Optimal Transform Domain Watermark Embedding Via Linear Programming

S. Pereira, S. Voloshynoskiy, and T. Pun

University of Geneva - CUI, 24 rue General Dufour, CH 1211,Geneva 4, Switzerland
http://cuiwww.unige.ch/~vision/
E-mail: {Shelby.Pereira,svolos,Thierry.Pun}@cui.unige.ch

Abstract

Invisible Digital watermarks have been proposed as a method for discouraging illicit copying and distribution of copyright material. In recent years it has been recognized that embedding information in a transform domain leads to mo re robust watermarks. A major difficulty in watermarking in a transform domain lies in the fact that constraints on the allowable distortion at any pixel may be specified in the spatial domain. The central contribution of the paper is the proposal of an approach which takes into account spatial domain constraints in an optimal fashion. The main idea is to structure the watermark embedding as a linear programming problem in which we wish to maximize the strength of the watermark subject to a set of linear constraints on the pixel distortions as determined by a masking function. We consider the special cases of embedding in the DCT domain and wavelet domain using the Haar wavelet and Daubechies 4-tap filter in conjunction with a masking function based on a non-stationary Gaussian model, but the algorithm is applicable to any combination of transform and masking functions. Our results indicate that the proposed approach performs well against lossy compression such as JPEG and other types of filtering which do not change the geometry of the image.

Key words: watermark, linear programming, copyright, DCT, wavelet

---

1 Introduction

The idea of using a robust digital watermark to detect and trace copyright violations has stimulated significant interest among artists and publishers in recent years. Podilchuk [1] gives three important requirements for an effective watermarking scheme: transparency, robustness and capacity. Transparency refers to the fact that we would like the watermark to be invisible. The watermark should also be robust against a variety of possible attacks by pirates.

These include robustness against compression such as JPEG, scaling and aspect ratio changes, rotation, cropping, row and column removal, addition of noise, filtering, cryptographic and statistical attacks, as well as insertion of other watermarks [2]. The other requirement is that the watermark be able to carry a certain amount of information i.e. capacity. In order to attach a unique identifier to each buyer of an image, a typical watermark should be able to carry at least 60-100 bits of information. However, most of the work in watermarking has involved a one bit watermark. That is, at detection a binary decision is made as to the presence of the watermark most often using hypothesis testing [3]. Barni [4] encodes roughly 10 bits by embedding 1 watermark from a set of 1000 into the DCT domain. The recovered watermark is the one which yields the best detector response.

Watermarking methods can be divided into two broad categories: spatial domain methods such as [5,6] and transform domain methods . Transform domain methods have for the most part focused on DCT [1,4,7], DFT [8,9] and most recently wavelet domain methods [1,10,11]. Transform domain methods have several advantages over spatial domain methods. Firstly, it has been observed that in order for watermarks to be robust, they must be inserted into the perceptually significant parts of an image. For images these are the lower frequencies which can be marked directly if a transform domain approach is adopted [12]. Secondly, since compression algorithms operate in the frequency domain (for example DCT for JPEG and wavelet for EZW) it is possible to optimize methods against compression algorithms as will be seen in section 3. Thirdly, certain transforms are intrinsically robust to certain transformations. For example, the DFT domain has been successfully adopted in algorithms which attempt to recover watermarks from images which have undergone affine transformations [8].

While transform domain watermarking clearly offers benefits, in some cases it is desireable to specify constraints in another domain (spatial or another transform domain). In this case the problem is more challenging since it is more difficult to generate watermarks in one domain while taking into account constraints in another. For example, the problem arises since constraints on the acceptable level of distortion for a given pixel may be specified in the spatial domain. In the bulk of the literature on adaptive transform domain watermarks, a watermark is generated in the transform domain and then the inverse transform is applied to generate the spatial domain counterpart. The watermark is then modulated as a function of a spatial domain mask in order to render it invisible. However this spatial domain modulation is suboptimal since it changes the original frequency domain watermark. In the case of a DFT domain watermark, multiplication by a mask in the spatial domain corresponds to convolution of the magnitude of the spectrum. Unfortunately, to correctly account for the effects of the mask at decoding a deconvolution problem would have to be solved. This is known to be difficult and to our knowledge in the context of watermarking this problem has not been addressed. Methods proposed in the literature simply ignore the effects of the mask at decoding. One alternative which has recently appeared is the attempt at specifying the mask directly in the transform domain and ignoring spatial domain masking [1]. However other authors (e.g. Swanson [13]). have noted the importance of masking in the spatial domain even after a frequency domain mask has been applied. It should be noted that masking in one domain is not easily formulated since defining a spatial mask influences a frequency mask and vice-versa.

