SIViP
DOI 10.1007/s11760-008-0102-3

ORIGINAL PAPER

Image-adaptive watermarking using 2D chirps

**Yimin Zhang · Bijan G. Mobasseri ·
Behzad M. Dogahe · Moeness G. Amin**

Received: 15 January 2008 / Revised: 5 December 2008 / Accepted: 15 December 2008

Abstract Linear chirps, a special case of polynomial phase exponentials, have recently been proposed for digital watermarking. In this work, we propose a known-host-state methodology for designing image watermarks that are robust to compression. We use a two-dimensional frequency-modulated chirp as a spreading function in a block-based spatial watermarking scheme. In each block, the chirp is used to embed binary phase information. Chirp parameters allow for spectral shaping of the watermark to match host content. Since host state is known to the embedder, it is possible to tune the chirp for optimum performance, particularly against compression. In contrast to existing chirp watermarking where only a single watermark is generally embedded, the proposed block chirp watermarking allows for a much higher payload. Detection is done using chirp transform subject to key exchange for security. We show that the proposed method significantly outperforms non-adaptive watermarking across all compression factors under variety of attacks.

Keywords Digital watermarking · Time–frequency · Chirp · Frequency-modulated function · Compression · Attack

1 Introduction

Digital watermarking is the process of securely embedding invisible signatures within a cover media with no visual impact. A robust watermark should always remain present once it is applied to the original digital media. Such a watermark can be used in a number of applications, including copyright protection, fingerprinting, broadcast monitoring, data authentication, medical safety and data hiding [1].

The use of chirps, a linearly frequency-modulated (FM) waveform long used in radar and many other signal processing applications [2,3], for digital watermarking was introduced by Stankovic et al. [4]. In their work, a chirp watermark is added to the entire image. The energy-concentrating property of Radon–Wigner transform is then used to establish the presence of the watermark by peak-searching. This algorithm is best suited to copyright and ownership verification applications, where a binary decision is sufficient to establish the presence or absence of the watermark. For high payload applications, however, this approach is not effective. Use of the chirp is part of a broader approach to watermarking using time–frequency representations [5–9]. For example, [5] and [8] utilize bilinear time–frequency transformations and the watermark is embedded in the joint time–frequency domain. Watermarking in time–frequency domain has also been used in fragile watermarking applications [9]. In a more recent work, the watermark is actually the slope of the embedded chirp in time–frequency plane [10]. Samples taken from the chirp are binarized and spread by cyclically shifted copies of a PN sequence. The resulting watermark vector is added to block DCT coefficients subject to Watson's perceptual metric. Chirp samples are detected by a non-blind correlation receiver and transferred to time–frequency representation using Wigner–Ville distribution. Chirp initial frequency and slope are then estimated using Hough–Radon transform.

This work was supported by ONR under Grant No. N00014-04-1-0630. Parts of this work were presented at SPIE Annual Conference, Denver, CO, August 2004, and IEEE International Conference on Acoustics, Speech, and Signal Processing, Philadelphia, PA, March 2005.

Y. Zhang · B. G. Mobasseri (✉) · B. M. Dogahe · M. G. Amin
Center for Advanced Communications, Villanova University,
800 Lancaster Ave., Villanova, PA 19085, USA
e-mail: bijan.mobasseri@villanova.edu

Published online: 08 January 2009

Since chirp's signature in Hough-Radon transform domain is a straight line, it can still be reliably detected in the presence of watermark bit errors, up to 20% according to [10]. The algorithm can embed one watermark per image unless the image can accommodate long PN sequences. This approach improves upon [11] where watermark bits are spread out by a PN sequence. Linear chirps have also been used in audio watermarking [12] using the same general principle in [10]. In [13] chirps are also used for watermark embedding but detection is performed using discrete polynomial-phase transform (DPT). DPT when applied to a single tone polynomial phase signal produces a spectral line. By successive phase unwrapping operation coefficients of phase polynomials are systematically estimated. The algorithm embeds one chirp/watermark in the image.

While a single bit of watermark may suffice in some applications, higher payload of the watermark is often desirable for a variety of purposes, such as the protection of intellectual property rights, medical data embedding and data hiding [1]. A watermark for such purposes should also be robust against different types of attacks. For example, image compression is one of the commonly encountered attacks. Although not directly exploited in this paper, the availability of higher payload also makes it possible to incorporate error correction coding for improved robustness of the watermark.

In this work, we propose a methodology that exploits the knowledge of known state for designing image watermarks with high payload and robustness to compression. We use a two-dimensional (2D) frequency-modulated chirp as a spreading function in a block-based spatial watermarking scheme. In works cited above, the chirp is used because of its linear signature in time–frequency plane. We are using the chirp for different reasons. Chirp is a tunable function and can be spectrally shaped by choosing its initial frequency and chirp rate. This property allows for the design of an image-adaptive watermark for added robustness. For example, the space-varying frequency of the chirp makes the watermark robust to stationary filtering attacks. This is in contrast to spread spectrum watermarking where the watermark is spread out into wideband noise by a PN sequence and is statistically and spectrally unrelated to the host signal [14]. In this model each bit of the watermark array is modulated by a 2D PN sequence and added to non-overlapping sets of image pixels driven by a density metric. Although PN sequences provide robustness against malicious attacks through processing gain, they are inflexible in terms of spectral shaping. The ability to spectrally shape the watermark by choosing the appropriate chirp allows for the design of spreading functions that have minimum overlap with the image data. In fact, our approach can potentially achieve zero bit error rate (BER) by exploiting knowledge of the host signal at the encoder. Chirps can be chosen to provide the desired power distribution over frequency. Polynomial phase watermark energy is highly localized, leading to improved detection through matched filtering and linear chirp transforms. This in turn allows for reduction of watermark energy. In addition, there has been considerable work in time–frequency processing techniques in the areas of speech, communications, fault structures, automation, biomedicine, radar and sonar [15–17]. These techniques provide easily accessible information about the signal spectral localization over short time periods and spatial segments.

This paper is organized as follows: Sect. 2 introduces the embedding of watermark using FM spreading functions. The watermark detection and recovery are discussed in Sect. 3. Section 4 considers the design of watermark spreading function based on 2D chirp signals. Several practical issues are discussed, including the chirp parameter selection and adaptive power allocation. Section 5 investigates the impact of JPEG image compression and the design and optimization of the chirp parameters. Computer simulation examples are provided in Sect. 6. The robustness of the proposed watermark against different attacks is examined in Sect. 7.

2 Watermark embedding

The problem considered is embedding a digital watermark containing N-bit information in a gray-scale image. The image is partitioned into non-overlapping blocks whose sizes depend on the picture size and the amount information to hide. If N_p is the number of bits that a block can host, then $\lceil N/N_p \rceil$ blocks are needed, where $\lceil x \rceil$ denotes the minimum integer equal to or larger than x.

As an example, we consider in this paper a 32 × 32 binary seal, depicted in Fig. 1b, to be embedded in the 512 × 512 gray-scale picture of *Lena*, which is shown in Fig. 1a. If the watermark is embedded through binary phase modulations, each block hosts one bit of information. To embed the entire watermark payload, the image is partitioned into

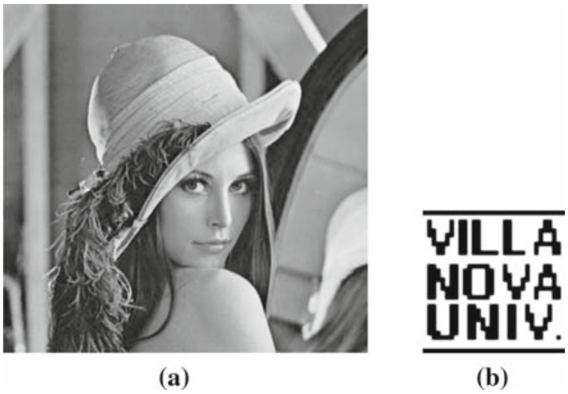


Fig. 1 Original image and binary seal: **a** gray-scale image of Lena (512 × 512) and **b** binary seal (32 × 32)

32×32 blocks, each of size 16×16 pixels. In each block (m, n), $m, n = 0, \ldots, 31$, the watermarked image $G(x, y)$ is expressed as

$$\begin{aligned} G(m,n,x,y,k) &= I(m,n,x,y) + \mathcal{Q}\left\{k\mathrm{Re}[s(m,n)\tilde{W}(x,y,\Theta_0)]\right\} \\ &= I(m,n,x,y) + \mathcal{Q}\left[ks(m,n)W(x,y,\Theta_0)\right] \end{aligned} \tag{1}$$

where $I(m, n, x, y)$ is the original image at block (m, n), extending over the spatial axes, x and y, with $(x,\ y) \in [0, \ldots, 15]$. In the above equation, $\tilde{W}(x, y, \Theta_0)$ is the complex 2D FM spreading function, which is defined by a set of parameters Θ_0, and $W(x, y, \Theta_0) = \mathrm{Re}[\tilde{W}(x, y, \Theta_0)]$. As we explain later, for 2D chirp waveforms, $\Theta_0 {=} (\beta_x, \beta_y, f_x, f_y)$, where β_x and β_y are the chirp rates in the x and y directions, and f_x and f_y are the corresponding initial frequencies. When the 2D chirp is symmetric in the x and y directions, i.e., $\beta_x = \beta_y = \beta_0$, $f_x = f_y = f_0$, Θ_0 reduces to (β_0, f_0). In addition, $s(m, n)$ represents the information to be mapped into the 2D spreading function in block (m, n). When binary phase data modulation is used, $s(m, n)$ takes the value of $+1$ or -1, corresponding to either 0 (black) or 1 (white) of the seal pixels. The parameter k is introduced to control the watermark strength and, subsequently, the image-to-watermark ratio, which is often referred to as the peak signal-to-noise ratio (PSNR). Moreover, $\mathrm{Re}[\cdot]$ denotes the real-part operator, emphasizing the fact that while the original 2D FM spreading function is complex, the hidden information in the image is real. $\mathcal{Q}[x] = \lfloor x + 0.5 \rfloor$ is rounding operation, where $\lfloor \cdot \rfloor$ stands for rounding down to the nearest integer.

3 Watermark detection and recovery

3.1 Watermark detection and parameter estimation

We consider blind decoding of the watermarked image, that is, the unmarked image is not used in the detection. When the parameter set Θ_0 that defines the 2D FM spreading function is not available at the detector, it must be estimated prior to detection. On the other hand, when Θ_0 is known at the detector, parameter estimation can be skipped.

