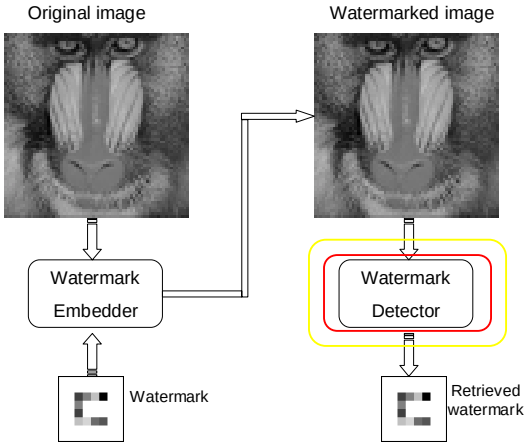
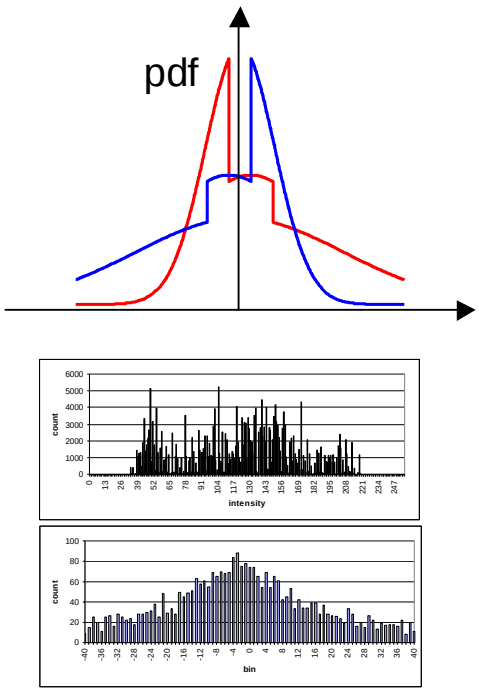
Carnegie Mellon

Electrical & Computer ENGINEERING

A Framework for Optimal Watermark Detection

Trista Pei-chun Chen and Tsuhan Chen
Advanced Multimedia Processing Lab





Technical Report AMP 01-03
September 2001

Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213

A Framework for Optimal Public Watermark Detection

Trista Pei-chun Chen and Tsuhan Chen*

Abstract—We propose a general framework for public watermark detection where watermark information can be detected or extracted without using the original image. In this framework, we use a maximum-likelihood watermark detector that is optimal based on the probability distribution of the noise due to the original image and possibly other distortions. Several existing watermark detectors are shown to be special cases of the proposed framework and hence have sub-optimal performance. The proposed framework also incorporates watermarks designed according to human visual system characteristics. While several existing watermark systems consider the human visual system to make the embedded watermark invisible, we show that such consideration can be incorporated into watermark detection too. We demonstrate this framework in two domains, the pixel domain and the discrete cosine transform domain, but the framework can be applied to other domains as well. When watermark embedding and detection are performed in the pixel domain, we model the probability distribution of the noise, the pixel values of the original image, as a Gaussian mixture. When watermark embedding and detection are performed in the discrete cosine transform domain, we model the probability distribution of the noise, in this case the discrete cosine transform coefficients of the original image, as a generalized Gaussian distribution. Experiment results show that the proposed watermark detector performs better than conventional methods in both domains.

Index Terms-- public watermark, blind watermark detection, human visual system, maximum-likelihood detection, Gaussian mixture, generalized Gaussian distribution.

EDICS: 5-AUTH

The authors are with Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213.
*Corresponding author: Tsuhan Chen, Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, Tel: (412) 268-7536 Fax (412) 268-3890, E-mail: tsuhan@cmu.edu

I. INTRODUCTION

With the success of information technology, more digital data can be accessed and processed easily. Users can download, store, and re-transmit digital media, such as images, video, and audio, without offering appropriate credit to the content owner. Digital watermark provides a solution to this problem by embedding the owner information in the content. In addition to protecting the copyright of digital media, digital watermark can be used to authenticate the integrity of the content [1]. Research activities in digital watermark have been very active in recent years [2]-[8].

A watermark system is shown in Figure 1. The watermark embedder inserts the watermark, which is certain binary information that carries the copyright or ownership information into the original image. The watermark itself can be an image, such as a logo. The image with the watermark embedded is called the watermarked image. Given the watermarked image, one can retrieve the watermark using the watermark detector. Note that in literature, watermark detection is sometimes referred to correlating the watermarked image with a preset watermark to determine the presence of the preset watermark in the content [9]-[12]. Instead of determining the presence of the preset watermark, the watermark detector in this paper extracts the watermark without using the preset watermark.