In this publication we develop a new approach for the mathematical modelling of the embedding process. In particular, we derive an optimized strategy for embedding a watermark in the wavelet and DCT domains when the masking constraints are specified in the spatial domain. In fact, the key idea is to optimize the encoding of the watermark with respect to the detector while using all available information about the image. This framework overcomes the problems with many proposed algorithms which adopt a suboptimal spatial domain truncation and modulation as determined by masking constraints. Furthermore we will develop an algorithm which is image dependent. Unlike many of the embedding strategies described in the literature which treat the image as noise possibly modelled by a probability distribution, the algorithm we describe uses information about the image at embedding. We consider only the problem of generating watermarks which are robust against attacks that do not change the geometry of the image. We will work with an 80 bit watermark which corresponds to a capacity sufficient for most watermarking applications. We begin in section 2 by presenting the spatial domain masking methods we adopt in the rest of the paper. In section 3 the embedding algorithm is described and applied to the case of DCT domain embedding. Then, in section 4 we derive a new channel coding strategy which greatly improves the performance of the underlying algorithm. In section 5 we show how the algorithm can be applied in the wavelet domain. In section 6 we present our results and a comparison of the DCT and wavelet domain algorithms followed by the conclusion in section 7.

2 Spatial Domain Masking

Recently Voloshynovskiy proposed a spatial domain texture masking method based where the image is first modelled as the sum of the local mean and an error term, the latter of which is modelled by a generalized Gaussian distribution as detailed in [14]. A noise visibility function (NVF) at each pixel position is then obtained as:

$$NVF(i,j) = \frac{w(i,j)}{w(i,j) + \sigma_x^2}, \tag{1}$$

where $w(i,j) = \gamma[\eta(\gamma)]^{\gamma} \frac{1}{||r(i,j)||^{2-\gamma}}$ and $r(i,j) = \frac{x(i,j)-\bar{x}(i,j)}{\sigma_x}$,$\eta(\gamma) = \sqrt{\frac{\Gamma(\frac{3}{\gamma})}{\Gamma(\frac{1}{\gamma})}}$ and $\Gamma(t) = \int_0^{\infty} e^{-u} u^{t-1} du$ is the gamma function. The parameter γ is called the *shape parameter* which is in the range $0.3 \leq \gamma \leq 1$, and $\bar{x}(i,j)$ is the local mean calculated in a 3×3 window.

The particularities of this model are determined by the shape parameter γ and the global image variance σ_x^2. To estimate the shape parameter, we use the *moment matching* method in [15,16]. The shape parameter for most real images is in the range $0.3 \leq \gamma \leq 1$. Once we have computed the noise visibility function we can obtain the allowable distortions by computing:

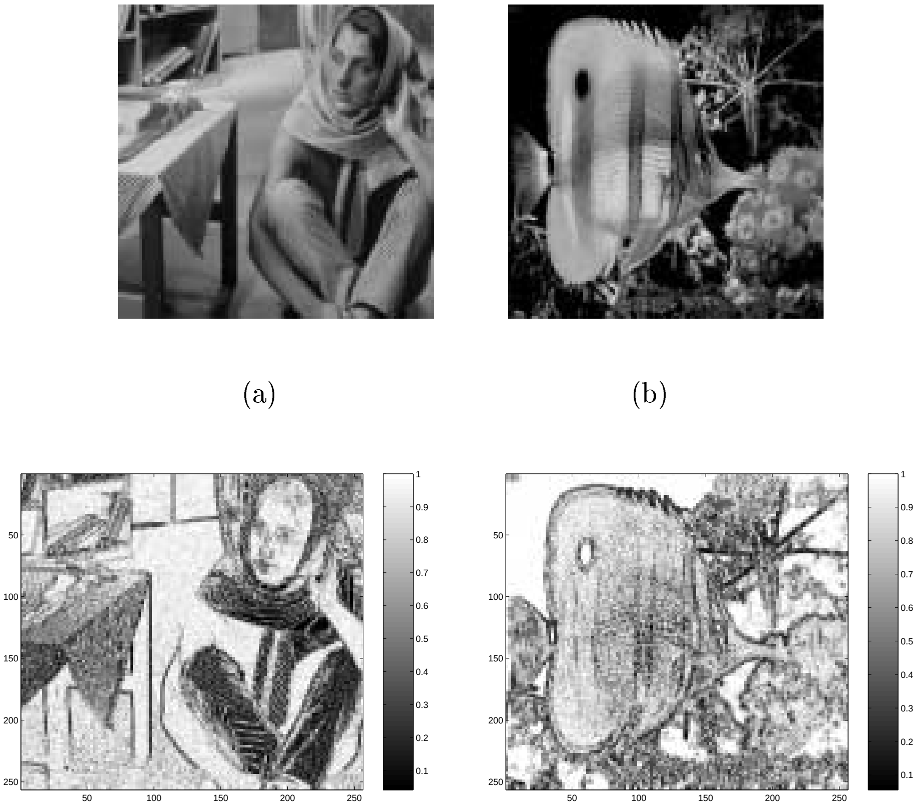
$$\Delta_{pi,j} = (1 - NVF(i,j)) \cdot S + NVF(i,j) \cdot S_1 \tag{2}$$