The detector estimates $\hat{\Theta}_0$ by maximizing the following criterion,

$$\hat{\Theta}_0 = \arg\max_{\Theta} |C(m,n,\Theta,k)|, \tag{2}$$

where

$$\begin{aligned} C(m,n,\Theta,k) &= \sum_{x=0}^{T-1}\sum_{y=0}^{T-1} G(m,n,x,y,k)W(x,y,\Theta) \\ &= \sum_{x=0}^{T-1}\sum_{y=0}^{T-1} I(m,n,x,y)W(x,y,\Theta) \\ &\quad + \sum_{x=0}^{T-1}\sum_{y=0}^{T-1} \mathcal{Q}[ks(m,n)W(x,y,\Theta_0)] \\ &\quad \times W(x,y,\Theta) \\ &= C_I(m,n,\Theta) + C_W(m,n,\Theta,k) \end{aligned} \tag{3}$$

is the 2D chirp transform of the watermarked image using the 2D FM spreading functions defined over all possible values of Θ. In (3),

$$C_I(m,n,\Theta) = \sum_{x=0}^{T-1}\sum_{y=0}^{T-1} I(m,n,x,y)W(x,y,\Theta)$$

is the chirp transform output due to the original image, whereas

$$\begin{aligned} C_W&(m,n,\Theta,k) \\ &= \sum_{x=0}^{T-1}\sum_{y=0}^{T-1} \mathcal{Q}[ks(m,n)W(x,y,\Theta_0)]W(x,y,\Theta) \end{aligned}$$

is the chirp transform output corresponding to the watermark. The latter achieves its maximum when $\Theta = \Theta_0$. When the watermark has enough energy such that $|C_W(m, n, \Theta_0, k)| > |C_I(m, n, \Theta_0)|$, then the spreading function parameters Θ_0 may be reliably estimated by locating the peak of $C(m, n, \Theta)$ in the Θ-domain.

3.2 Watermark recovery

When the watermark is detected and spreading function parameters Θ_0 are known or reliably estimated, the watermark information at block (m, n) can be recovered from the phase information of the matched filter output, i.e., $C(m, n, \Theta_0)$. In particular, when the binary phase modulation is used, the embedded information is estimated as

$$\hat{s}(m,n) = \begin{cases} 1, & \text{if } C(m,n,\Theta_0,k) \geq 0 \\ 0, & \text{if } C(m,n,\Theta_0,k) < 0. \end{cases} \tag{4}$$

The problem with any peak finding algorithm such as (2) is that it is not secure. Other decoders can recover Θ_0 just as well. The solution is to spread the watermark energy across image blocks to minimize watermark detection probability. The problem is that the embedded information may not have enough strength to warrant reliable parameter estimation of the spreading function in each individual block. In this case, we resort to using the sum of the transform values over all blocks. Define

$$\tilde{C}(\Theta,k) = \sum_m\sum_n |C(m,n,\Theta,k)|, \tag{5}$$

$$\tilde{C}_I(\Theta) = \sum_m\sum_n |C_I(m,n,\Theta)|, \tag{6}$$

$$\tilde{C}_W(\Theta,k) = \sum_m\sum_n |C_W(m,n,\Theta,k)|. \tag{7}$$

When the hidden information is embedded such that

$$\tilde{C}_W(\Theta_0, k) \gg \tilde{C}_I(\Theta_0), \tag{8}$$

then the presence of the watermark can be detected and reliable estimation of $\hat{\Theta}_0$ can be achieved by maximizing the following criterion,

$$\hat{\Theta}_0 = \arg\max_{\Theta} \tilde{C}(\Theta, k). \tag{9}$$

To enhance watermark detection security further, we use different chirp rates $\beta_0(m, n)$ and initial frequencies $f_0(m, n)$ depending on the block index (m, n), where $0 \le m \le M-1$ and $0 \le n \le N-1$, and M and N denote the number of blocks in the x and y directions, respectively. The selection of chirp rates and initial frequencies is based on BER analysis which leads to the selection of the appropriate (β, f) region. We have observed that for natural images there is an overlap of the desirable (β, f) regions. A strategy may be designed so that the propagation of secret key to the decoder is not necessary. As such, a (β_0, f_0) region is selected such that different images are likely to have acceptable BER values. Because each image has its respective (β_0, f_0) region that produces a low BER value, the common (β_0, f_0) region should be chosen as an overlapped set shared by a large number of images. Otherwise, the boundary of this region needs to be communicated to the decoder.

Blocks are watermarked in raster scan, left to right, top to bottom. If each block is to receive a different chirp, there are $M \times N$ pairs of (β_0, f_0) available for assignment. However, different blocks could also use the same (β_0, f_0). The assignment of different chirp parameters to image blocks is governed by a pointer derived from a pseudorandom sequence. Let V pairs of (β_0, f_0) be used in watermarking $M \times N$ image blocks. Define a pointer v such that,

$$v = PN[(mN + n) \bmod V, \text{ para}] \tag{10}$$

where PN represents a predefined PN mapping, $v \in \{0, 1, \ldots, V-1\}$ and para is a set of parameters, often the seed, used in the PN generator formula. One way to define v is to use a random permutation of integers $\{0 : V-1\}$ under a key. Once a secure pointer is defined, the (m, n)th image block is watermarked by a chirp whose parameters are given by $(\beta_0(m, n), f_0(m, n)) = (\beta_0(v), f_0(v))$. This is a pseudorandom assignment of chirp parameters to image blocks selected from an appropriate region of (β, f) plane.

Watermark detection is blind, i.e. the unwatermarked image is not required. The detector only needs the seed of the PN sequence and the region of (β, f) plane used for watermarking. The decoder then reconstructs v and uses it as a pointer in (β, f) plane to recover the V pairs of (β_0, f_0). Using the correct key will cause a peak in (5) signaling that the image carries a watermark. If the key is not known, the decoder must do an exhaustive search over all anticipated combinations of (β_0, f_0). If V distinct pairs are used without repetition, then the decoder needs to search over $V!$ combinations; and this assumes that the V pairs are known in the first place. Once it is established that the image is carrying a watermark, the watermark can be extracted from the (m, n) block using $(\beta_0(m, n), f_0(m, n))$. All the procedures discussed in the previous sections then apply with the understanding that $(\beta_0(m, n), f_0(m, n))$ is now used in place of (β_0, f_0).

As it becomes obvious from the above discussion, the use of pseudorandom $(\beta_0(m, n), f_0(m, n))$ parameters has two advantages. First, it distributes the watermark energy over a region in the (β, f) domain and thus reduces the energy at any specific (β_0, f_0) point. As a result, the detectability of the watermark by unauthorized users is considerably more difficult. Second, even when the existence of the watermark is suspected, the watermark information cannot be reliably recovered without the secret key.

4 Spreading function design

4.1 2D chirp spreading functions

In this section, we consider the design of watermark spreading functions. A 2D complex chirp function is used as a simple example of 2D FM spreading function, given by [15,4]

$$W(x, y, \beta_x, \beta_y, f_x, f_y) = e^{j\pi(\beta_x x^2 + \beta_y y^2) + j2\pi(f_x x + f_y y)}, \tag{11}$$

where β_x and β_y are the chirp rates along the x and y axes, and f_x and f_y are the respective initial frequencies. These four variables form the spreading function parameters, i.e., $\Theta_0 = (\beta_x, \beta_y, f_x, f_y)$. For notational simplicity and without loss of generality, we consider symmetric cases and denote $\beta_x = \beta_y = \beta_0$, and $f_x = f_y = f_0$. In this case, Θ_0 simplifies to $\Theta_0 = (\beta_0, f_0)$, and the 2D chirp spreading function becomes

$$W(x, y, \beta_0, f_0) = e^{j\pi\beta_0(x^2+y^2) + j2\pi f_0(x+y)}. \tag{12}$$

The variables x and y take values from $[0, \ldots, T-1]$, forming $T \times T$ blocks. Therefore, the instantaneous frequency in (12) ranges from f_0 to $\beta_0(T-1) + f_0$.

The spectrum of the 2D chirp spreading function is important in determining the watermark detectability and robustness. In designing the chirp spreading function, the image and the chirp spreading function need to be weakly correlated, and the chirp must be robust against image compression. For this purpose, the high-frequency band is first excluded from consideration and then the parameters are optimized by choosing those where the image spectrum is low.

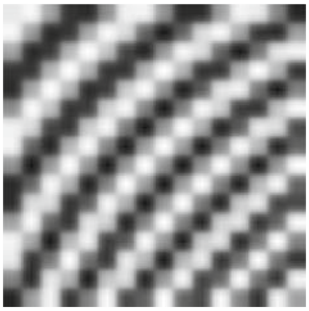


Fig. 2 Quantized 2D chirp spreading function

Figure 2 illustrates the quantized waveform of a 2D chirp function. The chirp transform spectrum, using 2D chirp spreading function $W(x, y, \beta, f)$, of the first (upper-left corner) block of the original *Lena* as well as that of $W(x, y, \beta_0, f_0)$ are shown in Fig. 3. The PSNR is 40 dB and $(\beta_0, f_0) = (0.011, 0.11)$. There are numerous image quality metrics [18]. PSNR is a traditional quality measure and is a readily computable quantity. There is no one figure for the PSNR to guarantee transparency but 35–40 dB has been often used [19]. For example, at 20:1 JPEG compression, which results in acceptable quality, the PSNR is about 33 dB [20]. Since the higher the PSNR the more difficult it is to detect the watermark, PSNR = 40 dB adopted here is a conservative choice. It is clear that the image is lowpass with its high power located at small values of β and f. On the other hand, the watermark exhibits a peak value at (β_0, f_0) and other high values away from the origin.

4.2 Chirp parameter selection

For the binary phase modulation, the error probability, i.e., embedding information s and deciding erroneously in favor of $\hat{s} \neq s$, is given by

$$P_e = P(\hat{s} \neq s) = P(s = -1)P(\hat{s} = +1|s = -1) + P(s = +1)P(\hat{s} = -1|s = +1). \quad (13)$$

The error probability is computed as the total number of error bits divided by the total number of information bits. It is noted that, although we used the stochastic term probability here for convenience, the image information over different blocks is determinant and is known at the embedder.

The probability is evaluated for all blocks. To ensure correct detection at the (m, n)th block, the following condition should be satisfied:

$$C_W(m, n, \beta_0, f_0, k) \overset{>}{\underset{<}{}} - C_I(m, n, \beta_0, f_0), \quad \text{if } s(m, n) \overset{>}{\underset{<}{}} 0. \quad (14)$$

The chirp transform of the embedded spreading function at block (m, n), $C_W(m, n, \beta_0, f_0, k)$, takes the form of

$$C_W(m, n, \beta_0, f_0, k) = s(m, n)H(\beta_0, f_0, k), \quad (15)$$

where $H(k) = |C_W(m, n, \beta_0, f_0, k)|$ is the magnitude which depends on k but is independent of m and n. Note that, for a given value of k, $H(k)$ does have some weak dependence on β_0 and f_0 because of the rounding operation.