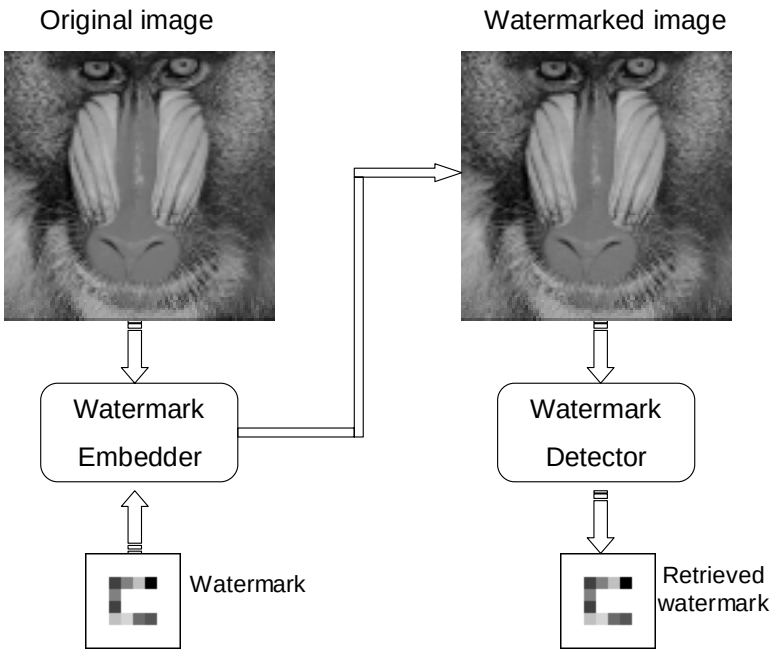


Figure 1. A WATERMARK SYSTEM

Watermark techniques can be categorized into two types, *private* and *public* [4], according to the way they detect the watermark. Private watermark systems need the original image to be subtracted from the watermarked image to detect the watermark. On the other hand, public watermark systems do not need the original image to detect the watermark. The public watermark system is also called the blind watermark system [13]. To resolve the rightful ownership, it is better to adopt public watermark systems [14]. Private watermark systems are restricted to be used by the authorized users or content owners, who have the access to the original image. Moreover, requiring the original image to detect the watermark needs extra storage of the original image at the detector's side or extra bandwidth to send the original image from the embedder to the detector. Therefore, we will focus on public watermark systems in this paper. Some prior work on public watermark detection can be found in [13][15]. The proposed optimal public watermark detector is better than the prior work by adopting human visual system characteristics and by better modeling the probability distribution of the noise, which we will explain briefly as follows.

In order not to interfere with the viewing experience, the embedded watermark should be imperceptible or invisible. Therefore, human visual system (HVS) [16] characteristics are often included to meet this

criterion [3][4]. In this paper, we will show that the watermark detector can use the knowledge of how the watermark is related to the HVS to improve the detection performance.

Based on the general framework of public watermark detection considering the HVS, we show how better modeling of the probability distribution function (pdf) of the noise can improve the watermark detection performance. The noise contains the original image and possibly other image artifacts. When the watermark is embedded and detected in the pixel domain, we model the pdf of the noise in the pixel domain as a Gaussian mixture. The result is compared to special cases that use a simple Gaussian pdf. Other modeling work in the literature includes using the Cauchy distribution [17][18]. Comparing modeling using the Gaussian mixture with the Cauchy pdf, we found that the former can model pdf's with multiple modes (with examples shown in Figure 3) and has a closed-form expression. When the watermark is embedded and detected in the discrete cosine transform (DCT) domain, we model the pdf of the DCT coefficients as a generalized Gaussian distribution [19]. The result is then compared to other special cases of DCT domain watermark methods. The concept of better modeling the pdf of the noise can be applied to watermark methods in other domains. We can also improve the detection performance by providing the watermark detector with some side information.

This paper is organized as follows. In Section II, we propose a general framework for public watermark detection based on maximum-likelihood (ML) detection. This public watermark detection framework in particular considers the watermark that is designed based on the HVS, and better models the pdf of the noise. In Section III, we show how the proposed ML watermark detector performs in the pixel domain. In Section IV, we show how the proposed ML watermark detector performs in the DCT domain. We conclude our work and discuss future research directions in Section V.

II. Public Watermark Detection: Framework and Optimal Detector

As mentioned in the previous section, to resolve the rightful ownership and not to restrict only the authorized users to extract the watermark, we focus on public watermark systems. In this section, we first

form a general framework for public watermark systems. Other watermark detection methods are within this framework as special cases. Watermark detection in the proposed framework is by performing ML detection. In particular, this framework considers the case where the watermark is designed based on the HVS.

A. A General Framework for Public Watermark Detection

A watermark is represented by a number of bits. Watermark detection can be formulated as a hypothesis-testing problem given the received watermarked image. To insert a watermark to an original image, the watermark embedder alters some signal values of the original image, such as pixel values or transform domain coefficients (DCT coefficients for example), to embed a bit 1 or 0 for each bit of the watermark. We call the resulting signal values the watermarked signals. To detect the watermark from the watermarked image, the watermark detector performs bit-wise detection for each bit of the watermark.

If we consider the watermark as the message in a communication system that the receiver wants to detect, the *noise* in the detection process then includes the original image, compression artifacts if the watermarked image is compressed, and other possible distortions and attacks to the watermark. Since we are considering public watermark detection, the original signal is the major noise term that affects the watermark detection. Other types of noise, e.g., compression artifacts and attacks, have less impact compared to the original image.