where S_0 and S_1 are the maximum allowable pixel distortions in textured and flat regions respectively Typically S_0 may be as high as 30 while S_1 is usually about 3. We note that in flat regions the NVF tends to 1 so that the first term tends to 0 and consequently the allowable pixel distortion is at most S_1 which is small. Intuitively this makes sense since we expect that the watermark distortions will be visible in flat regions and less visible in textured regions. Examples of NVFs for two images are given in figure 2. We note that the model correctly identifies textured and flat regions. In particular the NVF is close to 0 in textured regions and close to 1 in flat regions.

While this model accurately models textures, the importance of luminance masking has also been noted in the literature. In particular at high luminance levels the sensitivity of the HVS follows Weber's law which states that $\frac{\delta l}{l} = k_{Weber}$ where δl is the local change in luminance and l is the luminance of the background. At lower luminance levels the HVS is more sensitive to noise. Osberger [17] uses the DeVries-Rose law at low luminance levels (typically $< l_{th} = 10cd/m^2$) which states that $\frac{\delta l}{l} = \sqrt{\frac{l}{l_{th}}} * k_{Weber}$. The complete curve is shown in 2. The x-axis contains the luminance between 0 and 255 while the y-axis indicates the contrast sensitivity threshold (CST) given in terms of the change in luminance divided by the luminance. In order to incorporate luminance masking into the model we propose multiplying the texture component by $(1 + k * CST(x_{i,j}))$ to obtain

$$\Delta_{pi,j} = (1 + k \cdot CST(x_{i,j})) \cdot (1 - NVF(i,j)) \cdot S_0 + NVF(i,j) \cdot S_1 \tag{3}$$

We do not multiply the non-texture component since experiments indicate this becomes visible when increased. Experiments indicate that choosing $k = 5$ yields good results. This corresponds to increasing the allowable distortion by 5 in textured areas where the luminance level is high.



(a) (b)

(c) (d)

Fig. 1. Original images of Barbara (a) and Fish (b) along with their NVF as determined by a generalized gaussian model (c) and (d).

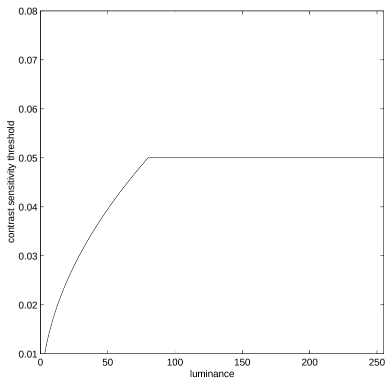


Fig. 2. Sensitivity of the HVS to changes in luminance

3 Problem Formulation

Having derived the spatial domain masking methods, we now mathematically formulate the embedding process as a constrained optimization problem. We assume that we are given an image to be watermarked denoted $\boldsymbol{I}$. If it is an

RGB image we work with the luminance component. We are also given a masking function $\boldsymbol{V}(\boldsymbol{I})$ which returns 2 matrices of the same size of $\boldsymbol{I}$ containing the values $\Delta_{pi,j}$ and $\Delta_{ni,j}$ corresponding to the amount by which pixel $I_{i,j}$ can be respectively increased and decreased without being noticed. We note that these are not necessarily the same since we also take into account truncation effects. That is pixels are integers in the range $0-255$ consequently it is possible to have a pixel whose value is 1 which can be increased by a large amount, but can be decreased by at most 1. In the general case, the function $\boldsymbol{V}$ can be a complex function of texture,luminance, contrast, frequency and patterns, however we choose to use the masking functions described in section 2. We wish to embed $\boldsymbol{m} = (m_1, m_2 ... m_M)$ where $m_i \in \{0, 1\}$ and M is the number of bits in the message. In general, the binary message may first be augmented by a checksum and/or coded using error correction codes to produce a message $\boldsymbol{m}_c$ of length $M_c = 512$. Without loss of generality we assume the image $\boldsymbol{I}$ is of size 128×128 corresponding to a very small image. For larger images the same procedure is adopted for each 128×128 large block. To embed the message, we first divide the image into 8×8 blocks. In each 8×8 block we embed 2 bits from $\boldsymbol{m}_c$. In order to embed a 1 or 0 we respectively increase or decrease the value of a DCT coefficient in order to change the sign of the coefficient if necessary. Once the DCT domain watermark has been calculated, we compute the inverse DCT transform and add it to the image in the spatial domain. At decoding, we take the sign of the DCT coefficient, apply the mappings $(+ \rightarrow 1)$,$(- \rightarrow 0)$ and then decode the BCH codes to correct possible errors.