Substituting (15) in (14) yields

$$H(k) > -s(m, n)C_I(m, n, \beta_0, f_0). \quad (16)$$

Therefore, the error probability in (13) becomes

$$P_e = P\left[H(k) < -s(m, n)C_I(m, n, \beta_0, f_0)\right]. \quad (17)$$

Figure 4 illustrates $C_I(m, n, \beta_0, f_0)$ for four hypothetical blocks carrying watermark information bits $\{-1, +1, +1, +1\}$. The first two bars results in erroneous decision because they exceed the marked threshold determined by the watermark energy. The next two bars result in correct decision since the magnitude of the chirp transform falls below the threshold.

The chirp parameter selection, in essence, is to find (β_0, f_0) such that, given an embedding power (or, equivalently, k), the error probability depicted in (17) is minimized. Note that it is in fact a known-host-state problem. Therefore, the optimum values of (β_0, f_0) can be selected by searching (β, f) such that the above error probability is minimized. When a fixed watermark power is applied in all the blocks, $H(k)$ is a constant, and the error probability can be readily determined by examining the value of $-s(m, n)C_I(m, n, \beta_0, f_0)$. Examples will be given in Sect. 6.

4.3 Adaptive chirp power allocation

The known-host-state method allows us to embed the watermark in such a way as to push the decision metric into the correct decision region. As we discussed earlier, to ensure correct detection at the (m, n)th block, condition (16) should be satisfied. Because both $s(m, n)$ and $C_I(m, n, \beta_0, f_0)$ are known to the embedder, we can choose different values of k at different blocks. In this case, (16) can be modified as

$$H(k(m, n)) > -s(m, n)C_I(m, n, \beta_0, f_0). \quad (18)$$

That is, at each block (m, n), the minimum value of $k(m, n)$ can be chosen to satisfy the above equation to maintain error-free detection. It is noted that when $s(m, n)$ and $C_I(m, n, \beta_0, f_0)$ have the same sign, the right-hand side of (18) becomes negative and the above requirement is satisfied irrespective of the value of $k(m, n)$. In this case, k can be set to zero, and no chirp is actually added to the image block. It is noted that, if blind detection is required without a priori chirp parameter information at a detector, the total watermark energy needs to

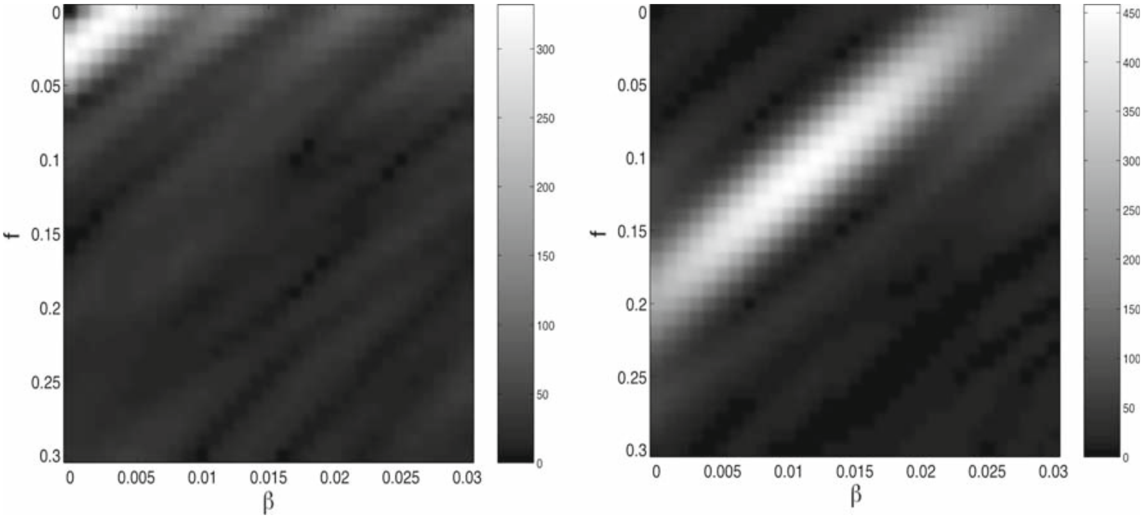


Fig. 3 Chirp transform spectra of a picture block $C_I(\beta, f)$ (*left*) and the 2D chirp spreading function $C_W(\beta, f)$ (*right*)

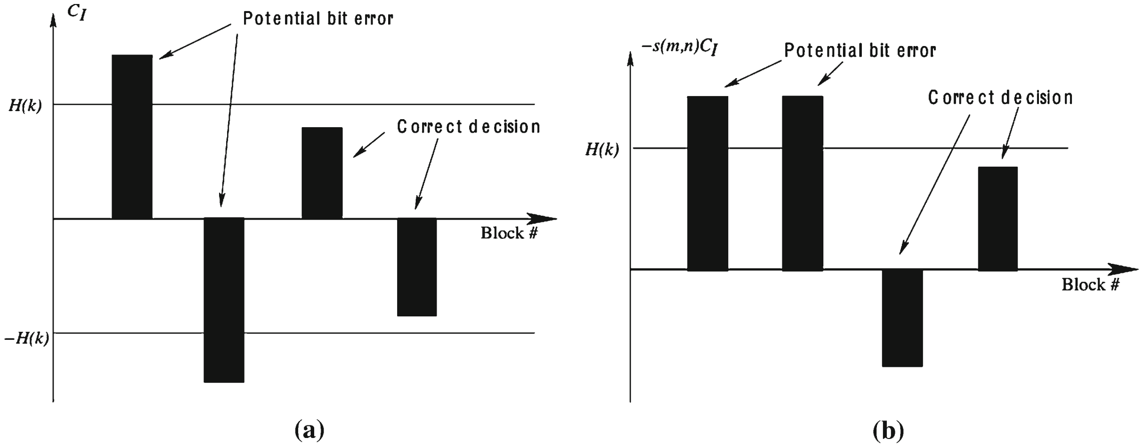


Fig. 4 Determining potential error detection from the relationship of $C_I(m, n, \beta_0, f_0)$ and $H(k)$, where four blocks carrying watermark bits $\{-1, +1, +1, +1\}$: **a** plot of $C_I(m, n, \beta_0, f_0)$ and **b** plot of $-s(m, n)C_I(m, n, \beta_0, f_0)$

be maintained such that detection and parameter estimation can be carried out successfully.

5 Chirp parameters optimization for JPEG compressed images

A common signal processing operation on images is JPEG compression. JPEG is designed to attenuate or remove high frequencies in the image by quantizing the DCT coefficients. The question to be answered is whether the added watermark will also be removed by the quantizer. If the watermark is removed from all DCT coefficients, the decoder will fail to recover the watermark. The watermark BER is affected by varying levels of compression, and more importantly, the choice of $\{\beta_0, f_0\}$ to make the watermark robust to compression.

Rewrite (1) for the underlying chirp signal case as

$$G(m,n,x,y,k) = I(m,n,x,y) + \mathcal{Q}[ks(m,n)W(x,y,\beta_0,f_0)]. \tag{19}$$

Baseline JPEG consists of four sequential steps: (a) block DCT; (b) quantization; (c) zigzag scan; and (d) entropy coding. The goal here is to spectrally shape the chirp to make it most robust to JPEG for a given quality factor Q.

Figure 5 shows how chirp energy distribution can be changed to counter JPEG quantization matrix. When $\beta_0 = 0$, the watermark is sinusoidal, and the energy is localized in the DCT domain. On the other hand, when $\beta_0 = 0.033$ which is relatively large, the energy is distributed in the DCT domain, particularly in the high frequency regions.

These figures illustrate how chirp's spectrum is modified by different choices of $\{\beta_0, f_0\}$. The distribution of DCT coefficients should be closely matched to the quantization

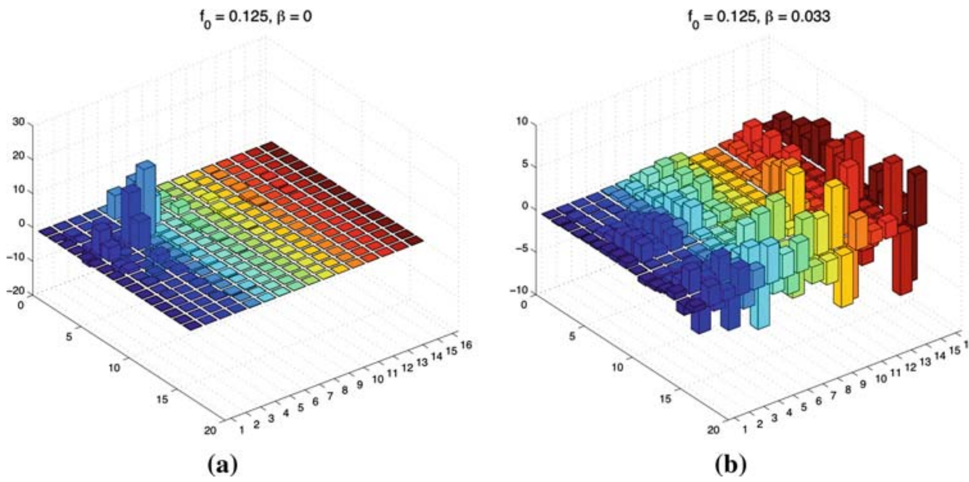


Fig. 5 Block 16 × 16 DCT of a chirp for different $\{\beta_0, f_0\}$: **a** $\beta_0 = 0$, $f_0 = 0.125$ and **b** $\beta_0 = 0.033$, $f_0 = 0.125$

matrix to produce the lowest BER. Since quantization matrix tends to compress higher frequency bands more aggressively, compression affects chirps with higher frequency contents more. The advantage of using a chirp versus a sinusoid is clearly demonstrated here. Sinusoid's energy is concentrated at specific frequency bands and can be easily removed by selective filtering or compression. In addition, there are virtually no degrees of freedom to spread the spectrum and optimize detection for varying JPEG Q factors.