In the proposed framework, each watermark bit can be embedded N times. Embedding the same bit multiple times can improve the robustness of this embedded bit, at the expense that fewer distinctive bits can be embedded in total. The watermark detector determines the value of each watermark bit from N received watermarked signals, $y_1, y_2, \ldots, y_N$. We can write the two hypotheses associated with a bit 1 and 0 as follows:

$$\begin{aligned} H_1 &: y_i = x_i + w_i \\ H_0 &: y_i = x_i - w_i \end{aligned} \qquad i = 1, \ldots, N \tag{1}$$

where y_i represents the received watermarked signal, x_i represents the original signal, and w_i represents the watermark amplitude. If a bit 1 is embedded, the original signal x_i is raised by w_i. If a bit 0 is embedded, the original signal x_i is lowered by w_i. To further include the HVS characteristics detailed in Section III and Section IV, we let the watermark amplitude to be a function, $f_i(x_i)$, of the original signal x_i. This is an extension to [6] that assumes the watermark amplitude to be a linear transformation of the original signal. In addition to the dependency on x_i, $f_i(\cdot)$ can be different for each of the N signals in general, so we have:

$$\begin{aligned} H_1 &: y_i = x_i + f_i(x_i) \equiv g_{i,1}(x_i) \\ H_0 &: y_i = x_i - f_i(x_i) \equiv g_{i,0}(x_i) \end{aligned} \qquad i = 1, \ldots, N \tag{2}$$

where we represent the watermarked signal with a bit 1 embedded as $y_i = g_{i,1}(x_i)$ and the watermarked signal with a bit 0 embedded as $y_i = g_{i,0}(x_i)$. Note that these two functions $g_{i,1}(x_i)$ and $g_{i,0}(x_i)$ can be different for each i, similar to $f_i(x_i)$. We then rewrite (2) in the vector form as follows:

$$\mathbf{y} = \mathbf{x} + \theta \mathbf{w} \tag{3}$$

where $\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$, $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}$, $\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{pmatrix}$, and $\theta = \pm 1$, which corresponds to a bit 1 or 0. The task of the watermark detector is to determine the value of θ by performing the maximum *a posterior* (MAP) detection.

$$\frac{p(H_1|\mathbf{y})}{p(H_0|\mathbf{y})} = \frac{p(\mathbf{y}|H_1)p(H_1)}{p(\mathbf{y}|H_0)p(H_0)} \begin{array}{c} H_1 \\ \geq \\ < \\ H_0 \end{array} 1 \tag{4}$$

where $p(\mathbf{y}|H_1)$ and $p(\mathbf{y}|H_0)$ are the pdf of $\mathbf{y}$ given hypotheses H_1 and H_0, respectively. The MAP detection is simplified to the maximum-likelihood (ML) detection given that the *a priori* probabilities for both hypotheses are equal, i.e., $p(H_1) = p(H_0)$ [20]. Thus, (4) becomes:

$$\frac{p(\mathbf{y}|H_1)}{p(\mathbf{y}|H_0)} \begin{array}{c} H_1 \\ \geq \\ < \\ H_0 \end{array} 1 \tag{5}$$

Assume that the selection of N signals are random enough so that y_i's are independent with each other conditioned on H_1 (and H_0). That is,

$$p(\mathbf{y}|H_1) = \prod_{i=1}^{N} p(y_i|H_1) \text{ and } p(\mathbf{y}|H_0) = \prod_{i=1}^{N} p(y_i|H_0) \tag{6}$$

Hence, ML detection can be written as:

$$\prod_{i=1}^{N} \frac{p(y_i|H_1)}{p(y_i|H_0)} \begin{array}{c} H_1 \\ \geq \\ < \\ H_0 \end{array} 1 \tag{7}$$

From (2), the pdf of y_i under hypothesis H_1 (or H_0) is:

$$p(y_i|H_1) = p\left[g_{i,1}^{-1}(y_i)\right] \cdot \left| \frac{d\left[g_{i,1}^{-1}(y_i)\right]}{dy_i} \right| = p\left[g_{i,1}^{-1}(y_i)\right] \cdot J_{i,1}(y_i) \tag{8}$$

or

$$p(y_i|H_0) = p\left[g_{i,0}^{-1}(y_i)\right] \cdot \left|\frac{d\left[g_{i,0}^{-1}(y_i)\right]}{dy_i}\right| = p\left[g_{i,0}^{-1}(y_i)\right] \cdot J_{i,0}(y_i)$$

where we define $J_{i,1}(y_i) \equiv \left|\frac{d\left[g_{i,1}^{-1}(y_i)\right]}{dy_i}\right|$ and $J_{i,0}(y_i) \equiv \left|\frac{d\left[g_{i,0}^{-1}(y_i)\right]}{dy_i}\right|$. Note that inverse functions $g_{i,1}^{-1}(\cdot)$ and $g_{i,0}^{-1}(\cdot)$ exist only if functions $g_{i,1}(\cdot)$ and $g_{i,0}(\cdot)$ are one-to-one and onto. Substituting (8) into (7) and taking the logarithm, we obtain that the watermark detector can determine the bit value by the following inequality:

$$\sum_{i=1}^{N}\left\{\ln\left[p\left(g_{i,1}^{-1}(y_i)\right)\right] - \ln\left[p\left(g_{i,0}^{-1}(y_i)\right)\right]\right\} + \sum_{i=1}^{N}\left(\ln\left|J_{i,1}(y_i)\right| - \ln\left|J_{i,0}(y_i)\right|\right) \begin{array}{c} H_1 \\ \geq \\ < \\ H_0 \end{array} 0 \qquad (9)$$

