Image watermarking using block site selection and DCT domain constraints

Adrian G. Borş and Ioannis Pitas

Department of Informatics
University of Thessaloniki
Thessaloniki 540 06, Greece

pitas@zeus.csd.auth.gr

Abstract: In this paper we propose an image watermarking algorithm based on constraints in the Discrete Cosine Transform (DCT) domain. An image watermarking algorithm has two stages: signature casting (embedding) and signature detection. In the first stage it embeds an identifying label in the image. This is recognized in the second stage. The proposed algorithm has two processing steps. In the first step certain pixel blocks are selected using a set of parameters while in the second step a DCT coefficient constraint is embedded in the selected blocks. Two different constraint rules are suggested for the parametric modification of the DCT frequency coefficients. The first one embeds a linear constraint among certain selected DCT coefficients and the second defines circular detection regions according to the given parameters. The watermarks cast by the proposed algorithm are resistant to JPEG compression and filtering.

OCIS codes: (100.0100) Image Processing

References

1. B. M. Macq, J.-J Quisquater, " Cryptology for Digital TV Broadcasting," *Proc. of the IEEE,* vol. 83, no. 6, pp. 944-957, June 1995.
2. W. Bender, D. Gruhl, N. Morimoto, "Techniques for data hiding," MIT Media Lab., Technical Report 1995.
3. E. Koch, J. Zhao, "Towards robust and hidden image copyright labeling," *Proc. of IEEE Workshop on Nonlinear Signal and Image Processing,* Neos Marmaras, Greece, pp. 452-455, 20-22 June 1995.
4. O. Bruyndonckx, J.-J Quisquater, B. Macq, "Spatial method for copyright labeling of digital images," *Proc. of IEEE Workshop on Nonlinear Signal and Image Processing,* Neos Marmaras, Greece, pp. 456-459, 20-22 June 1995.
5. M. Barni, F. Bartolini, V. Capellini, A. Piva, "A DCT-domain system for robust image watermarking," *Signal Processing,* vol. 66, no. 3, pp. 357-372, 1998.
6. I. Pitas, T. H. Kaskalis, "Applying signatures on digital images," *Proc. of IEEE Workshop on Nonlinear Signal and Image Processing,* Neos Marmaras, Greece, pp. 460-463, 20-22 June 1995.
7. A. G. Borş, I. Pitas, "Embedding parametric digital signatures in images," *European Signal Processing Conference, EUSIPCO'96*, Trieste, Italy, Sep. 10-13, pp. 1701-1704, 1996.
8. I. J. Cox, J. Kilian, T. Leighton, T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia, *IEEE Trans. on Image Processing,* vol. 6, no. 12, pp. 1673-1687, Dec. 1997.
9. A. Papoulis, *Probability and Statistics.* Prentice Hall, 1990.
10. S. A. Kassam, *Signal detection in Non-Gaussian Noise,* Springer-Verlag, 1988.
11. G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. on Consumer Electronics,* vol. 38, no. 1, pp. 18-34, Feb. 1992.
12. A. K. Jain,*Fundamentals of Digital Image Processing,* Prentice Hall, 1989.
13. A. C. Hung, "PVRG-JPEG CODEC 1.1," Stanford University, Technical Report, 1993.
14. A. V. Oppenheim, R. W. Schafer,*Discrete-Time Signal Processing,* Prentice-Hall, 1989.
15. B. Widrow, S.D. Stearns, *Adaptive Signal Processing.* Englewood Cliffs, NJ: Prentice Hall, 1985.
16. E. S. Chng, S. Chen, B. Mulgrew, "Gradient radial basis function networks for nonlinear and nonstationary time series prediction," *IEEE Trans. on Neural Networks,* vol. 7, no. 1, pp. 190-194, Jan 1996.

1. Introduction

Digital watermarks in the general context of TV broadcasting and cryptology were discussed in [1]. To avoid the unauthorized distribution of images or other multimedia property, various solutions have been proposed. Most of them make unobservable modifications to images, that can be detected afterwards . Such image changes are called watermarks. The watermark should not alter visibly the image and it should be robust to alterations which may be caused by various image processing techniques.

Algorithms proposed for watermarking have been reported in various papers [2 - 8]. They are either stochastic [3,5,6] or deterministic [4]. These algorithms are either in image intensity domain [4,6] or in frequency domain [3,5,8]. In [3] the middle range DCT frequency coefficients from the 8×8 pixel blocks are used for embedding a constraint. In [8] the signature is embedded in the DCT coefficients obtained after applying the DCT transform in the entire image.