The question of chirp survival after JPEG cannot be discerned solely by observing chirp DCT since the quantizer operates on the DCT of the image plus chirp and not the DCT coefficients individually. Denote $G(m', n', k)$ as the (m', n')th block of the watermarked image where each block is of size 8×8 to match the JPEG compression standard, and let $\mathcal{G}(m', n', k) = \text{dct}[G(m', n', k)]$. We also define $\mathcal{I}(m', n') = \text{dct}[I(m', n')]$ and $\mathcal{W}(m', n', \beta_0, f_0, k) = \text{dct}[\tilde{W}(m', n', \beta_0, f_0, k)]$ in a similar way, where $\tilde{W}(m', n', \beta_0, f_0, k)$ is the watermark defined at the (m', n')th block with chirp parameters (β_0, f_0). Then, at the (m', n')th block, the (i, j)th quantized DCT coefficient is given by

$$\mathcal{Q}\left(\frac{\mathcal{G}_{i,j}(m', n', k)}{q_{i,j}}\right) = \mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}} + \frac{\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)}{q_{i,j}}\right) \tag{20}$$

where $q_{i,j}$ is (i, j)th JPEG quantization coefficient, $\mathcal{I}_{i,j}(m', n')$ is the (i, j)th element of $\mathcal{I}(m', n')$, and $\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)$ is the (i, j)th element of $\mathcal{W}(m', n', \beta_0, f_0, k)$, $i, j = 0, \ldots, 7$. The decoder then performs an inverse quantization on (20) followed by inverse DCT to obtain $\hat{G}(m', n', k)$, i.e.,

$$\hat{G}(m', n', k) = \text{dct}^{-1}\left[\hat{\mathcal{G}}(m', n', k)\right], \tag{21}$$

where

$$\begin{aligned}\hat{\mathcal{G}}_{i,j}(m', n', k) &= q_{i,j} \cdot \mathcal{Q}\left(\frac{\mathcal{G}_{i,j}(m', n', k)}{q_{i,j}}\right) \\ &= q_{i,j} \cdot \mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}} + \frac{\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)}{q_{i,j}}\right).\end{aligned} \tag{22}$$

It is clear that the watermark will be eliminated unless the chirp has enough strength (i.e., k is sufficiently large) to survive compression and decompression cycle. In particular, quantization will remove the watermark if the following condition holds

$$\begin{aligned}&\mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}} + \frac{\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)}{q_{i,j}}\right) \\ &\quad = \mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}}\right)\end{aligned} \tag{23}$$

for all $i, j = 0, \ldots, 7$.

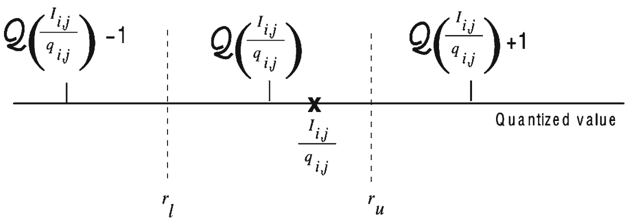
To have a closer look, define the upper and lower cross over boundaries by

$$r_u(m', n', i, j) = \mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}}\right) + 0.5, \tag{24}$$

and

$$\begin{aligned}r_l(m', n', i, j) &= \mathcal{Q}\left(\frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}}\right) - 0.5 \\ &= r_u(m', n', i, j) - 1.\end{aligned} \tag{25}$$

In Fig. 6, the "×" mark shows the quantized DCT term before rounding, i.e., $\mathcal{I}_{i,j}(m', n')/q_{i,j}$. The contribution of



$Q(\frac{I_{i,j}}{q_{i,j}})$ -1
$Q(\frac{I_{i,j}}{q_{i,j}})$
$Q(\frac{I_{i,j}}{q_{i,j}})$ +1
Quantized value
$\frac{I_{i,j}}{q_{i,j}}$
r_l
r_u

Fig. 6 Diagram showing how the watermark may be removed by the quantization step in JPEG compression

the watermark in this DCT coefficient will be removed if

$$r_l(m', n', i, j) < \frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}} + \frac{\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)}{q_{i,j}} < r_u(m', n', i, j), \tag{26}$$

or, equivalently,

$$r_l(m', n', i, j) - \frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}} < \frac{\mathcal{W}_{i,j}(m', n', \beta_0, f_0, k)}{q_{i,j}} < r_u(m', n', i, j) - \frac{\mathcal{I}_{i,j}(m', n')}{q_{i,j}}. \tag{27}$$

The watermark will be completely removed in a block in the process of JPEG compression when all the 64 coefficients satisfy the above condition.

6 Simulation results

In this section we provide experimental results on the performance of chirp watermarking and compare it with spatial and spectral spread spectrum watermarking. The cover image is 512 × 512 *Lena* shown in Fig. 1a and the watermark is a 32 × 32 binary seal depicted in Fig. 1b.

6.1 Chirp parameter selection

To understand the effect of different chirp parameters on the watermark detection performance, we first show in Fig. 7 the BER performance versus the chirp rate β_0, where the initial frequency is fixed at $f_0 = 0.2333$. It is clear that, when JPEG compression is not applied, a high value of β_0 tends to provide low BER performance. This is because the image is fundamentally lowpass. With moderate JPEG compression, however, the high-frequency bands will be suppressed and an optimum range of β_0 must therefore be defined.

The next example shows the sensitivity of $|C_I(m, n, \beta_0, f_0)|$, i.e., the magnitude of chirp transform of the original image, to the values of (β_0, f_0). Two different sets of chirp parameters, $(\beta_0, f_0) = (0.008, 0.08)$ and $(\beta_0, f_0) = (0.011, 0.11)$, are considered and compared. Figure 8 shows their histogram over 1024 blocks. The energy of the matched filter output of the watermark, corresponding to PSNR = 35, 40, and 45 dB, are depicted in the figures. For the chirp watermark $(\beta_0, f_0) = (0.008, 0.08)$, there are 25, 81, and 175 image blocks with a magnitude value of the chirp transform higher than that of the watermark at PSNR = 35, 40, and 45 dB, respectively. For the other chirp watermark, i.e., $(\beta_0, f_0) = (0.011, 0.11)$, there are 8, 30, and 76 blocks having a magnitude value higher than that of the watermark at the corresponding PSNR levels. Clearly, at the specific values of PSNR, $(\beta_0, f_0) = (0.011, 0.11)$ is preferable, since there are fewer blocks which could cause detection errors.

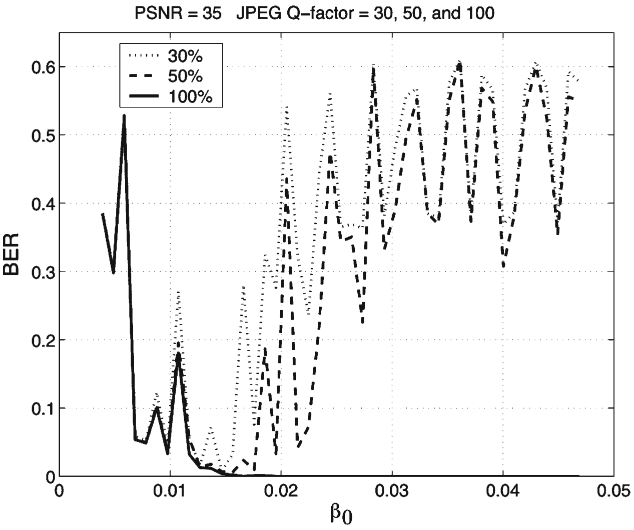


Fig. 7 BER versus chirp rate (PSNR = 35 dB)

6.2 Adaptive watermark power allocation

It is evident from the previous example that, the required watermark level to ensure correct detection is different for different image blocks. To find the minimum watermark energy for each block, we plot in Fig. 9a the magnitude value of the chirp transform of the original image in a sequential order. The dashed lines show the chirp transform of the watermark, $C_W(m, n, \beta_0, f_0, k)$, at PSNR = 40 dB. As we discussed in Sect. 4.2, $C_W(m, n, \beta_0, f_0, k)$ takes value of either $H(k) = |C_W(m, n, \beta_0, f_0, k)|$ or $-H(k)$, depending on the value of the watermark bit $s(m, n)$. At those blocks where the magnitude of the image contribution exceeds the watermark output, there is a possibility that the watermark information bit is wrongly decoded. However, whether it occurs or not depends on the sign of the embedded information.

To incorporate the watermark information, therefore, we plot in Fig. 9b the result of $-s(m, n)C_I(m, n, \beta_0, f_0)$. A decision error will occur in each of the blocks if $-s(m, n) C_I(m, n, \beta_0, f_0) > H(k(m, n))$. In other words, we can choose a different value of $k(m, n)$ at different block to adjust the watermark energy such that $H(k(m, n))$ merely exceeds $-s(m, n)C_I(m, n, \beta_0, f_0)$. As such, the watermark energy is minimized while low error-rate watermark embedding is

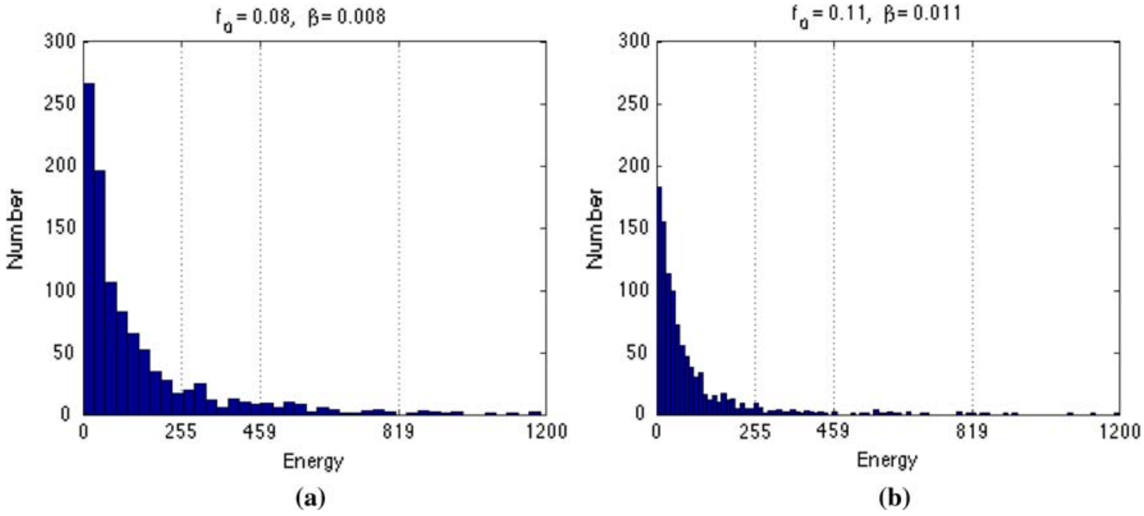


Fig. 8 Histogram of the matched filter output of the original image (*dashed vertical lines* represent the energy level of the matched filter output of the watermark, corresponding to PSNR = 35 dB (*right*), 40 dB (*middle*), and 45 dB (*left*)). **a** $\beta_0 = 0.008$, $f_0 = 0.08$.; **b** $\beta_0 = 0.011$, $f_0 = 0.11$