The central problem with this scheme is that during embedding we would like to increase or decrease the DCT coefficients as much as possible for maximum robustness, but we must satisfy the constraints imposed by $\boldsymbol{V}$ in the spatial domain. In order to accomplish this, we formulate the problem for each 8×8 block, as a standard constrained optimization problem as follows. For each block we select 2 mid-frequency coefficients in which we will embed the information bits. We then have:

$$\min_{\boldsymbol{x}} \boldsymbol{f}^t \boldsymbol{x} \qquad ; \qquad \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b} \tag{4}$$

$\boldsymbol{x} = [x_{11} \dots x_{81} x_{12} \dots x_{82} \dots x_{18} \dots x_{88}]^t$ is the vector of DCT coefficients arranged column by column. $\boldsymbol{f}$ is a vector of zeros except in the positions of the 2 selected coefficients where we insert a -1 or 1 depending on whether we wish to respectively increase or decrease the value of the coefficients as determined by $\boldsymbol{m}_c$. $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$ contains the constraints which are partitioned as follows.

$$\boldsymbol{A} = \begin{bmatrix} IDCT \\ ---- \\ -IDCT \end{bmatrix} ; \boldsymbol{b} = \begin{bmatrix} \Delta_p \\ ---- \\ \Delta_n \end{bmatrix} \tag{5}$$

where IDCT is the matrix which yields the 2D inverse DCT transform of $\boldsymbol{x}$ (with elements of the resulting image arranged column by column in the vector). We take Δ_p and Δ_n to be column vectors where the elements are taken column wise from the matrices of allowable distortions. Stated in this form the problem is easily solved by the well known Simplex method. Stated as such the problem only allows for spatial domain masking, however many authors [18] suggest also using frequency domain masking. This is possible by adding the following constraints:

$$\boldsymbol{L} \leq \boldsymbol{x} \leq \boldsymbol{U} \tag{6}$$

Here $\boldsymbol{L}$ and $\boldsymbol{U}$ are the allowable lower and upper bounds on the amount we by which we can change a given frequency component. The Simplex method can also be used to solve the problem with added frequency domain constraints.

We note that by adopting this framework, we in fact allow *all* DCT coefficients to be modified (in a given 8×8 block) even though we are only interested in 2 coefficients at decoding. This is a novel approach which has not appeared in the literature. Other publications select a subset of coefficients to mark while leaving the rest unchanged. This is necessarily suboptimal relative to our approach. We are in fact trying to maximize the detector response within spatio-frequency percitibility bounds.

4 Watermark Embedding Based on Magnitude

Rather than coding based on the sign of a coefficient as in [7], we propose using the magnitude of the coefficient. To encode a 1 we will increase the *magnitude* of a coefficient and to encode a 0 we will decrease the *magnitude*. At decoding a threshold T will be chosen against which the magnitudes of coefficients will be compared. The coding strategy is summarized in table 1 where c_i is the

Table 1
Magnitude Coding

sign(c_i)	bit	Coding
$+$	0	decrease c_i (set $\boldsymbol{L}$ to stop at 0)
$-$	0	increase c_i (set $\boldsymbol{U}$ to stop at 0)
$+$	1	increase c_i
$-$	1	decrease c_i

selected DCT coefficient. The actual embedding is performed by setting $\boldsymbol{f}$ in equation 4 based on whether we want to increase or decrease a coefficient.

The major advantage of this scheme over encoding based on the sign is that the image is no longer treated as noise. As noted by Cox [19] this is an important characteristic of the potentially most robust schemes since *a priori* information on the information is used to maximize decoder response. Clearly the best schemes should not treat the image as noise since it is known at embedding. In our case, based on the observed image DCT coefficient we encode as indicated in table 1. At decoding the image is once again not noise since it contributes to the watermark. Another important property of this scheme is that it is highly image dependent. This is an important property if we wish to resist against the watermark copy attack [20] in which a watermark is estimated from one image (tyically by denoising) and added to another image to produce a fake watermark. If this is done, the watermark will be falsely decoded since at embedding and decoding the marked image is an integral part of the watermark. Consequently changing the image implies changing the watermark.

It is also possible to incorporate JPEG quantization tables into the model in order to increase the robustness of the algorithm. Assume for example that we would like to aim for resistance to compression at JPEG quality factor 10. Table 2 contains the threshold value below which a given DCT coefficient will be set to 0. In order to improve the performance of the algorithm we can add

