11] also involves DFT magnitudes but the watermark itself is made circularly symmetric so that the log polar coordinate transformation is not required. Bas et al [12] use geometrically invariant feature points to embed the watermark.

Our approach, based on selective embedding in low frequency DFT magnitudes, is related, yet significantly different from the above schemes. First, we attempt to hide more information into the images than just a single bit watermark. For example, we can hide about 500 bits in a 512×512 peppers image that survives the print-scan operation with 7% errors (without coding). Second, we hide in dynamically selected low frequency DFT magnitudes unlike other DFT based schemes (e.g., [10], [11]), where embedding is done in a predefined set of mid-frequency coefficients. The hiding scheme, initially designed to survive the print-scan process, turns out to be robust against a variety of other attacks such as those in *Stirmark* [13], e.g., heavy JPEG compression, rows and columns removal, Gaussian or median filtering, aspect ratio change, and random bending. The hidden data also survives a very limited amount of rotation and cropping that might happen during the scanning process. Note that the scheme is not designed to survive heavy rotation or cropping.

4.1 The Print-Scan Channel

We conducted experiments with several grayscale images in order to understand the print-scan process. In the experiments, we assume a degree of control over the printing and the scanning operation[1]. The images were printed at high resolution, with several dots dedicated to one pixel, e.g., a 512×512 image is printed onto an A4 page with "fit page" option so that the size of printed image on the paper is about 8"×8". We use a commercially available printer and scanner (specifically, Lexmark optra S 1620 laser printer and CanoScan N670U flatbed scanner). The images were printed and scanned at varying resolutions (300 to 1200 dpi for printing, and 75 to 1200 dpi for scanning). At the time of scanning, the image is cropped and resized using bicubic interpolation to its original size. Note that explicit registration of the image features is not performed after scanning since it is assumed that the original image is not available at the decoder. We studied the Fourier spectra of the original and scanned images and made the following observations regarding the print-scan channel behavior.
1) The low and mid frequency bands are preserved much better than the high frequencies. In general, the lower the frequency, the better its chances of surviving the print-scan process.
2) In the low and mid frequency bands, the coefficients with low magnitudes get washed out, while those with high magnitudes are preserved much better. It can be seen from Figure 3 that the coefficients with low magnitudes are hit more severely than their neighbors with higher magnitudes. This is a significant characteristic of the channel and has been observed consistently for different images and various printer or scanner resolutions.
3) Coefficients with higher magnitudes (who do not get severely corrupted) see a gain of roughly unity. This means that the images do not undergo filtering during the process. Although this is contrary to the print-scan model proposed in [10], it can be explained by the fact that several dots are dedicated to a pixel of the printed image, so that blurring does not occur.
4) Slight modifications to the selected high magnitude low frequency coefficients does not cause significant perceptual distortion to the image.

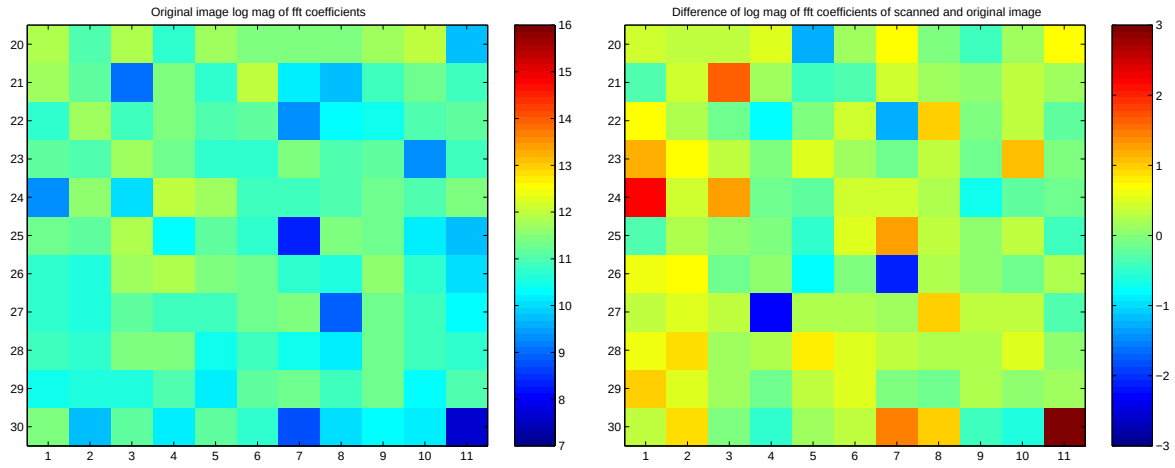
Based on these observations, we propose a method where data is hidden dynamically into selected low frequency coefficients, as described in the following section.

4.2 Selective Embedding in Low Frequencies

Information is hidden only in the DFT coefficients that lie in a low frequency band, and also satisfy a threshold criteria. Hence the name: selective embedding in low frequencies (SELF).

Consider an N×N host image in which data is to be hidden. Let us denote the natural logarithm of the magnitudes of 2D DFT of the whole image by c_{ij}, $0 \leq i, j \leq N-1$. We embed in a given coefficient c_{ij} only if it lies in a predetermined frequency band and also exceeds a threshold t_{ij}. Let us define the band as an indicator function b_{ij}, such that if $b_{ij} = 1$, the coefficient c_{ij} lies in the band. Note that b_{ij}, t_{ij} and the quantization interval Δ are design parameters that are shared between the encoder and the decoder. Embedding is done using choice of scalar quantizers. We send either $Q_1(c_{ij})$ or $Q_0(c_{ij})$

[1]This is expected to hold in a document authentication application, in which standardized machines could be employed.