An watermarking algorithm has two stages: watermarking casting and detection. By means of watermark casting a specific code assigned to the owner is embedded in the image. In the detection stage the algorithm identifies the given code. Signal detection theory is a well-established field with many applications [9,10]. A watermarked image can be processed by means of various image transformations and processing algorithms which may be able to destroy, intentionally or not, the digital watermark. Image compression is the most likely transformation that an image may undergo. The standard still image compression algorithm is JPEG [11]. JPEG is based on the minimization of the energy in the Discrete Cosine Transform (DCT) domain [12]. In the case of lossy compression, the image suffers information loss in the high frequency domain.

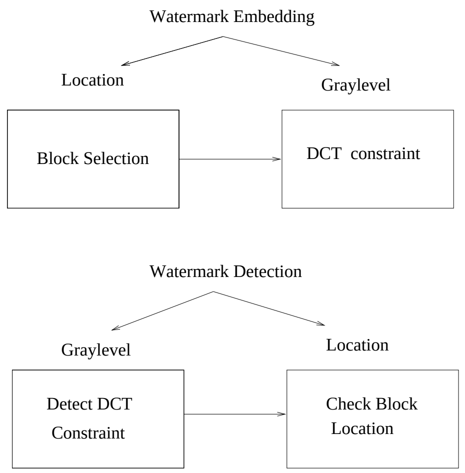


Figure 1. The processing blocks for watermark embedding and detection.

In the proposed watermark casting algorithm from [7], the image is partitioned in 8×8 pixel blocks similar to the JPEG algorithm [11]. The watermarking algorithm consists of two steps. The first step selects certain blocks according to a Gaussian network. In the selected blocks we modify DCT coefficients such that they fulfill a given constraint. The parameters of the Gaussian functions and of the imposed constraints on the DCT coefficients make up the watermark code. In the detection stage we first check for the DCT constraints and afterwards for the respective block location. The processing blocks for watermark embedding and detection are shown in Figure 1.

In Section 2 we propose a technique for chosing block sites. The DCT constraint embedding step is explained in Section 3. The detection stage is presented in Section 4. This analysis is necessary in order to determine the suitable watermark parameters such that each watermark is distinctly identified from all the others. The simulation results for applying the proposed algorithms in graylevel and color images are provided in Section 6. In Section 7 the conclusions of the present study are drawn.

2. Block site selection

The proposed watermark casting algorithm has two stages, each stage using a distinct part of the watermarking code. Let us denote a watermark by its parameter vector :

$$\mathbf{S}_k = [\mathbf{L}_k, \mathbf{C}_k] \tag{1}$$

where k is the owner identifier, $\mathbf{S}_k$ denotes the watermarking vector, $\mathbf{L}_k$ the component vector used for identifying the sites and $\mathbf{C}_k$ the parameter vector for embedding the DCT coefficient constraint.

We consider the image partitioned in block sites situated on a rectangular grid. The block size in this partition is 8×8, similar to that used by JPEG algorithm [11]. Each block has assigned a number denoting its location.

The block sites for embedding the watermark are selected based on a Gaussian network classifier [16]. Such a classifier, used in various time-series modeling applications [16] is emploied here for chosing certain block sites in the image. The differences between the locations of each two consecutively selected blocks are considered as input features for the classifier :

$$G(\mathbf{x}) = \sum_{i=1}^{L} \lambda_i \exp\left[-\sum_{j=1}^{N} \frac{(x_j - w_{i,j})^2}{r_{i,j}^2}\right] \tag{2}$$

where x_j is the difference between two consecutive chosen sites (measured by number of blocks) for the last $j = 1, \ldots, N$ sites, L is the number of Gaussian functions, $G()$ is the output of the network. $\lambda_i, w_{i,j}, r_{i,j}, N, L \in \mathbf{L}_k$ are the Gaussian parameters provided by the watermarking code. In the following we consider a row-column scanning order for chosing the appropriate blocks. The condition of selecting a block site is that the Gaussian network output is above a chosen threshold $\alpha \in (0, 1)$:

$$G(\mathbf{x}) > \alpha \tag{3}$$

where $G(\mathbf{x})$ is given in (2). This condition is fulfilled for a certain data range. The proces of selecting the block sites is illustrated in Figure 2.