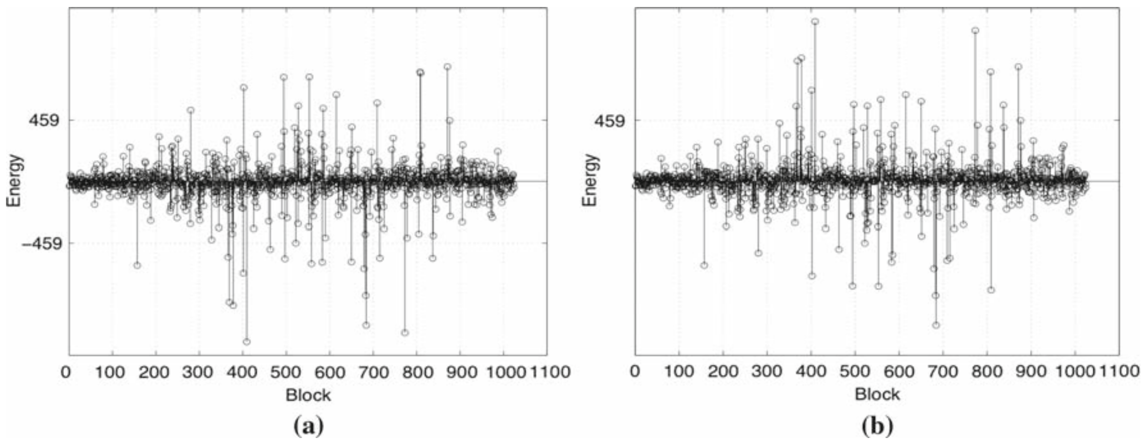


Fig. 9 Matched filter output of the original image. *Dashed line* corresponds to watermark output at (PSNR = 40 dB). **a** $C_I(m, n, \beta_0, f_0)$; **b** $-s(m, n)C_I(m, n, \beta_0, f_0)$

assured. The PSNR required to achieve BER = 0 is 51 dB. Note that the reduction of watermark energy may make it fragile to attacks such as compression.

6.3 Watermark detection capability

For blind watermark detection, the watermark should have enough strength so that its presence can be declared and the chirp parameters can be estimated. Figure 10a and b show the sum of chirp transform spectra of all blocks, respectively, in the absence of the watermark (i.e., $\tilde{C}(\Theta, k) = \tilde{C}_I(\Theta)$) and in the presence of the watermark (i.e., $\tilde{C}(\Theta, k) = \tilde{C}_I(\Theta) + \tilde{C}_W(\Theta, k)$). The watermark chirp parameters are $\beta_0 = 0.011$ and $f_0 = 0.11$, and PSNR = 40 dB. It is evident from this figure that the watermark is visibly present.

6.4 Effect of DC components

The previous simulation results have assumed that the DC component of the watermark spreading function is removed, as discussed in Sect. 3. The next example shows that if the DC component is not removed, there may be a significant bias which, in turn, will affect watermark detections.

Figure 11a shows the histograms of $C(m, n, \beta_0, f_0, k)$. The chirp parameters are $(\beta_0, f_0) = (0.009, 0.09)$ and the PSNR is 40 dB. The histograms corresponding to watermark information +1 and −1 are both depicted. It is obvious that the high bias as well as high overlapping observed in this figure will make the watermark decision difficult. When the DC components are removed, however, as shown in Fig. 11b, no bias and little overlapping are observed.

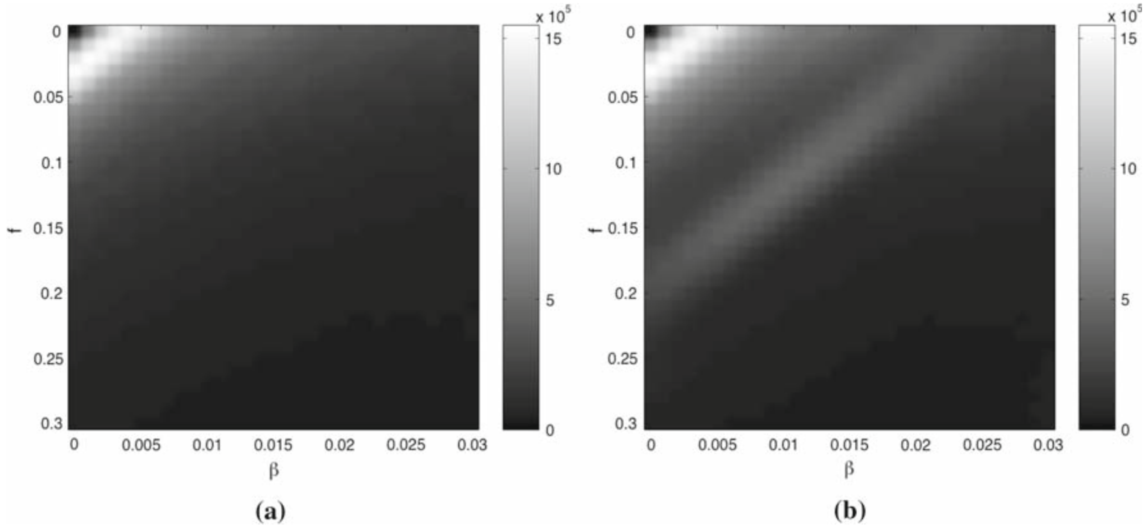


Fig. 10 Sum of chirp transform spectra of all blocks: **a** original *Lena* and **b** watermarked *Lena*

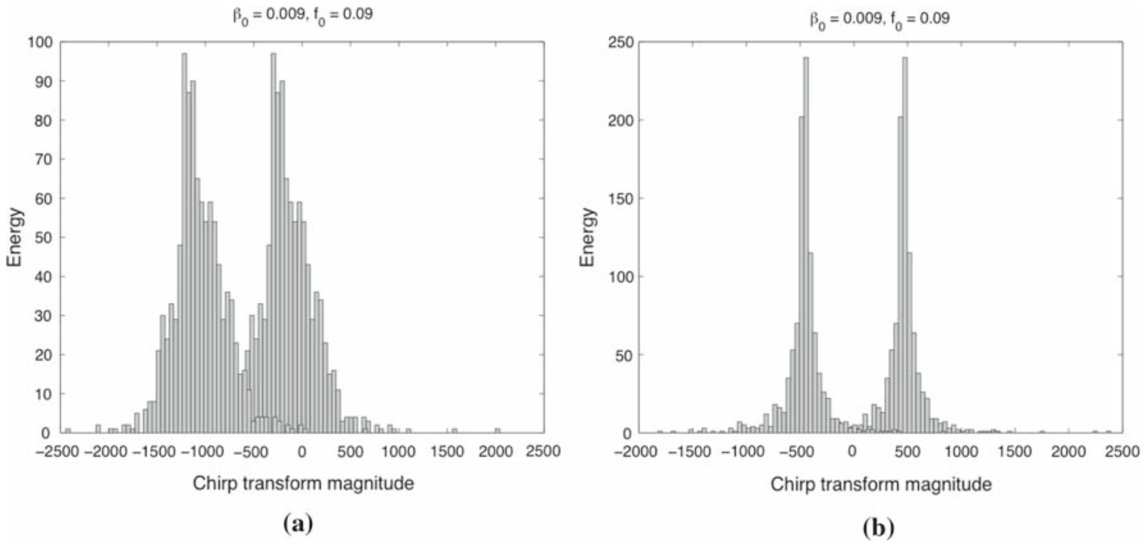


Fig. 11 Histograms of $C(m, n, \beta_0, f_0)$ with $s(m, n) = +1$ and -1 embedded ($\beta_0 = 0.009,\ f_0 = 0.09$): **a** DC components not removed and **b** DC components removed

6.5 Secure watermark embedding using pseudo-random parameters

Figure 12a shows the BER plot for detecting an embedded watermark in *Lena*, where PSNR = 40 dB and the JPEG quality factor is $Q = 50$. The (β_0, f_0) parameters are pseudo-randomly selected and distributed over the region between the two zigzag patterns. Figure 12b shows the sum of chirp transform spectra of all blocks of the watermarked *Lena*. Comparing this figure with Fig. 10b, the advantage of using secure watermark embedding is clear. There is no detectable trace of watermark since the watermark energy is now distributed over a number of different (β_0, f_0) combinations.

Table 1 Bit error count comparison (out of 1024)

Chirp type	No compression	$Q = 75$	$Q = 50$
Chirp with fixed parameters	14	19	39
Chirp with encrypted parameters	20	25	49

Table 1 compares the numbers of bit errors for chirps with fixed ($\beta_0 = 0.011,\ f_0 = 0.11$) and pseudo-random parameters. The BER for the secure watermark with pseudo-random parameters is slightly worse than that of the fixed watermark parameter case because of the inclusion of suboptimal (β_0, f_0) parameters.

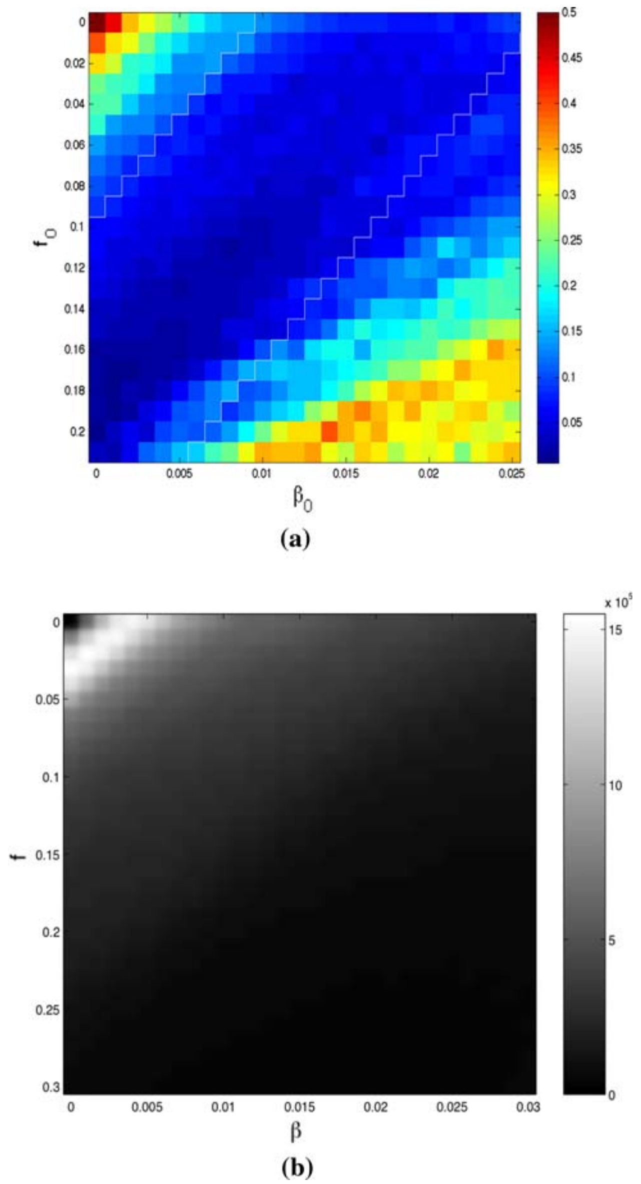


Fig. 12 Secure watermark embedding: **a** error-rate plot overlaid by the (β_0, f_0) region used in watermark embedding and **b** sum of chirp transform spectra of all blocks of watermarked *Lena*

6.6 JPEG compressed imagery

Figure 13 shows contours of the BERs for *Lena*. The purpose of BER contours is to tune the chirp for survival against compression. We sweep (β_0, f_0) over a range of values. For each pair we compute the corresponding BER. The end result is a BER surface, BER (β_0, f_0), that is represented using contour command. The numbers attached to each contours are simply the number of bit errors(out of 1024 embedded bits). As an example, the region between contours 50 and 10 are those (β_0, f_0) pairs that if selected will result in $\frac{10}{1024} \leq BER \leq \frac{50}{1024}$. What is noteworthy is that there are a multitude of chirps that can meet specified BER levels across a wide range of compression. This property allows for secure embedding of chirps with different parameters that are pseudorandomly chosen from the set of eligible pairs. For comparison, we also show the same results for the Elaine picture in Fig. 14. The Elaine picture is shown in Fig. 15b.