Table 2
Thresholds at JPEG quality factor 10

30	30	30	40	60	100	130	130
35	35	35	50	65	130	130	130
40	35	45	45	65	130	130	130
40	45	60	75	130	130	130	130
50	55	95	130	130	130	130	130
60	90	130	130	130	130	130	130
125	130	130	130	130	130	130	130
130	130	130	130	130	130	130	130

bounds based on the values in table 2 to the amount we increase a coefficient. In particular, if we wish to embed a 1 we need only increase the magnitude of a coefficient to the threshold given in table 2 in order for it to survive a compression at quality factor 10. This is accomplished by setting the bounds $\boldsymbol{L}$ and $\boldsymbol{U}$. Since 2 bits are embedded per block, the remaining energy may be used to embed the other bit. It is important to note that it may not be possible to achieve the threshold since our visibility constraints as determined by $\boldsymbol{V}$ in the spatial domain must not be violated, however the algorithm will embed as much as much energy as possibly via the minimization in equation 4. We note that we choose only to embed the watermark in randomly chosen coefficients where the value in table 2 is less than 70 since for larger values we will require more energy to be sure that the coefficient survives at low JPEG compression. We avoid the 4 lowest frequency components in the upper left hand part of the DCT block since these tend to be visible even with small modifications.

5 Wavelet Domain Embedding

In order to embed the message, in the wavelet domain, we perform a similar optimization as the one performed in the DCT domain. We first divide the image into 16×16 blocks and perform the 1-level wavelet transform. In order to embed a 1 or 0 we adopt a differential encoding strategy in the lowest subband (LL). In particular we choose four neighbouring coefficients and increase two coefficients while decreasing the other two. The choice of which two to increase or decrease is a function of whether we wish to encode a 1 or a 0 so that at decoding we take the difference between the sums of the two pairs of coefficients and apply the mappings $(+ \rightarrow 1)$,$(- \rightarrow 0)$. We note that it is important to select a 2×2 block of *neighbouring* coefficients since the underlying assumption is that the difference on average is 0. In order to embed the largest possible values while satisfying masking constraints, the problem is formulated for each 16×16 block as a constrained optimization problem. In the case of the Haar wavelet, for a 16×16 block, we have 64 coefficients available in the LL subband. In each block we encode 8 bits by selecting 32 coefficients grouped into 8 2×2 blocks. We then have equation 4 as before however $\boldsymbol{x} = [x_{1,1} \ldots x_{16,1} x_{12} \ldots x_{16,2} \ldots x_{1,16} \ldots x_{16,16}]^t$ is the vector of coefficients arranged column by column. Furthermore, $\boldsymbol{f}$ is a vector of zeros except in the positions of the selected coefficients where we insert a (-1) or (1) depending on whether we wish to respectively increase or decrease the value of a coefficient. The constraints are now partitioned as:

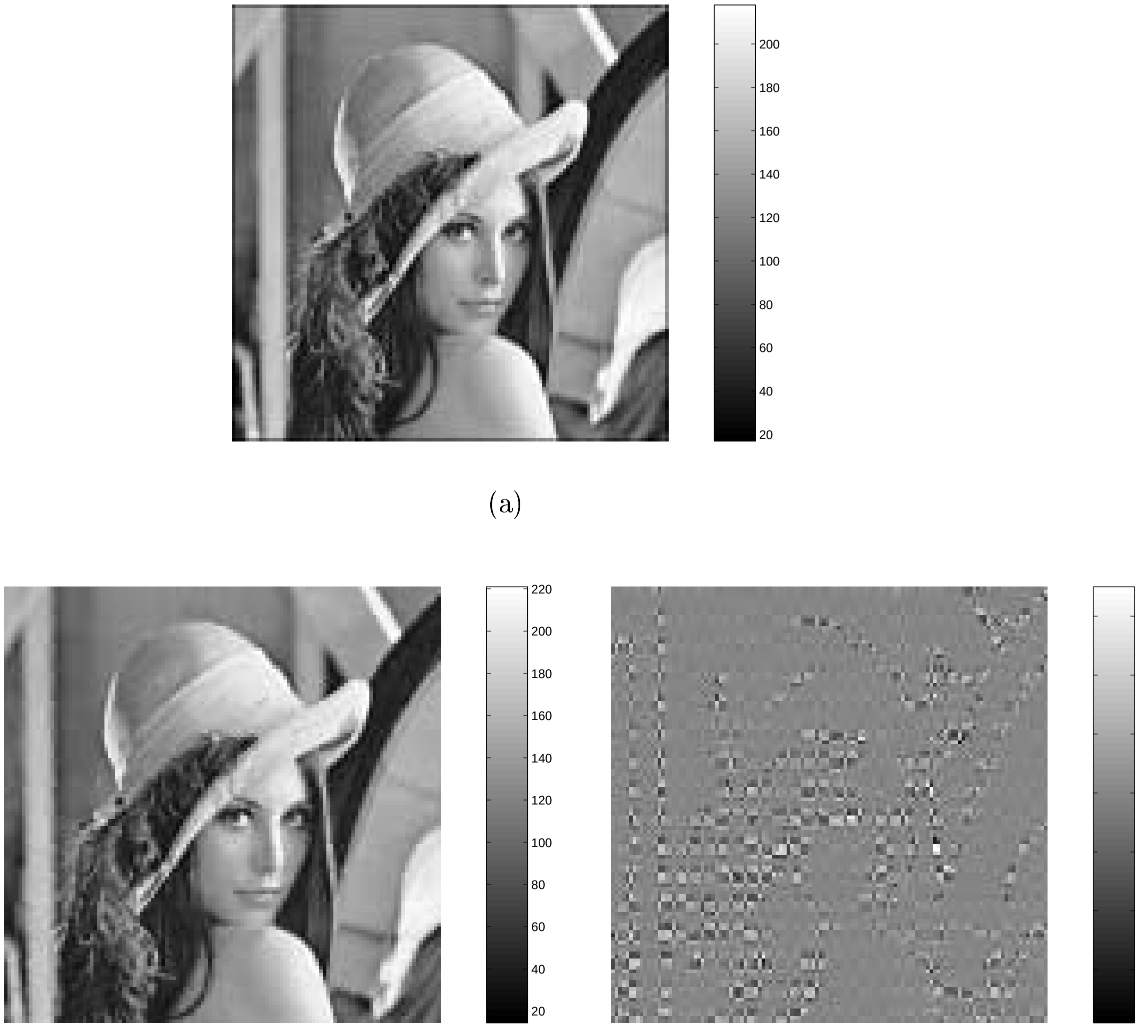
$$\boldsymbol{A} = \begin{bmatrix} IDWT \\ ---- \\ -IDWT \end{bmatrix}; \ \boldsymbol{b} = \begin{bmatrix} \Delta_p \\ ---- \\ \Delta_n \end{bmatrix} \tag{7}$$