The above (9) is the proposed framework for optimal watermark detection. The terms $g_{i,1}^{-1}(y_i)$, $g_{i,0}^{-1}(y_i)$, $J_{i,1}(y_i)$, and $J_{i,0}(y_i)$, which are all derived from $f_i(\cdot)$, incorporate the HVS into the watermark detection framework. In the simple case where the watermark amplitude is independent of the original signal (that is, $y_i = x_i \pm w_i$ with w_i independent of x_i), $g_{i,1}^{-1}(y_i)$ is simply $y_i - w_i$, $g_{i,0}^{-1}(y_i)$ is $y_i + w_i$, and both $J_{i,1}(y_i)$ and $J_{i,0}(y_i)$ are equal to unity.

Consideration of the HVS ensures that the image distortion due to watermark embedding is imperceptible. In the pixel domain watermarking, we consider the Weber's law [21]. In the DCT domain watermarking, we consider *frequency selectivity* and *contrast sensitivity* [16]. We will explain these cases in detail in Section III and IV. The HVS models used in this paper are only some examples. Other watermark domains and their corresponding HVS models can be integrated into the proposed framework in (9) as well.

B. Optimal Watermark Detector

Based on the proposed watermark detection framework, we can design the optimal watermark detector by estimating the pdf, $p(x)$, of the noise x, in various watermarking domains. Recall that the noise x is mainly due to the original image, with perhaps some compression artifacts and attacks. We call the compressed or attacked watermarked image the corrupted watermarked image. Estimating the pdf of the noise can be done 1) without any side information and 2) with some side information.

1) Without Side Information

Without any side information, a watermark detector can only detect the watermark from the (possibly corrupted) watermarked image. Since the embedded watermark is imperceptible, the corrupted watermarked image is close to the noise (with original image and some artifacts). We can then get a good estimate of the pdf of the noise by modeling the pdf of the corrupted watermarked image.

2) With Side Information

Given certain side information, the watermark detector may get a better estimate of the pdf of the noise. Side information includes:

- Information about the original image—Such information can be explicitly the model of the pdf of the original image, and the parameters. In a special case, the original image itself can be the side information, in which case the public watermark system becomes the private watermark system.
- Information about distortions or attacks—Such information includes types of distortions or attacks that are applied to the watermarked image. The detector can use such information to restore the corrupted watermarked image. For example, if the watermark detector is given the information that the watermarked image has undergone low-pass filtering, it can sharpen the corrupted watermarked image before using this image to estimate the pdf of the noise. We again

assume that by modeling the pdf of the restored watermarked image, we get a good estimate of the pdf of the noise.

In addition, if information about both the original image and the distortion/attack types is given, the detector can estimate the pdf of the overall noise better. For example, if the watermark detector is given the information that the original image has pdf $p_{x_0}(x_0)$ and the extra noise term, with pdf $p_\delta(\delta)$, is added to the watermarked image, then the pdf $p_x(x)$ of the overall noise can be represented as the convolution of these two pdf's:

$$p_x(x) = \int p_{x_0}(t) \cdot p_\delta(x-t)dt \tag{10}$$

The resulting $p_x(x)$ is used in (9). A good reference of getting the pdf of the extra noise terms can be found in [5].

III. Pixel Domain Watermark Detection

A. Gaussian Mixture Modeling

We first demonstrate the proposed watermark detector in the pixel domain. In this case, the detector needs to estimate the pdf of the noise from the corrupted watermarked image. We model the pdf of the noise as a Gaussian mixture with M components.

$$p(x) = \sum_{j=1}^{M} w_j N_{(m_j,\sigma_j^2)}(x) \tag{11}$$

where $N_{(m_j,\sigma_j^2)}(x)$ is a Gaussian distribution function with mean m_j and variance σ_j^2. w_j is the weight of each component. By substituting (11) into (9), we can rewrite (9) as:

$$\sum_{i=1}^{N}\left\{\ln\left[\sum_{j=1}^{M} w_j N_{(m_j,\sigma_j^2)}\left(g_{i,1}^{-1}(y)\right)\right]-\ln\left[\sum_{j=1}^{M} w_j N_{(m_j,\sigma_j^2)}\left(g_{i,0}^{-1}(y)\right)\right]\right\}+\sum_{i=1}^{N}\left(\ln\left|J_{i,1}(y_i)\right|-\ln\left|J_{i,0}(y_i)\right|\right) \begin{array}{c} H_1 \\ \geq \\ < \\ H_0 \end{array} 0 \quad (12)$$

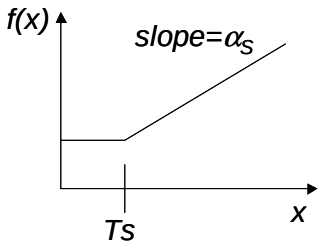
Compared with conventional watermark detectors, which assume that the pdf of the pixel values is Gaussian distributed [13], the proposed detector models the pdf better by a Gaussian mixture after observing the histogram of the pixel values.