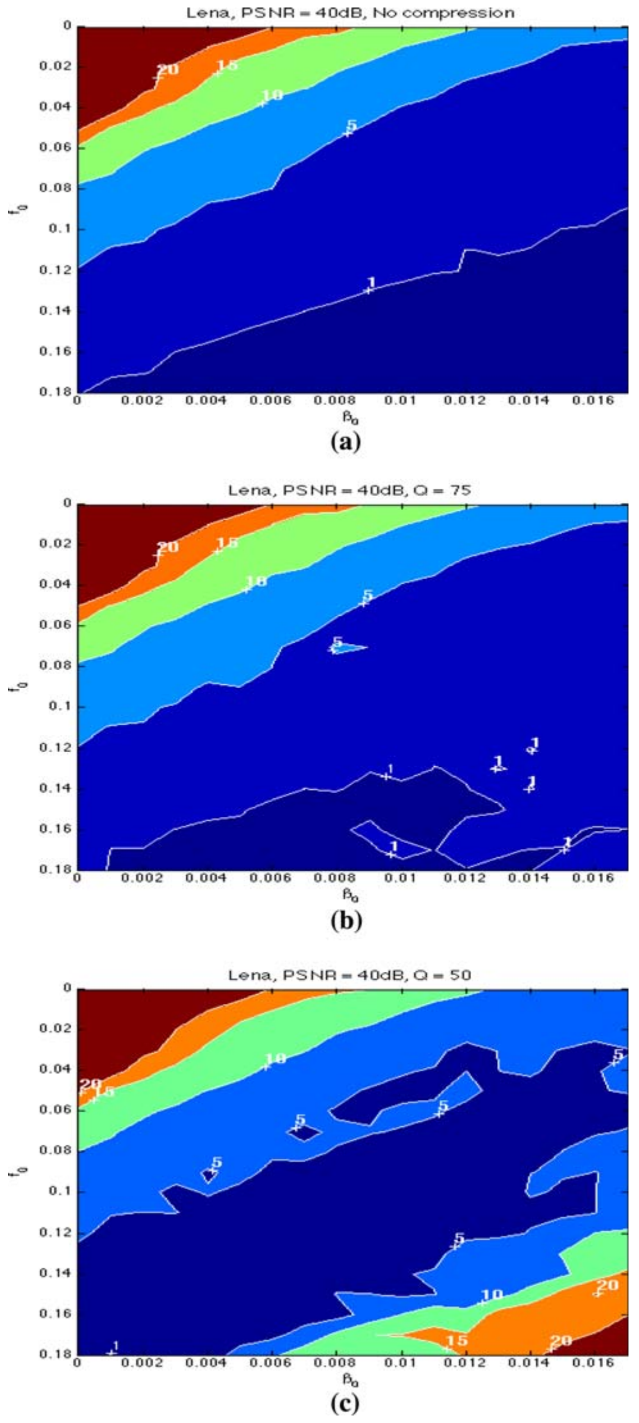


Fig. 13 Bit error count contours for *Lena*. Image carries 1024 bits. **a** No compression, **b** with compression ($Q = 75$), **c** with compression ($Q = 50$)

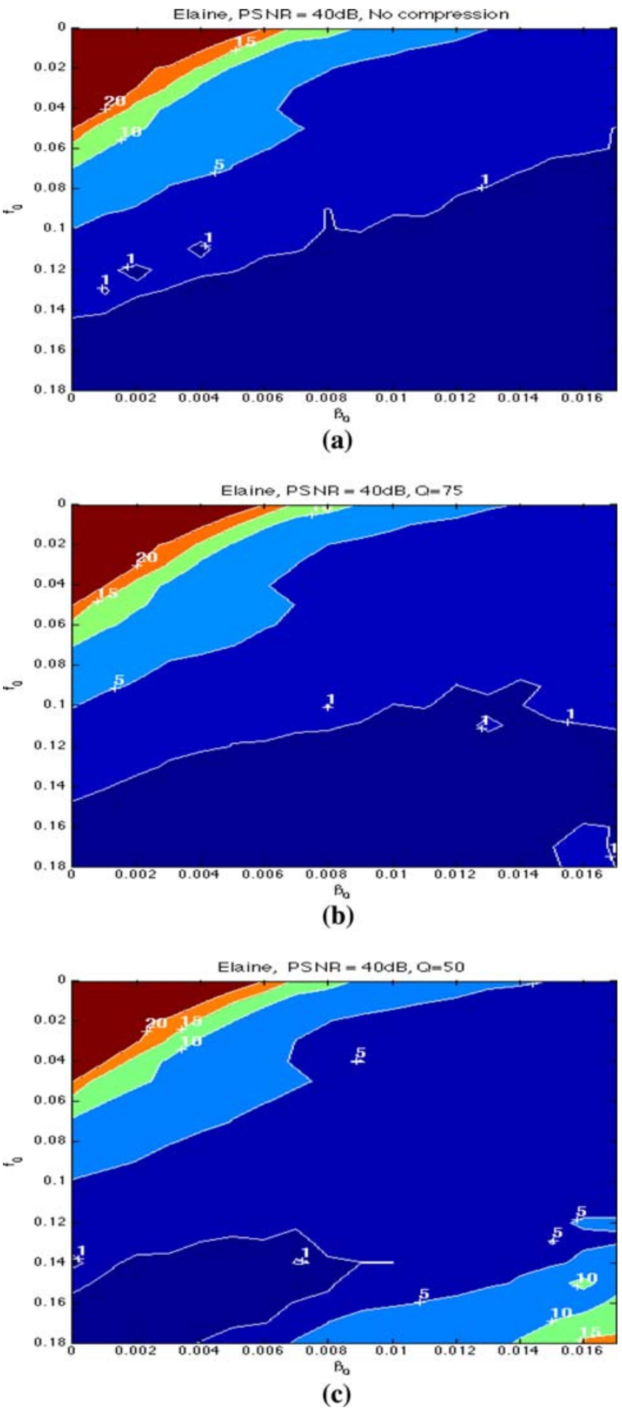


Fig. 14 Bit error count contours for *Elaine*. Image carries 1024 bits. **a** No compression, **b** with compression ($Q = 75$), **c** with compression ($Q = 50$)

It is observed from these figures that, for higher compression factors, the low BERs appear in lower (β_0, f_0) regions. This behavior is consistent with JPEG compression as more high frequency components are suppressed, along with the watermark. Similar simulations have been carried out for Elaine. Trends are similar but variations of BER vs. (β_0, f_0) are clearly image-dependent. It is often desired to select (β_0, f_0) pairs that survive compression across a range of Q factors. Figure 13 can be used to identify overlapping portions of (β_0, f_0) to achieve certain BERs. It is interesting to note that for Elaine, it is possible to select chirps that meet BER < 0.01 across all $Q \geq 50$. The same cannot be said for *Lena*.

An inspection of BER contours reveals another important property of the chirp. As shown in Fig. 14a, to achieve BERs below 0.01 using sinusoids allows the use of only a limited number of frequencies below $f_0 = 0.143$. Using a chirp instead greatly expands the possible (β_0, f_0) pairs that achieve the required BER. The expanded choice is important in tuning the watermark in order to counter compression effects. For example, in *Lena* there are no sinusoids that achieve BER < 0.01 for $Q = 75$, whereas there are plenty of chirps that would achieve this specified BER. This expanded choice is critical in implementing secure embedding of watermark.