where IDWT is the matrix which yields the 2D inverse DWT transform of $\boldsymbol{x}$ (with elements of the resulting image arranged column by column in the vector). We also note that we take Δ_p and Δ_n to be column vectors where the elements are taken columnwise from the matrices of allowable distortions. The problem is once again solved by the Simplex method. Furthermore, extra constraints in the frequency domain can also be incorporated as before via equation 6. Unfortunately, in this case it is not possible to optimize the method relative to JPEG compression since the quantization matrice are specified in the DCT domain.

6 Results

Both algorithms based on magnitude coding were tested on 5 small images. These were Lena, Bear, Girl, Boat, and Watch which were all resized to 128×128. Prior to embedding the 80 bit message, we first append a 20 bit checksum and then encode the message using turbo codes [21] to yield a binary message of length 512. Turbo codes provide near optimum performance for Gaussian channels and are consequently superior to other codes used currently in watermarking (mostly BCH and convolution). We note that even though this channel is not Gaussian, tests indicate that turbo codes outperform BCH codes [22]. In fact, JPEG compression introduces quantization noise which is difficult to model. However, the development of optimal codes for quantization channels is well beyond the scope of this paper, but is discussed by Eggers [23]. The 20 bit checksum is essential in determining the presence of the watermark. At detection if the checksum is verified we can safely say (with probability $\frac{1}{2^{20}}$ of error) that a watermark was embedded and successfully decoded.

With respect to wavelet domain watermarking, both the Haar wavelet and the Daubechies 4-tap filter were tested. In the case of the Haar wavelet, the algorithm was resistant down to a level of 70% JPEG quality factor. Better results were obtained for the 4-tap Daubechies filter where the algorithm is robust down to a level of 50% quality factor and is resistant as well to low and high pass filtering. By resistant, we understand that all the bits are correctly decoded and the checksum verified. We note that for the case of the Daubechies 4-tap filter, some minor modifications must be made to the embedding strategy. In particular, when taking the inverse DWT we obtain a block size which is bigger than the original block. These boundary problems are well known in the wavelet literature. The difficulties are easily overcome by imposing that the extra boundary pixels be constrained to be 0. This is done in practice by setting the appropriate values in Δ_p and Δ_n to 0.1 and -0.1 respectively. We do *not* set these all the way to zero since often this leads to an overly constrained problem. An example is given in figure 3 where the original image (128×128), watermarked image (Daubechies 4-tap filter) and watermark (difference between original and watermarked) are presented. We observe that the watermark is stronger in textured regions as expected. We note that the watermark is slightly visible along the long vertical edge to the left of the image. This is a limitation of the visibility model which does not take into account the high amount of structure to which the eye is particularly sensitive. In order to overcome this problem more sophisticated models



(a)

(b)

(c)

Fig. 3. Original image Lena(a) along with watermarked image (b) and watermark in (c)=(a)-(b).

are being developed which take into account the presence of lines in the image. In these regions, the allowable distortion must be reduced. Maximizing the strength of the watermark while minimizing visibility in an automatic way over a wide range of images is a delicate problem since each image is unique and presents its own difficulties.

On a Pentium 233MHZ computer the algorithm takes 20 minutes to embed the watermark. This time is non-negligible. The problem arises from the fact that a formidable optimization problem must be solved at embedding. That is at each block we have $2 * 16 * 16 = 512$ constraints. On the other hand the optimization in each block is independent once the global mask has been calculated. Consequently the algorithm can be carried out in parallel.