The sites which fulfill (2) make up a time series. Let us assume that $\lambda_i = 1$ for $i = 1, \ldots, L$. We consider that a Gaussian function is activated if its output is above a certain threshold. Providing the distances between the chosen block sites, we consider the simplifying model where each Gaussian function is activated, in a certain order, only one at the time. In order to fulfill this condition, the parameters of the Gaussian network must form a circulant matrix :

$$w_{i,j} = w_{(i+1) \bmod L, j-1} \tag{4}$$

$$r_{i,j} = r_{(i+1) \bmod L, j-1} \tag{5}$$

for $i = 1, \ldots, L$ and $j = 2, \ldots, N$, where $\bmod L$ denotes the operation modulo L. The number of distinct network parameters is $2L$. After the L-th Gaussian unit is activated, the process continues with the first unit.

Let us consider known the first $N - 1$ block site locations $x_j = d_j$ for $j = 1, \ldots, N - 1$. In the signature embedding stage we have to find the location of the next block site that fulfills the condition (2). From (2) and (3), after assuming that only one Gaussian function is activated at the time and after applying the function ln, we obtain the following range of distances expressed in number of blocks :

$$\{w_{i,N} - T_{i,N} < x_N < w_{i,N} + T_{i,N},\} \text{ where} \tag{6}$$

$$T_{i,N} = r_{i,N} \sqrt{\ln\frac{1}{\alpha} - \sum_{j=1}^{N-1} \left(\frac{d_j - w_{i,j}}{r_{i,j}}\right)^2}. \tag{7}$$

This condition ensures that only one Gaussian function is used for selecting a certain block site. The distances between each two sites are calculated recursively, based on the site locations previously found.

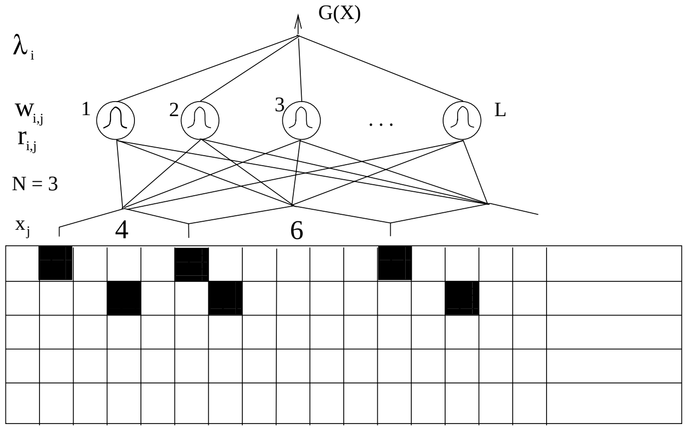


Figure 2. The selection of a block site using a Gaussian network where the selected blocks are black.

The distances $T_{i,N}$ given by (7) provide an interval from where to chose the next site. The sites that fulfill the condition (3) alternate with those which do not fulfill it. We impose the condition that only one Gaussian function is activated for a chosen block location. In order to fulfill this we impose :

$$\min\{w_{i,j-1}, w_{i+1,j}\} > T_{i,j} > 1, \text{ for } j = 2, \ldots, N\ ,\ i = 1, \ldots, L. \tag{8}$$

This condition ensures that only one Gaussian function is activated each time when selecting a block site.

Based on the watermarking parameters we can evaluate the bounds of the total number of selected blocks from the image. The number of blocks can be calculated by dividing the total number of blocks to the distance between two consecutive blocks :

$$\frac{MRL}{64\sum_{i=1}^{L}(w_{i,N} - T_{i,N})} > n > \frac{MRL}{64\sum_{i=1}^{L}(w_{i,N} + T_{i,N})}, \tag{9}$$

where $M \times R$ is the image size and $T_{i,N}$ are the distances recursively calculated in (7) for $i = 1, \ldots, L-1$.

The proposed algorithm for block site selection allows to embed the watermark locally, in a certain part of the image. The location of these regions depends on the watermark parameters and scanning order. Such a region contains all the image blocks located between the location of the block which activates the first Gaussian function and the block which activates the last (i.e., the L-th) function from the Gaussian network. We evaluate the bounds for the number of pixels contained in such a region :

$$64 \sum_{i=1}^{L} (w_{i,N} - T_{i,N}) < s_k < 64 \sum_{i=1}^{L} (w_{i,N} + T_{i,N}) \tag{10}$$

where s_k is the number of pixels from a signed region n is the total number of block sites, $T_{i,N}$ is provided in (7) and 64 is the number of block pixels. This region represents the area from the image from where the watermark can be identified without the need of knowing the rest of the image. After embedding the watermark in a certain region, the process is repeated until the entire image is watermarked.