The number of components M of a Gaussian mixture affects the modeling performance significantly. If M is too small, the Gaussian mixture cannot model the histogram well. On the other hand, if M is too big, the Gaussian mixture may over fit the histogram of the corrupted watermarked image and stray away from the desired pdf of the noise. We can dynamically determine the number of Gaussian mixture components M as follows.

Start from one component, i.e., $M=1$. M is incremented by one at a time. Use the estimation-maximization (EM) algorithm [22] to estimate the pdf. We test the correlation (or the relative entropy [23]) between the newly estimated pdf and the histogram. If it is greater than the correlation between the old estimated pdf and the histogram by more than a threshold ε_ρ (or smaller than the relative entropy between the old estimated pdf and the histogram by more than a threshold ε_d), we continue increasing M. Otherwise, we stop and use the current value M. The small value ε_ρ (or ε_d) is to make sure M components can model the pdf much better than $(M-1)$ components. Thus, it prevents the Gaussian mixture from over fitting the histogram of the corrupted watermarked image.

B. HVS in the Pixel Domain

In the pixel domain, we design the watermark amplitude $f_i(x_i)$ based on the Weber's law [21]. The relation is as follows:



$$f(x) = \alpha_S \cdot \max\{x, T_S\} \tag{13}$$

where x is the pixel value, α_S is the proportionality multiplier, and T_S is the fixed amplitude used when the pixel value is too small. This relation indicates that the watermark amplitude in the pixel domain is proportional to the pixel value; when the pixel value is smaller than T_S, we can still alter the pixel value to some extend while keeping the watermark imperceptible. The two constants α_S and T_S are adjustable depending on the imperceptibility requirement of the application. Note that the choice of α_S should not be greater than unity or it will violate the one-to-one and onto conditions of functions $g_{i,1}(x)$ and $g_{i,0}(x)$. In addition to the Weber's law, other pixel domain HVS models can be incorporated into the proposed framework as well.

C. Experiments

To show how different pdf modeling methods affect the watermark detection performance, four methods are compared: (a) single Gaussian pdf (Gaussian mixture with one component); (b) Gaussian mixture with two components; (c) Gaussian mixture with the number of components determined as described in Section III-A with $\varepsilon_\rho = 0$; (d) Gaussian mixture with the number of components determined as described in Section III-A with $\varepsilon_\rho = 0.005$.

The three test original images are 8-bit graylevel images of size 512×512 (Figure 2 (a)-(c)). The 10 test watermarks are 4-bit graylevel images of size 8×8 (Figure 2 (d)-(m)), each carrying 256 bits of information. All watermarks are chosen so that the *a priori* probabilities for both hypotheses are equal.

With all combinations of originals and watermarks, there are 30 sets of data. The experiment results shown in the following are the average of these 30 sets of data.

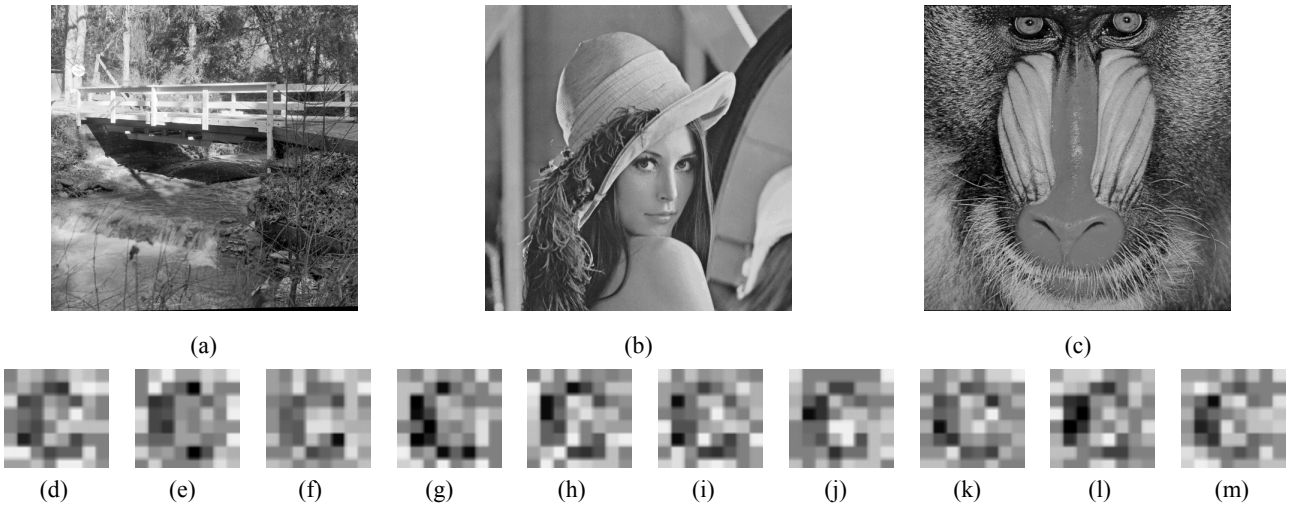
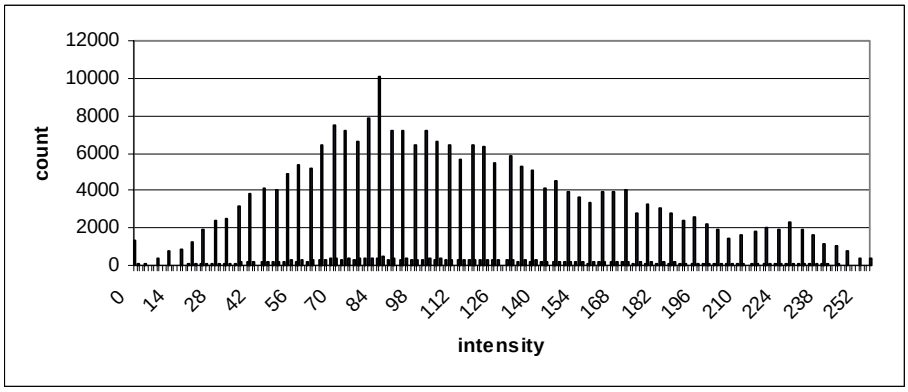