6.7 Comparison with spread spectrum watermarking

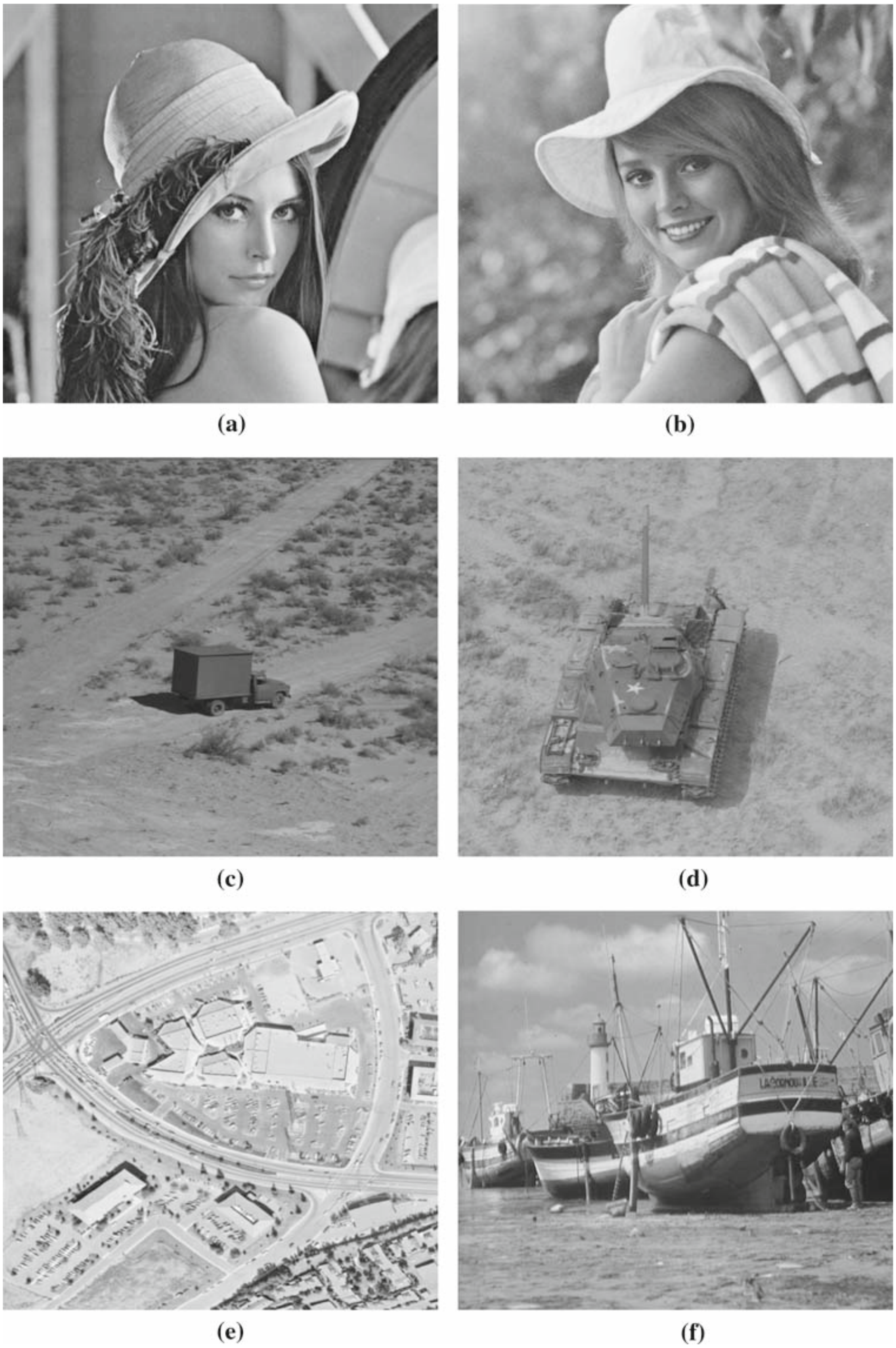
We now compare performance chirp watermarking with that of [14] and [11]. The former is spatial spread spectrum where PN sequences are used to spread a single watermark bit throughout an image block. The latter embeds the watermark in selected coefficients of block DCT. Both approaches are among the more established watermarking approaches and provide a benchmark. Our test database consists of *Lena*, *Elaine*, *Truck*, *Tank*, *Aerial* and *Boat*, all shown in Fig. 15.

In Fig. 16, we show the BER versus Q-factor for six different images. The PSNR is 42.1 dB for both 2D chirp and PN sequence spreading functions. An m-sequence is used as a PN sequence. Note that the watermark using m-sequences can only take certain discrete values of the PSNR while the 2D chirp watermark enjoys greater flexibility in selecting the PSNR. The results corroborate our theoretical understanding that a tunable watermark outperforms a static one. These figures obviously show the better performance of chirp watermark versus m-sequence watermark as far as bit error is concerned, especially for the cases with JPEG compression. As expected, compared to the m-sequences the chirp watermark is significantly less sensitive to JPEG compression.

Another point of comparison is with DCT watermarking [11] whereby a single watermark bit is spread out over 8 × 8 blocks and additively modifies selected DCT coefficients. The watermarked coefficients range from 7th to 28th in JPEG zigzag scan order. Image quality is monitored by Watson's perceptual masking model [21]. At $Q = 75$, they report BER in the range of 10% using the Laplacian model to 25% for the generalized Gaussian model.

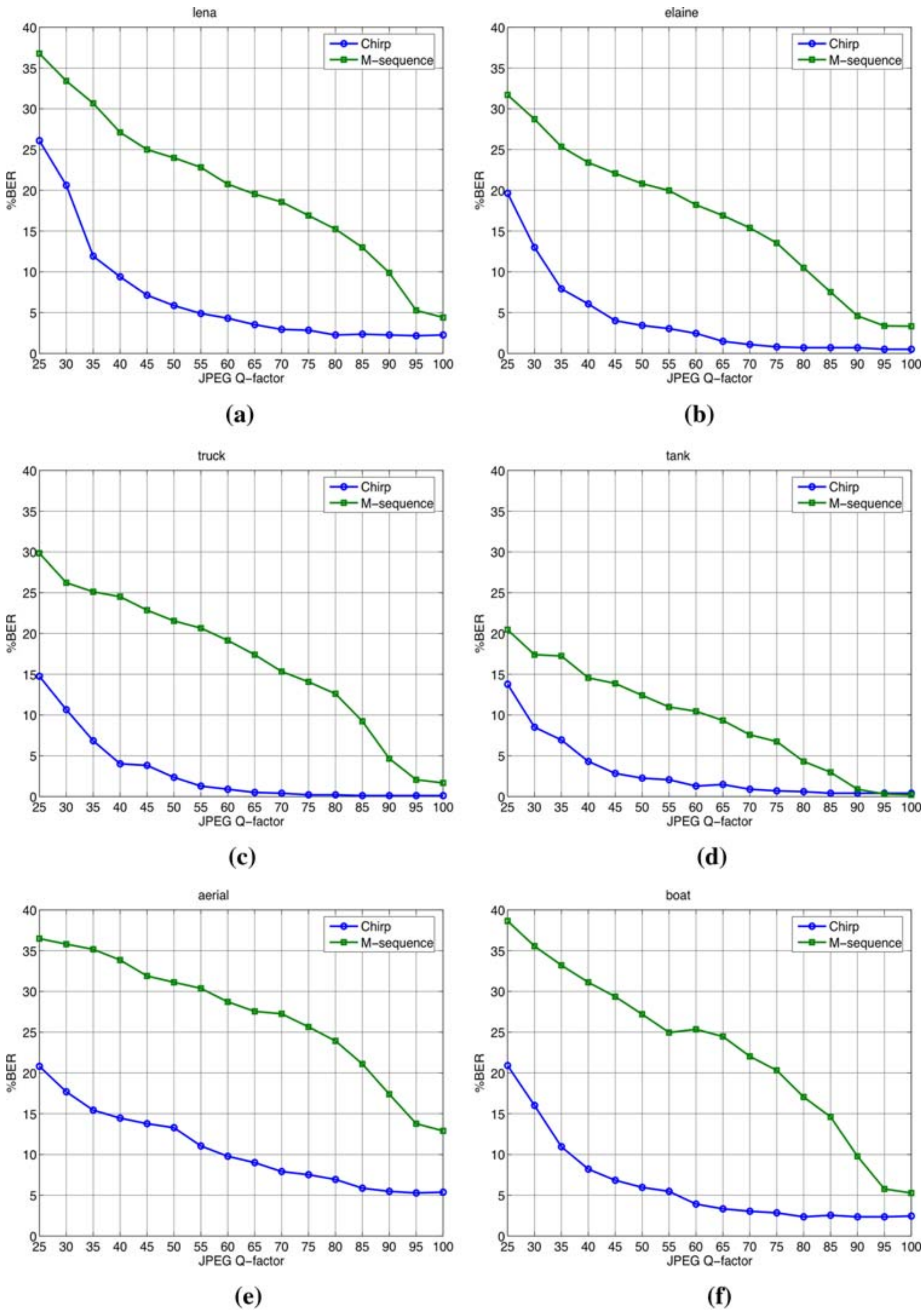
We simulated the same algorithm over 16 × 16 blocks to match the block size used in chirp watermarking. PSNR is

Fig. 15 Images used in performance evaluation: **a** Lena, **b** Elaine, **c** Truck, **d** Tank, **e** Aerial, **f** Boat



maintained at 40 dB. Keeping PSNR and block size the same, BER curves were also generated for chirp watermarking. The two curves are shown superimposed in Fig. 17. DCT watermarking outperforms [14] but chirp outperforms both [14] and [11] across almost all Q factors. The reason DCT watermarking excels the chirp for $Q < 30$ is possibly due to the fact that the chirp was not optimized for this particular range of Q values. Recall that from BER contours in Fig. 15, it is possible to select appropriate chirps to meet BER levels over a range of compressions. Another reason for the better performance of DCT watermarking over spatial spread spectrum is that the former also benefits from some degree of spatial tuning. This tuning is achieved by judicial selection of DCT coefficients for watermarking. However, the evidence

Fig. 16 BER versus JPEG Q-factor for watermarks using 2D chirp and m-sequence: **a** Lena, **b** Elaine, **c** Truck, **d** Tank, **e** Aerial, **f** Boat



(a)

(b)

(c)

(d)

(e)

(f)

here shows that spectral tuning of the chirp provides a better mechanism for watermark compression survival.

7 Watermark robustness against attacks

In this section, we examine the robustness of the 2D chirp watermark in the presence of attacks as well as compression. The results are compared to those of the M-sequence-based spread spectrum watermark. The effects of scaling, blurring, and additive noise are evaluated using *Lena*. JPEG compression was performed with different Q factors. In all the examples, the watermark energy is 42.1 dB lower than the peak image level (PSNR $= 42.1$ dB). The 2D chirp watermark is parameterized by $\beta_0 = 0.011$ and $f_0 = 0.11$.

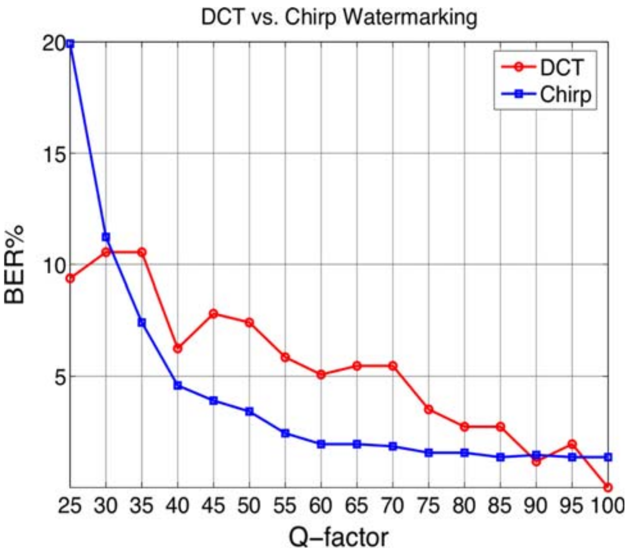


Fig. 17 BER for DCT and chirp watermaking for block size of 16 × 16 and PSNR of 40 dB

It is pointed out that, similar to spread spectrum watermarks, the 2D chirp watermark has no inherent resilience to geometric attacks such as rotation, scale and/or translation of watermarked image. Any of the above three operations will cause loss of synchronization and possibly massive detection failure. Nevertheless, we notice that increasing the size of the partitioned blocks will likely enhance the robustness against geometric attacks [4]. Recent work has addressed robustness of chirp watermarking to rotation by mapping the chirp from Cartesian coordinates to log polar domain [22]. This approach combines tunability of the chirp with rotation invariant property of log polar representation.