3. The DCT coefficient constraint

In the second step DCT constraints are embedded in the selected blocks. Two distinct algorithms are considered for embedding the constraint in the DCT coefficients. The first algorithm embeds a linear constraint on selected DCT frequency coefficients and the second defines circular detection regions around the DCT watermark parameters. Both algorithms are designed in order to bring parametric modifications in the DCT coefficients that are resistant to the JPEG algorithm.

JPEG [11] is the standard image compression algorithm and it consists of three main processing blocks: Discrete Cosine Transform (DCT), quantizer and entropy coder. The quantization of the DCT coefficients aims to represent them with no greater precision than it is necessary to achieve the desired image quality. Thus, JPEG is a lossy compression scheme. In JPEG algorithm after shifting the image samples from unsigned integers [0, 255] to signed integers [-127 128], the forward DCT for 8×8 pixel blocks, is given by [11,13] :

$$F_j(u,v) = \frac{1}{4} C(u) C(v) \left[\sum_{k=0}^{7} \sum_{l=0}^{7} f_j(k,l) cos \frac{(2k+1)u\pi}{16} cos \frac{(2l+1)v\pi}{16} \right] \tag{11}$$

$$C(u), C(v) = \frac{1}{\sqrt{2}} \text{ for } u,v = 0 \text{ and } C(u), C(v) = 1 \text{ otherwise}$$

$f_j(k,l)$ are the image pixel values from the j-th block, and $F_j(u,v)$ are the transform coefficients.

It was suggested in [3] that certain modifications which are brought in the DCT coefficient domain, under certain conditions, may subsist to JPEG compression. The energy of the image is concentrated in the low frequency domain and the high frequency components represent only a small ratio from the total energy. The watermark embedded in the DCT coefficients should produce unnoticeable distortions in the image. The coefficients corresponding to the high frequency domain are cut off by the JPEG quantizer and they are not suitable to be used for embedding the given constraint. On the other hand, modifications in the low frequencies may cause visible alterations in the image. The most suitable frequencies to be used are those in the middle frequency range :

$$(u_i, v_i) \in \{(1,1), (1,2), (2,1), (3,1), (2,2), (1,3)\},\ i = 1, \ldots, K. \tag{12}$$

The number K of frequency pairs used for embedding the given constraint is provided by the watermark vector (1).

The linear constraint in the DCT coefficient domain is :

$$Y = \mathcal{F}\mathbf{Q} \tag{13}$$

where $\mathcal{F}$ is the vector of the modified DCT frequency coefficients, $\mathbf{Q} \in \mathbf{C}_k$ is the weighting vector provided by the watermark (1). The vector $\mathbf{Q}$ is characteristic to a watermark and contains the parameters corresponding to the selected DCT frequencies. In order not to produce significant distortions in the image these parameters must not be in a big ratio with respect to each other. This constraint is embedded in all the block sites which fulfill (2,3). The DCT frequency coefficients $\mathcal{F}$ are modified based on a least mean squares approach [15] such that we obtain a minimal distortion in the image, in order to fulfill (13). The relationship (13) is essentially a linear and time invariant system in the DCT domain [14].

In the second approach, circular regions are defined around certain DCT coefficients. Let us consider H, the number of circular regions. For a selected block site, we evaluate the Euclidean distance between its DCT coefficient vector and that of the watermark. The chosen DCT coefficients are changed to the value of the closest watermark parameter vector :

$$\|\mathcal{F} - \mathbf{Q}_k\|^2 = \min_{i=1}^{H} \|\mathcal{F} - \mathbf{Q}_i\|^2 \text{ then } \mathcal{F} = \mathbf{Q}_k, \tag{14}$$

where $\mathbf{Q}_i$ for $i = 1, \ldots, H$ is the set of coefficient vectors provided by the watermark and $\|.\|$ denotes the Euclidean distance. This approach is similar to the DCT vector quantization where the vector size is K.