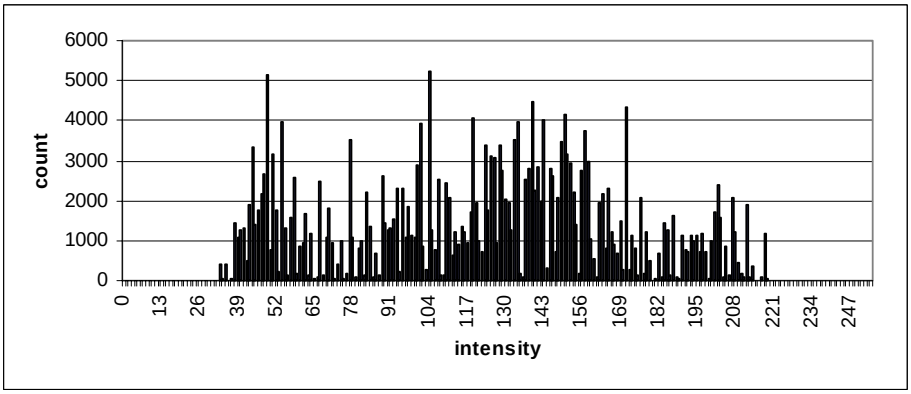
Figure 2. TEST IMAGES: (a)~(c) ORIGINAL IMAGES: BRIDGE, LENA, AND MANDRILL; (d)~(m) WATERMARKS

Each bit of the watermark is distributed randomly across the original image according to a key to $N = 1024$ pixels. The amount by which the original pixel value is altered is determined by (13). We use $T_S = 20$ and $\alpha_S = 0.05$ in the experiments. The watermarked images have an average PSNR of 32.0dB, indicating the watermark is imperceptible. At the watermark detector, each bit of the watermark is detected by performing ML detection as shown in (12). We assume that the detector knows the key the embedder uses to distribute each bit of the watermark. The key itself can be protected by encryption techniques such as those in [24].

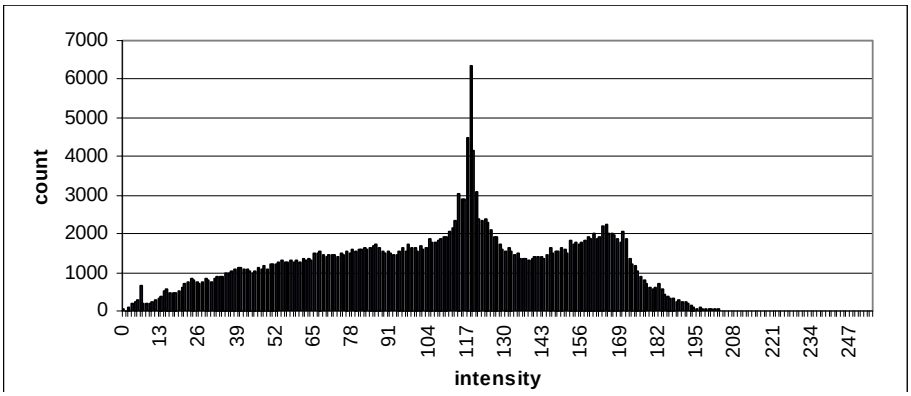
We use the received watermarked image to estimate the pdf of the noise in the experiment. Histograms of the pixel values of the received watermarked images are shown in Figure 3. We see that such distribution cannot be modeled well by a single Gaussian function, so we use the Gaussian mixture to model the pdf of the received watermarked image.