The results for DCT domain embedding relative to JPEG compression were far superior than for the wavelet domain. Indeed, it is possible to recover 80 bits after compression at JPEG quality factor 10% for a 128×128. This suggests that there is much to be gained by matching the transform domain of the watermark with the transform domain where compression takes place. We note that the resulting DCT watermarked image is visually indistinguishable from the wavelet domain watermarked image presented in figure 3. This results from the fact that the maximal distortions in both cases are specified by the same spatial domain mask. Once again, as with wavelet domain embedding, DCT domain embedding requires roughly 20 minutes for a 128×128 image. Unfortunately, the algorithm cannot be applied to a full frame DFT. The problem lies in the excessive computational load. For a 128×128 image, if we wish to structure the embedding problem in the magnitude of the DFT, we will have $128 * 128 * 2 = 32,768$ constraints which is overwhelming and more importantly, it is impossible to adapt to local frequency charactersitics of the image, inducing poor imperceptibility .

7 Conclusion and Further Research Directions

In this article we have described a new mathematical model which describes the process of embedding a watermark in a transform domain when the masking constraints are specified in the spatial domain. This model has 5 characteristics which make it extremely appealing:

(1) The algorithm is extremely flexible in that constraints as determined by masking functions can be easily incorporated in the spatial domain and any linear transform domain may be used although here we considered the special cases of the Haar and Daubechies wavelets as well as DCT domain embedding. Also, extra constraints may be added in the frequency domain.
(2) We show how to handle problems with truncation in an optimal way and propose the novel approach of modifying all coefficients even though we are only interested in a subset.
(3) The algorithms resist well against JPEG compression and we observe in particular that matching the embedding domain with the compression domain and incorporating JPEG quantization tables at the embedding stage leads to considerable gains.
(4) The algorithm generates a non-additive and image dependent watermark which resists the watermark copy attack [20].
(5) At the embedding stage the image is not treated as noise which is an important property of the most robust watermarking schemes as noted by Cox [19]. In fact the algorithm uses available information about the image at the embedding stage to maximize the decoder response.

While much has been accomplished by structuring the problem of watermarking within this framework, many new research directions arise. We note five possiblities in particular:

(1) While the DCT domain algorithm resists well against JPEG compression further research is needed in order to adapt the wavelet domain approach so that it is resistant against EZW and SPIHT compression.

(2) Work is currently also under way to apply the ideas of [8] so as to make the algorithm resistant to geometric changes as well.

(3) Another topic of further research is the incorporation of more sophisticated spatial domain masks. Most of the masks proposed in the watermarking literature model texture, luminance and/or frequency. Osberger [17] however identifies several higher order factors which have been used to weight distortion metrics (typically the distortion produced by compression algorithms). The most important factors are:
   - **Contrast:** Regions which have a high contrast with their surrounds attract our attention and are likely to be of greater visual importance.
   - **Size:** Larger regions attract our attention more than smaller ones however a saturation point exists after which the importance due to size levels off .
   - **Shape:** Regions which are long and thin (such as edges) attract more attention than round flat regions.
   - **Colour:** Some particular colours (red) attract more attention. Further more the effect is more pronounced when the colour of a region is distinct from that of the background.
   - **Location:** Humans typically attach more importance to the center of an image.
   - **Foreground/Background:** Viewers are more interested in objects in the foreground than those in the background.
   - **People:** Many studies have shown that we are drawn to focus on people in a scene and in particular their faces, eyes, mouth and hands.

   We note that these factors are specified in the spatial domain and not easily converted to the frequency domain. Further work could involve incorporating these elements in the attempt to generate more accurate spatial domain masks although detection of these elements is difficult to automate.

(4) While some work has been done in capacity (e.g. [24]) the bulk of the results concern additive watermarks. An interesting topic of further research is the calculation of the capacity of the proposed non-additive scheme.

ACKNOWLEDGMENTS

We are grateful to Dr. Alexander Herrigel and Digital Copyright Technologies for their work on the security architecture for the digital watermark and for the ongoing collaboration. We also thank Frédéric Deguillaume for many fruitful discussions. This work is financed by the Swiss Priority Program on Information and Communication Structures (project Krypict) and by the European Esprit Open Microprocessor Initiative (project JEDI-FIRE). This work is part of the European Patent application EU 978107084.

References

[1] C. I. Podilchuk and W. Zeng, "Image-adaptive watermarking using visual models," *IEEE Journal on Selected Areas in Communications* **16**, pp. 525–539, May 1998.

[2] F. A. P. Petitcolas and R. J. Anderson, "Attacks on copyright marking systems," in *2nd International Information Hiding Workshop*, pp. 219–239, (Portland, Oregon, USA), April 1998.