The DCT coefficients are defined in a certain range [11] and the watermark parameters $\mathbf{Q}_i$ must be from the same range. The chosen watermark parameter which make up the signature code should be chosen such that after embedding the constraint (13) or (14) we do not have large distortions in the image. After modifying the DCT coefficient values, the image is reconstructed based on the inverse DCT transform:

$$g_j(k,l) = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} C(u)C(v)\mathcal{F}_j(u,v) cos \frac{(2k+1)u\pi}{16} cos \frac{(2l+1)v\pi}{16} \tag{15}$$

where $g_j(k,l)$ are the pixel values of the j-th block from the watermarked image.

Several block sites with the location given by (7) fulfill the same location condition (2). However, we chose only the block site which provides a minimal image distortion after embedding the DCT coefficient constraint. The image distortion is measured by the block mean absolute difference before and after embedding the watermark :

$$k = \arg \min_{j=b+w_{j,N}-T_{j,N}}^{b+w_{j,N}+T_{j,N}} \sum_{x=0}^{7} \sum_{y=0}^{7} |f_j(x,y) - g_j(x,y)|, \tag{16}$$

where b denotes the location of the last chosen block site by (2, 3), $T_{j,N}$ is provided in (7) and k is the selected block.

4. Watermark detection

In this stage we define the criteria to certify if a certain image is signed or not with a given watermark. Let us denote by x the random variable associated to the block site location, by y the random variable associated to the DCT constraint given by either (13) or (14) and by $P(x,y|g,\mathbf{S}_k)$ the probability of jointly estimating these random variables when providing the watermark code $\mathbf{S}_k$ and the signed image g. In our approach it is not necessary to have available the original image for detecting the watermark. The probability $P(x,y|g,\mathbf{S}_k)$ can be decomposed, based on the Bayes rule, in :

$$P(x,y|g,\mathbf{S}_k) \propto P(x|y,g,\mathbf{L}_k)P(y|g,\mathbf{C}_k), \tag{17}$$

where each of these probabilities depends only on one component of the watermark code (1). $P(y|g, \mathbf{C}_k)$ is the probability of detecting the DCT coefficient constraint and $P(x|y, g, \mathbf{L}_k)$ is the probability of detecting the location constraint, provided that the respective blocks already fulfill the first constraint, in the DCT coefficient domain. The probability $P(x, y)$ depends only on the image and has been neglected in (17).

When evaluating the probability of the DCT coefficient constraint $P(y|g, \mathbf{C}_k)$, the algorithm should take into account the likelihood that the signal is distorted. For example, after applying the JPEG compression algorithm on a watermarked image, the constraints (13) or (14) imposed onto the DCT coefficients are likely to be modified. The spread of the DCT coefficients with respect to the embedded values is large for small JPEG quality factors (large compression ratios). This accounts for a certain distortion in the image. Such distortions should be taken into account in the embedding stage.

The JPEG compression algorithm produces a spread in the DCT values of the watermarked image. In order to avoid that DCT constraints (13) or (14) are not satisfied after JPEG compression we define certain detection regions in the DCT domain denoted as D_y^k. The detection region $\mathcal{D}_y^k$ in the case of (13) is given by :

$$\mathcal{D}_{y,L}^k = \{y \in I\!R \mid |y - \mathcal{F}\mathbf{Q}| < D_L\}, \tag{18}$$

and when embedding (14) :

$$\mathcal{D}_{y,C}^k = \{y \in I\!R \mid \|\mathcal{F} - \mathbf{Q}_i\|^2 < D_C^2,\ i = 1, \ldots, K\}, \tag{19}$$

where the parameters D_L and D_C depend on the assumed level of watermark corruption due to compression. A large value for D_C or D_L should be assumed when the watermarked image is going to be heavily compressed (low quality factor). By assuming a certain level of watermark distortion, (18, 19) instead of (13, 14) are considered in the detection stage.

For minimizing the false alarm errors, the block sites which are not chosen by (2), must not fulfill the conditions (18) or (19). If, for certain values D_L or D_C, some of these blocks undesirably fulfill the given constraint, their DCT coefficients are modified forward the opposite. The detection region size parameters, D_L and D_C are associated to the image distortion produced by the watermark. The same parameter D_L or D_C embedded in images with different DCT frequency distributions is likely to cause different distortions.

The last step in the detection stage consists in the block location detection. The detection region for the location, denoted as $\mathcal{D}_x^k$, is obtained from (2,3) where we consider only the blocks which already fulfill (18) or (19).