(a)



(b)



(c)

Figure 3. HISTOGRAM OF THE PIXEL VALUES OF IMAGES: (a) BRIDGE; (b) LENA; (c) MANDRILL

As an example, Approach (d) on the average uses four Gaussian mixture components to model the pdf of the received watermarked image. It requires three parameters to describe each Gaussian component—mean m_j, variance σ_j^2, and weight w_j. Therefore, a total of 12 parameters are needed to estimate the pdf of the image. EM algorithm is used to obtain these parameters.

We show the performance of the watermark detectors in Figure 4 and Figure 5. In Figure 4, we show the bit error rates (BER) of the retrieved watermark with respect to the original watermark at various quality factors of JPEG compression (see Appendix Section VII-A for more detail) applied to the watermarked image. In Figure 5, we show the peak signal to noise ratios (PSNR) of the retrieved watermark with respect to the original watermark at various quality factors. PSNR of the retrieved watermark can be considered because the watermarks used in this paper are images. Note that the average PSNR is obtained by first computing the average mean square error (MSE) of all data followed by calculating the PSNR from the resulting average MSE. When quality factor is greater than 80, BER of Approach (d) reaches zero and the corresponding PSNR goes to infinity; when quality factor is greater than 90, BER of Approach (b) and (c) goes to zero and the corresponding PSNR goes to infinity.

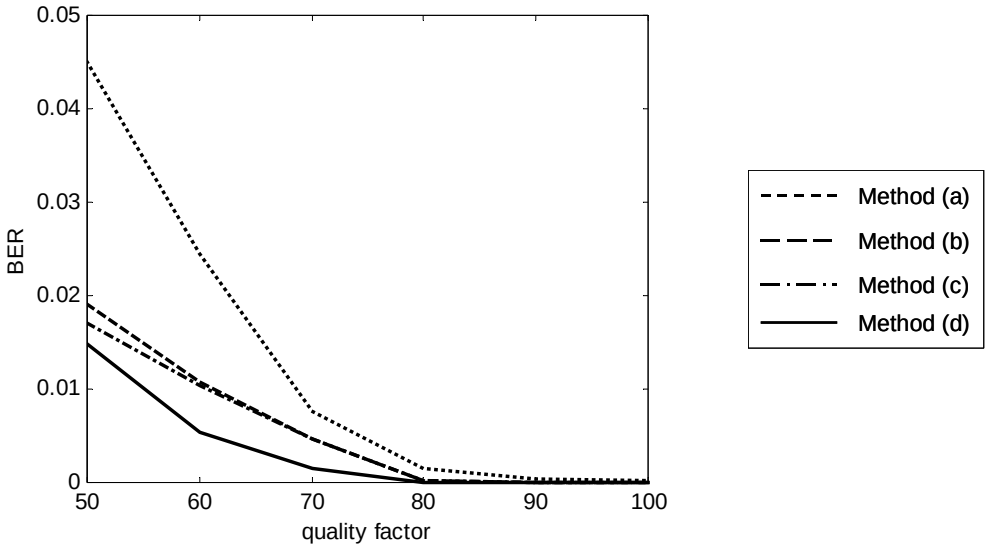


Figure 4. BER PERFORMANCE OF THE PIXEL DOMAIN WATERMARK APPROACHES (a)-(d)

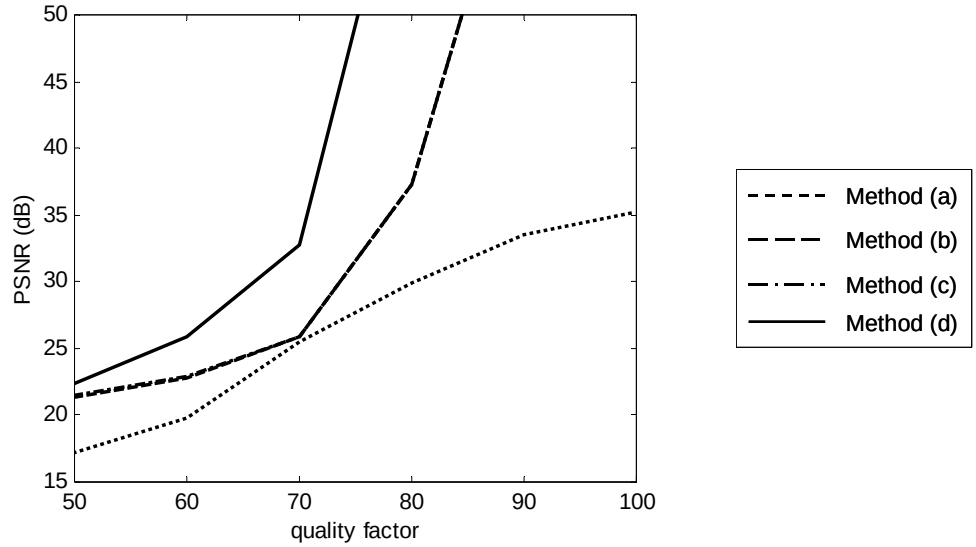


Figure 5. PSNR PERFORMANCE OF THE PIXEL DOMAIN WATERMARK APPROACHES (a)-(d)