(a) Original image spectrum

(b) Difference in log DFT magnitudes of scanned and original image

Figure 3: Print-scan channel: Almost all *dark blue* coefficients in the original image spectrum of (a) correspond to *dark red* points in the log transfer function of (b), e.g., (24,1),(25,7),(30,11), and so on. It indicates that the error is high for all coefficients that have low magnitudes.

Table 2: Number of bits hidden and error rates for various 512×512 images for the print-scan attack.

Image	# of bits hidden	Error % age	Insertion % age	Deletion % age
Peppers	488	1.64 %	3.07 %	3.89 %
Bridge	512	1.17 %	6.05 %	6.25 %
Baboon	1026	1.85 %	5.75 %	8.67 %

depending on the bit to be hidden. Thus, the modified coefficient, d_{ij} can be given as

$$d_{ij} = \begin{cases} Q_{b_l}(c_{ij}) & \text{if } b_{ij} = 1, \text{ and } c_{ij} > t_{ij}, \\ c_{ij} & \text{otherwise.} \end{cases} \tag{2}$$

Also note that symmetry of the DFT coefficients is maintained during the hiding process by modifying two symmetric coefficients in the same manner so that the inverse DFT gives real values.

4.3 Experimental Results

Table 2 shows the number of bits hidden for various 512×512 images along with the error, insertion, and deletion rates. Incorporation of a coding framework to counter insertions, deletions and errors is an area of future work. Figure 4 (a) and (b) show original and hidden bridge images respectively. Figure 4 (c) shows the image after it is printed and scanned. In spite of this distortion, the 512 hidden bits can be decoded with 10% errors.

The images with hidden data also survive other attacks such as Gaussian or median filtering, heavy JPEG compression, and aspect ratio change. Table 3 lists the performance of the hiding method against these attacks for various image. The "overall error" (defined

(a) Original 512×512 Bridge image

(b) image with 512 bits hidden

(c) Printed and scanned image

Figure 4: Print-scan example: the original, hidden, and attacked images.

as # errors + # insertions + 1/2· # deletions) is listed here to save space. Much less data can be hidden against the random bending attack. For example, 73 bits are hidden in peppers image and received with 20 % errors. Note that this performance is still good for watermarking applications, where the watermark sequence is known to the decoder and can be correlated with the hidden data to *detect* the watermark.

Table 3: Performance of the proposed SELF hiding scheme against various attacks.

Images	# bits hidden	Attacks: Overall error percentage					
		Print-Scan	JPEG compr. QF=10	3×3 Gaussian filter	4×4 Median filter	17 rows 5 cols removed	Aspect ratio change 0.8×1.0
Barbara	367	7.63%	1.77%	0%	2.72%	2.45%	0.27%
Man	1076	15.75%	8.59%	0.09%	3.86%	5.62%	0.09%
Couple	364	10.03 %	4.81%	0%	1.64%	1.24%	0.55 %

5 Conclusions

The main features of the proposed hiding techniques may be summarized as follows: (a) image adaptivity, which operationally reduces to selective embedding in coefficients; (b) a powerful coding framework, which provides flexibility in adaptation (based on the key concept of erasures at the encoder) as well as robustness, and (c) designs guided by an experimental understanding of the channel induced by the specific attacks that we wish to survive.

We believe that the preceding features will remain important as we push the frontiers of data hiding technology further. We feel that the key open issue today is to find a "killer app" for data hiding, and to tailor a hiding scheme for that application.

Some theoretical issues that are worth investigating further include finding optimal (or at least improved) approaches to some of the problems that we have proposed working solutions for, including our approach to dealing with selective use of coefficients for hiding, and our approach to joint source-channel coding for the hiding channel.

References

[1] K. Solanki, N. Jacobsen, S. Chandrasekaran, U. Madhow, and B. S. Manjunath, "High-volume data hiding in images: Introducing perceptual criteria into quantization based embedding," in *Proc. ICASSP*, Orlando, FL, USA, May 2002.

[2] N. Jacobsen, K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Image adaptive high volume data hiding based on scalar quantization," in *Proc. IEEE Military Comm. Conf. (MILCOM)*, Anaheim, CA, USA, Oct. 2002.

[3] K. Solanki, N. Jacobsen, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Robust image-adaptive data hiding based on erasure and error correction," Accepted for publication *IEEE Trans. on Image Processing*.

[4] K. Solanki, O. Dabeer, B. S. Manjunath, U. Madhow, and S. Chandrasekaran, "A joint source-channel coding scheme for image-in-image data hiding," in *Proc. ICIP*, Barcelona, Spain, Sept. 2003.

[5] M. H. M. Costa, "Writing on dirty paper," *IEEE Trans. on Info. Theory*, vol. 29, no. 3, pp. 439–441, May 1983.

[6] P. Moulin and M. K. Mihcak, "A framework for evaluating the data-hiding capacity of image sources," *IEEE Trans. on Image Processing*, vol. 11, no. 9, pp. 1029–1042, Sept. 2002.

[7] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. on Info. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[8] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for turbo-like codes," in *36th Allerton Conference on Communications, Control, and Computing*, Sept. 1998, pp. 201–210.

[9] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Info. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[10] C. Y. Lin and S. F. Chang, "Distortion modeling and invariant extraction for digital image print-and-scan process," in *Intl. Symp. on Multimedia Information Processing (ISMIP 99)*, Dec. 1999.

[11] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-D DFT domain," *IEEE Transactions on Image Processing*, vol. 10, no. 11, pp. 1741–1753, Nov. 2001.

[12] P. Bas, J.-M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1014–1028, Sept. 2002.

[13] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Workshop Information Hiding, IH'98, LNCS 1525, Springer-Verlag*, 1998, pp. 219–239.