A given site is considered as being signed when the probability of embedding both constraints, in the DCT domain and in the block site location is maximized. The statistical decision for the hypothesis H_0 for a certain signature $\mathbf{S}_k$ is taken based on the detection ratio defined as follows :

$$\frac{\int_{\mathcal{D}_x^k} \int_{\mathcal{D}_y^k} P(x, y|g, \mathbf{S}_k) dx dy}{\int_{\mathcal{D}_x^k} \int_{\mathcal{D}_y^k} q(x, y|g, \mathbf{S}_k) dx dy} \approx \frac{\sum_{i=1}^{p} s_i}{MR} > c, \tag{20}$$

where $q(x, y|\mathbf{S}_k)$ is a normalizing probability, s_i is the region size, measured in pixels where the watermark was successfully identified, p represents the number of regions and $c \in [0, 1]$. The bounds of the region size where the watermark can be successfully identified are given in (10). If the watermark is detected in the entire image then the above expression has a value equal to one. The expression (20) has a value zero in the case when the image was not signed with the respective watermark. If various distortions occur in the image than the expression (20) has an output in the interval $[0,1]$ depending by the level of corruption in image. Therefore, we consider a detection threshold of $c = \frac{1}{2}$ i.e., we can detect a given watermark if at least 50 % of the image candidate block sites satisfy both position and DCT constraints. A big advantage of the proposed scheme is that the digital watermark can be detected only in a part of the image depending on the scanning order as well. This property is very useful in the case of image sequences where the entire sequence is watermarked.

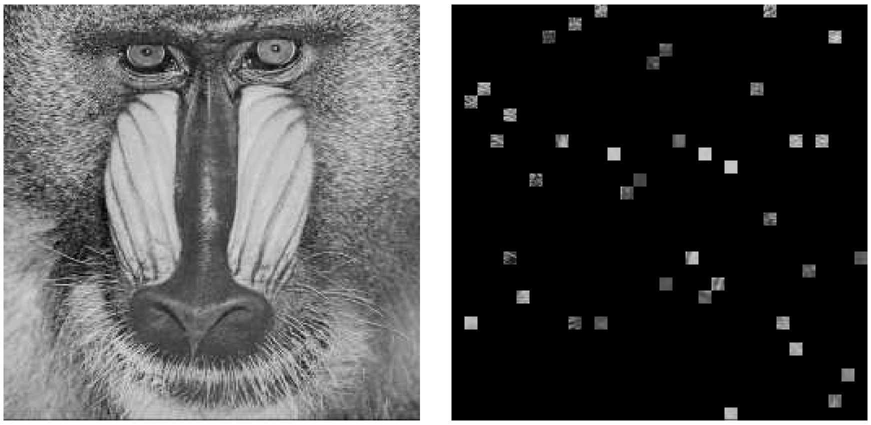
5. Simulation results

The proposed algorithms have been used to cast various watermarks on a large number of images. The original "Baboon" image is displayed in Figure 3 (a). The watermarked image based on the linear DCT constraint embedding algorithm (18) is shown in Figure 4 (a). In Figure 4 (b) the same image is watermarked according to (19). The block sites chosen to embed the DCT constraint are selected based on the procedure described in Section 2. The number of inputs and Gaussian functions for selecting the block sites are $N = 2$ and $L = 3$. The watermark characteristic DCT frequencies are F(2,1), F(1,2), and F(2,2). In order to measure the distortion produced in the image by the watermark we use the signal-to-noise ratio (SNR) between the watermarked and original image, where SNR expressed in dB is defined as :

$$SNR = 10 \log \frac{\sum_{i=0}^{M} \sum_{j=0}^{R} f(i,j)^2}{\sum_{i=0}^{M} \sum_{j=0}^{R} (f(i,j) - g(i,j))^2}, \tag{21}$$