7.1 Scaling

To evaluate the effect of scaling on watermark detection performance, the image is downsampled by 2:1 ratio and then upsampled to original dimensions using spline interpolation. The results are shown in Fig. 18a. The 2D chirp watermark consistently provided lower BER than the m-sequence over different JPEG compression Q factors. It can be explained that the scaling has less impact on a 2D chirp watermark because there is no significant presence of high frequency components in the 2D chirp whereas the spread spectrum signal is more sensitive to any change in the waveform. Note, however, that when the watermarked image is scaled using higher scaling ratio (e.g., 4:1), the BER of both types of watermarks becomes unacceptable (32.0 and 35.3% for 2D chirp and M-sequence watermarks, respectively, without JPEG compression).

7.2 Lowpass filtering

The watermarked image is put through 1-pixel Gaussian blur using Photoshop. Blurring is controlled by the support of the Gaussian function specified in pixels. The blurred image is then JPEG compressed. Note that compression is essentially a lowpass filter. Figure 18b shows reasonable resiliency of the 2D chirp watermark to combined filtering/compression, whereas the spread spectrum watermark consistently show BER values close to 50% for all JPEG compression ratios. Increasing the support of the Gaussian blur to 2 pixels results in noticeably degraded image and the BER values increase to unacceptable levels (38.4 and 50.8% for 2D chirp and M-sequence watermarks, respectively, without JPEG compression).

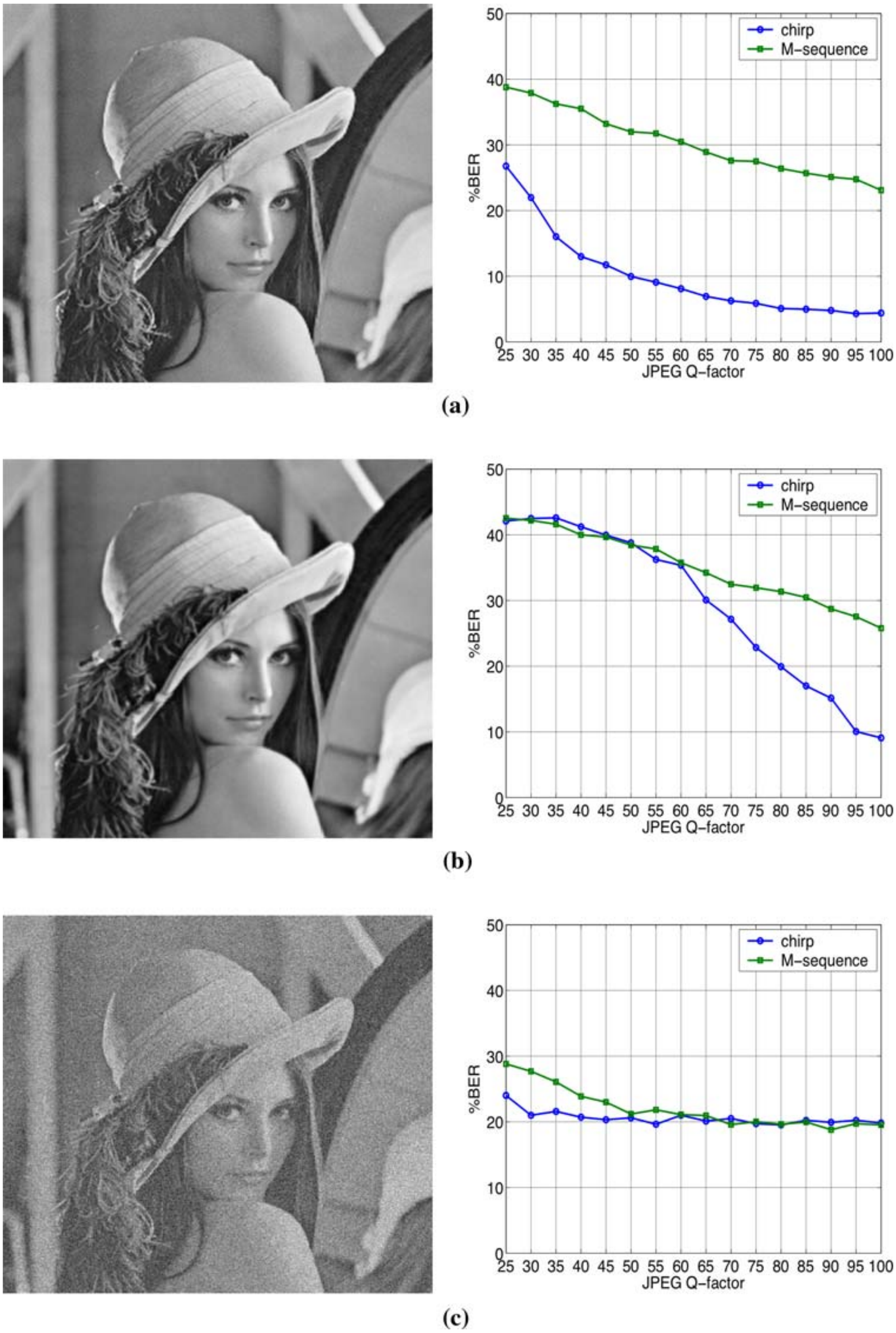
7.3 Additive noise

To measure the impact of random noise on BER, the watermarked image is corrupted by Gaussian noise before JPEG compression. The PSNR resulting from the noise is 20 dB. This is a fairly strong noise that has noticeably altered the image. The resulting BER values depicted in Fig. 18c show the impact of additional noise but at most operating Q factors, the decoded watermark will still be recognizable. Both types of watermarks show comparable BER performance.

8 Conclusions

In this paper, a novel method has been proposed for digital watermark embedding and detection. The watermarking is based on 2D FM spreading functions which can be designed for flexible spectrum allocation, so that the parameters can be optimized to distinguish itself from the original image for improved watermark detectability, and to avoid high frequencies so that the robustness to compression attacks can be maintained. In particular, we used 2D chirp signals as examples for extensive investigation. Performance advantages over spread spectrum techniques has been demonstrated. In addition, it has been shown that the adaptive chirp power allocation technique improves the performance as well as the imperceptibility of the watermarked image. The advantage of secure watermark embedding using random chirp parameters has also been demonstrated.

Fig. 18 Watermarked image and BER performance in the presence of attacks: **a** ratio 2:1 scaling attack, **b** 1-pixel Gaussian blurring attack, **c** noise attack (noise power is 20 dB below image peak level)



References

1. Langelaar, G.C., Setyawan, I., Lagendijk, R.L.: Watermarking digital image and video data. IEEE Signal Processing Mag. **17**(5), 20–46 (2000)
2. Kayshan, A.S.: Representation and analysis of complex chirp signals. Signal Processing **66**, 111–116 (1998)
3. Chen, V.C., Liang, H.: Time–Frequency Transforms for Radar Imaging and Signal Analysis. Artech House, London (2002)
4. Stanković, S., Djurović, I., Pitas, I.: Watermarking in the space/spatial-frequency domain using two-dimensionall Radon–Wigner distribution. IEEE Trans. Image Processing **10**, 650–658 (2001)
5. Mobasseri, B.G.: Digital watermarking in the joint time–frequency domain. In: Proc. IEEE International Conference on Image Processing, vol. 3, pp. 481–484 (2002)
6. Mobasseri, B., Zhang, Y., Dogahe, B.M., Amin, M.G.: Designing robust watermarks using polynomial phase exponentials. In: IEEE

Int. Conf. Acoustics, Speech, and Signal Processing, Philadelphia, PA, March (2005)

7. Zhang, Y., Dogahe, B.M., Amin, M.G., Mobasseri, B.: Digital watermarking using two-dimensional FM waveforms. In: Advanced Signal Processing Algorithms, Architectures, and Implementations XIV, vol. 5599, pp. 277–288. SPIE (2004)
8. Aviyente, S., Al-khassaweneh, M.: Robust watermarking in the joint spatial-spectral domain. IEEE Digital Signal Processing Workshop, Taos Ski Valley, NM (2004)
9. Barkat, B., Sattar, F.: A new time–frequency based private fragile watermarking scheme for image authentication. IEEE Int. Symp. Signal Processing Appl. **2**, 363–366 (2003)
10. Ramalingam, A., Krishnan, S.: Robust image watermarking using a chirp detection-based technique. IEE Proc. **152**(6), 771–778 (2005)
11. Hernández, J.R., Amado, M., Pérez-González, F.: DCT-domain watermarking techniques for still images: detector performance analysis and a new Structure. IEEE Trans. Image Processing **9**(1), 55–68 (2000)
12. Erkucuk, S., Krishnan, S., Zeytinoglu, M.: A Robust audio watermark representation based on linear chirps. IEEE Trans. Multimedia **8**(5), 925–936 (2006)
13. Le, L., Krishnan, S.: Time–frequency signal synthesis and its application in multimedia watermark detection. EURASIP J. Appl. Signal Processing **2006**(5), 925–936 (2006)
14. Kutter, M., Winkler, S.: A vision-based masking model for spread-spectrum image watermarking. IEEE Trans. Image Processing **11**, 16–25 (2002)
15. Cohen, L.: Time–Frequency Analysis. Prentice Hall, Upper Saddle River (1995)
16. Papandreou-Suppappola, A.: Applications in Time–Frequency Signal Processing. CRC Press, Boca Raton (2003)
17. Shen, H., Machineni, S., Gupta, C., Papandreou-Suppappola, A.: Time-varying multichirp rate modulation for multiple access systems. IEEE Signal Processing Lett. **11**(5), 497–500 (2004)
18. Avcibas, I., Memon, N., Sankur, B.: Steganalysis using image quality metrics. IEEE Trans. Image Processing **12**(2), 221–229 (2003)
19. Sheikh, H., Bovik, A.: Image information and visual quality. IEEE Trans. Image Processing **15**(2), 430–444 (2006)
20. Ebrahimi, F., Chamik, M., Winkler, S.: JPEG vs. JPEG 2000: an objective comparison of image encoding quality. Proc. SPIE Appl. Digital Image Processing XXVII **5558**, 303–308 (2004)
21. Watson, A.B.: Visula optimization of DCT quantization matrices for individual images. In: Proc. AIAA Computing in Aerospace, vol. 9, pp. 286–291, San Diego, CA (1993)
22. Fleming, C.E., Mobasseri, B.G.: Compression and rotation resistant watermark using a circular chirp structure. In: Proc. SPIE Conference Security, Steganography, and Watermarking of Multimedia Contents VIII, San Jose, CA, Jan. (2006)