[3] G. Voyatzis and I. Pitas, "The use of watermarks in the protection of digital multimedia products," *Proceedings of the IEEE* **87**, July 1999.

[4] M. Barni, F. Bartolini, V. Cappellini, A. Piva, and F. Rigacci, "A M.A.P. identification criterion for DCT-based watermarking," in *EUSIPCO'98*, (Rhodes, Greece), September 1998.

[5] W. Bender, D. Gruhl, and N. Morimoto, "Method and apparatus for data hiding in images," *U.S. Patent # 5689587* , 1996.

[6] I. Pitas, "A method for signature casting on digital images," in *Proceedings of the IEEE Int. Conf. on Image Processing ICIP-96*, pp. 215–218, (Lausanne, Switzerland), Sept.16-19 1996.

[7] S. Pereira and T. Pun, "A framework for optimal adaptive dct watermarks using linear programming," in *Tenth European Signal Processing Conference (EUSIPCO'2000)*, (Tampere, Finland), sep 5–8 2000.

[8] S. Pereira and T. Pun, "Fast robust template matching for affine resistant watermarks," in *3rd International Information Hiding Workshop*, (Dreseden, Germany), September 1999.

[9] M. Barni, F. Bartolini, A. D. Rosa, and A. Piva, "Capacity of the watermark-channel: How many bits can be hidden within a digital image ?," in *SPIE: Security and Watermarking of Multimedia Contents*, vol. 3657, pp. 437–448, (San Jose, California), January 1999.

[10] M. Barni, F. Bartolini, V. Cappellini, A. Lippi, and A. Piva, "A DWT-based technique for spatio-frequency masking of digital signatures," in *Security and Watermarking of Multimedia Contents*, P. W. Wong and E. J. Delp, eds., vol. 3657, pp. 31–39, The Society for Imaging Science and Technology (IS&T) and the International Society for Optical Engineering (SPIE), SPIE, (San Jose, California, U.S.A.), 25–27 Jan. 1999.

[11] W. Zhu, Z. Xiong, and Y. Q. Zhang, "Multiresolution watermarking for images and video," *IEEE Transactions on Circuits and Systems for Video Technology* **9**, pp. 545–550, June 1999.

[12] I. Cox, J. Killian, T. Leighton, and T. Shamoon., "Secure spread spectrum watermarking for images, audio and video.," in *Proceedings of the IEEE Int. Conf. on Image Processing ICIP-96*, pp. 243–246, (Lausanne, Switzerland), 1996.

[13] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Multiresolution scene-based video watermarking using perceptual models," *IEEE Journal on Selected Areas in Communications* **16**, pp. 540–550, May 1998.

[14] S. Voloshynovskiy, A. Herrigel, N. Baumgaertner, and T. Pun, "A stochastic approach to content adaptive digital image watermarking," in *Third International Workshop on Information Hiding*, (Dresden, Germany), September 29 - October 1st 1999.

[15] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. PAMI* **11**, pp. 674–693, 1989.

[16] D. Kazmer, P. Barkan, and K. Ishii, "Quantifying design and manufacturing robustness through stochastic optimization techniques," in *ASME Design Engineering Technical Conferences and Computers in Engineering Conference, 96-DETC/DAC-1600*, (Irvine, California), August 1996.

[17] W. Osberger, N. Bergmann, and A. J. Maeder, "An automatic image quality assessment technique incorporating higher level perceptual factors," in *IEEE ICIP-98*, (Chicago,USA), October 1998.

[18] M. D. Swanson, B. Zhu, and A. Tewfik, "Robust data hiding for images," in *7th IEEE Digital Signal Processing Workshop*, pp. 37–40, (Loen, Norway), September 1996. G:WM1-A23.

[19] I. J. Cox, M. L. Miller, and A. L. McKellips, "Watermarking as communications with side information," *Proceedings of the IEEE* **87**, pp. 1127–1141, July 1999.

[20] M. Kutter, S. Voloshynovskiy, and A. Herrigel, "Watermark copy attack," in *IS&T/SPIE's 12th Annual Symposium, Electronic Imaging 2000: Security and Watermarking of Multimedia Content II*, P. Wah Wong and E. J. Delp, eds., vol. 3971 of *SPIE Proceedings*, (San Jose, California USA), 23–28 jan 2000.

[21] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Comm.* , pp. 1261–1271, October 1996.

[22] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Content adaptive watermarking based on a stochastic multiresolution image modeling," in *EUSIPCO 2000*, (Tampere, Finland), submitted 2000.

[23] J. J. Eggers and B. Girod, "Quantization effects on digital watermarks," *Signal Processing* , submitted 2000.

[24] S. Servetto, C. Podilchuk, and K. Ramchandran, "Capacity issues in digital image watermarking," in *IEEE Int. Conference on Image Processing ́98 Proceedings*, Focus Interactive Technology Inc., (Chicago, Illinois, USA), October 1998.