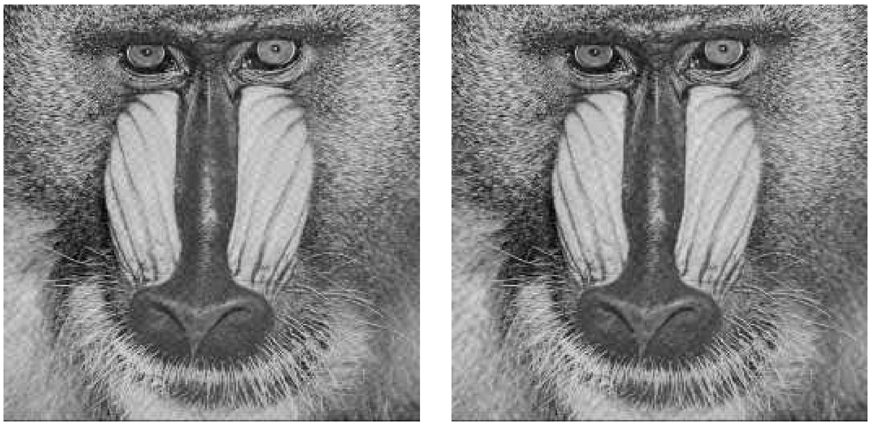
where f, g are the original and the watermarked image. The watermarked images from Figures 4 (a) and (b) have a SNR of 31.1 dB and 31.3 dB when compared to the original image. In order to find the watermark resistance to JPEG, we compress the watermarked images at various compression ratios and we detect the watermark in the compressed images according to (20). For the images represented in Figures 4 (a) and (b) the watermark was correctly identified after they were compressed at 13:1 and 18:1, respectively. The block sites that were chosen to embed the DCT coefficient constraint are displayed in Figure 3 (b). No periodicity with respect to their location in the image can be visually identified. The watermarked images shown in Figures 4 (a) and (b) contain some textured areas, slightly visible.

In Figures 5 (a) and (b), watermark detection curves when the watermark has been embedded based on (18) and on (19) are displayed. The SNR ratio provided near each curve denotes the embedding level D_C or D_L for the respective watermark. It is clear from these plots that the watermark embedded in an image based on a certain embedding level is able to resist up to a certain compression ratio. For example a watermarked image having high SNR of about 38dB (i.e., the signature is almost invisible), can resist to compression ratios up to 8:1 for the scheme given by (18) and up to 11:1 for the scheme given by (19) as it can be seen in Figures 5 (a) and (b). The algorithm which defines circular detection regions in the DCT coefficient domain (19) provides better detection capabilities when compared to the linear type DCT coefficient constraint algorithm (18). If the watermarks are not designed to resist at big compression ratios, the image distortions will not be significant. The image quality degrades if the watermarks are intended to resist at higher compression ratios. However, in these cases, after compressing the image, the distortions introduced by JPEG are significant as well.



(a) (b)

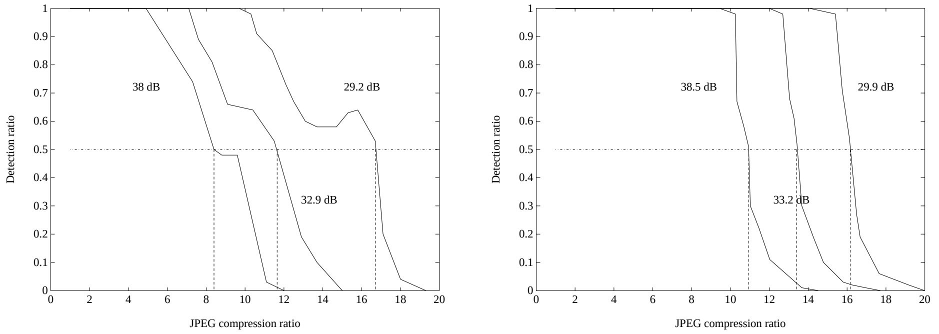
Figure 3. Watermarking the Baboon image: (a) Original image; (b) chosen image blocks for embedding the watermark



(a) (b)

Figure 4. Watermarking results: (a) when the DCT coefficient constraint is linear; (b) when embedding circular detection regions in the DCT domain.

In the case of color images the watermark is applied only in the luminance component (Y component in the YUV color system). The original color image "Lenna" is shown in Figure 5 (a). The image signed by the algorithm presented in (18), after it was compressed by JPEG with a compression ratio of 23:1, is presented in Figure 5 (b). In Figure 5 (c) the image signed by the algorithm based on (19) is displayed after it was compressed 22:1. The SNRs of these two images are 37.5 dB. The plots representing the robustness of these watermarks to JPEG compression are shown in Figures 5 (a), (b) for embedding the constraints (18) and (14), respectively. The watermarks resistance to color system transformation was successfully tested by transforming the signed image from the YUV color system to RGB and back to YUV.



(a) (b)

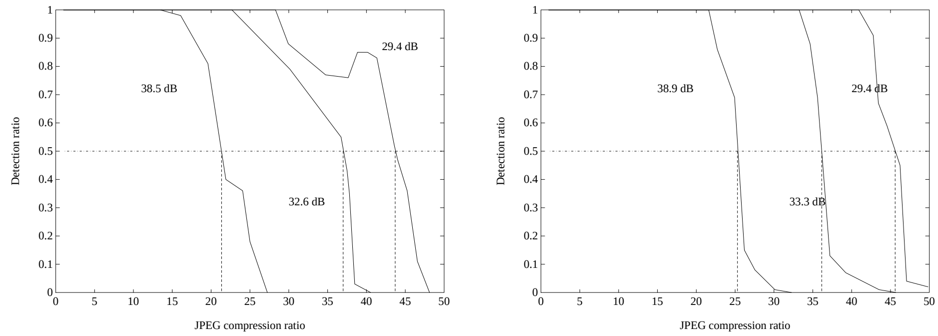
Figure 5. Watermark detection results after JPEG compression; (a) linear DCT constraint; (b) circular detection regions.

The watermarks cast by the proposed algorithms were tested at digital filtering as well. The plot representing the watermark detection results after filtering with a 3×3 window median filter is shown in Figure 8 (a) and with a 3×3 moving average filter in Figure 8 (b). The results provided by the algorithm based on linear constraints are marked with a dashed line and the results provided by the algorithm based on circular detection regions are marked with a continuous line. The watermarks embedded by the circular constraint algorithm (19) are always resistant to 3×3 median filtering. The watermark is resistant to 3×3 moving average filtering if it is rather "strong," i.e., it has a big watermark embedding level corresponding to $SNR < 23$dB for the linear constraint and to $SNR < 31$dB for a circular detection region.

All these results show that the proposed algorithms are able to cast watermarks which are resistant to JPEG compression and to representative filtering algorithms. The watermarks are detected only from the watermarked image and the watermark code respectively without any need to have the original image. The algorithm which embeds circular DCT detection regions (19) provides better results when compared to the algorithm which embeds a linear constraint in the DCT coefficients (18). The watermarks embedded by the proposed algorithm are also robust to certain image cropping or other distortions as extracting lines from the image. However, other geometrical distortions as rotation or zooming must be recovered before the watermark detection. The watermark can be applied recursively into an image sequence and afterwards identified separately in each frame.

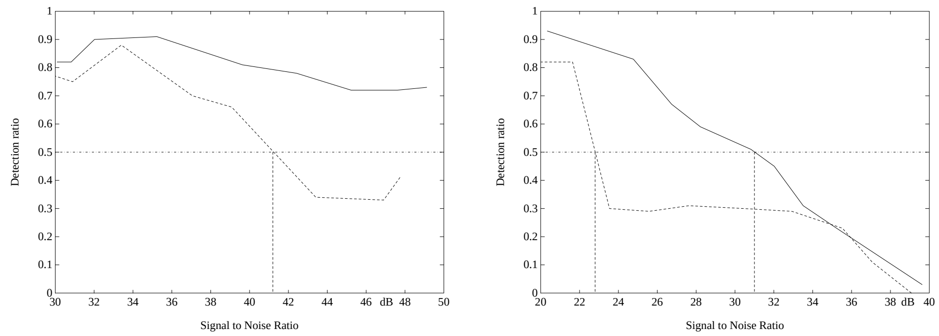
(a) (b) (c)

Figure 6. Compressed watermarked color images; (a) original color Lenna image; (b) image watermarked based on the linear constraint in the DCT coefficients; (c) image watermarked based on the circular detection regions in the DCT coefficients.



(a) (b)

Figure 7. Watermark detection results in the case of color images; (a) linear constraint in the DCT coefficients; (b) circular detection regions in the DCT coefficients.



(a) (b)

Figure 8. Watermark detection results after filtering; (a) median filter; (b) moving average filter.

6. Conclusions

A watermarking algorithm has two stages: embedding and detection. The watermarks generated by the watermark embedding algorithm should be robust to various image processing algorithms especially to JPEG compression. The algorithm proposed in this study has two processing steps : the selection of certain block sites in the image and the DCT constraint embedding. The selection of the block sites is done by a Gaussian network classifier. We suggest two different methods for embedding the constraints in the DCT coefficients. The first method embeds a linear constraint in certain selected DCT coefficients and the second method defines circular regions in the DCT coefficient space. The simulation results suggest that the proposed algorithms generate watermarks which are able to resist at certain JPEG compression ratios and to filtering.