It is shown from Figure 4 and Figure 5 that the proposed watermark detector (d), with Gaussian mixture modeling, dynamic determination of the number of components, and $\varepsilon_\rho = 0.005$, performs the best. In this setting, the Gaussian mixture well captures the pdf of the noise but prevents itself from over fitting the histogram of the pixel values of the received watermarked image. The conventional watermark detector (a), which assumes one single Gaussian, performs the worst. Watermark detector (b), which assumes two Gaussian mixture components, performs slightly better than (a). Watermark detector (c), which aggressively uses too many components to model the pdf, over fits the histogram of the pixel values of the received watermarked image and gives worse performance than the proposed watermark detector (d). In addition to the performance differences of these four approaches, all of them have lower detection BER and higher PSNR when the quality factor is larger in which the quantization is less serious.

We also show in Figure 6 that it is reasonable to use the received corrupted watermarked image to estimate the pdf of the noise (with original image and compression artifacts). We subtract the watermark $\pm f_i(x_i)$ from the received watermarked signal y_i to form the real noise. It is shown that the BER performances of using the corrupted watermarked image and using the real noise to estimate the pdf of

the noise are comparable. Thus, we use the received watermarked image to estimate the pdf of the noise for (12). Note that this experiment is only to validate the use of received watermarked image to estimate the pdf of the noise. The proposed watermark detector does not use the real noise since the real noise is not available to the watermark detector.

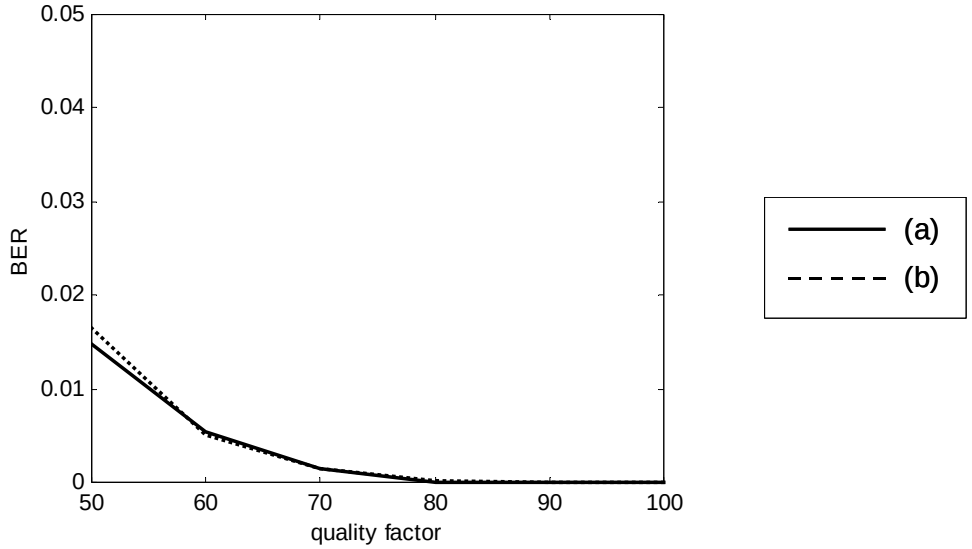


Figure 6. BER PERFORMANCE OF PIXEL DOMAIN WATERMARK DETECTORS USING (a) THE CORRUPTED WATERMARKED IMAGE AND (b) THE REAL NOISE

IV. DCT DOMAIN WATERMARK DETECTION

A. Generalized Gaussian Modeling

We will demonstrate in this section the proposed watermark detector in the DCT domain. The pdf of the noise from the received watermarked image is modeled as a *generalized Gaussian distribution* $p(x)$ [19],

$$p(x) = Ae^{-|\beta x|^c}; \beta = \frac{1}{\sigma}\sqrt{\frac{\Gamma(3/c)}{\Gamma(1/c)}}; A = \frac{\beta c}{2\Gamma(1/c)} \tag{14}$$

where σ is the standard deviation of the DCT coefficient x, c is the generalized Gaussian distribution parameter, and $\Gamma(\cdot)$ is the Gamma function. The Gaussian distribution is a special case of the generalized

Gaussian distribution when $c=2$. The Laplacian distribution is another special case when $c=1$. By substituting $p(x)$ in (14) into (9), we can rewrite (9) as:

$$\sum_{i=1}^{N}\left(\left|\beta_i \cdot g_{i,0}^{-1}(y_i)\right|^{c_i}-\left|\beta_i \cdot g_{i,1}^{-1}(y_i)\right|^{c_i}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \sum_{i=1}^{N}\left(\ln\left|J_{i,0}(y_i)\right|-\ln\left|J_{i,1}(y_i)\right|\right) \tag{15}$$

(15) is the proposed watermark detector in the DCT domain, which performs ML detection and takes into account that the watermark amplitude is a function of the original DCT coefficient. This is more general than the watermark detector proposed by Hernández *et al* [19] because we consider the HVS. In particular, if we let $f_i(x_i)$ to be a constant value d_i, $g_{i,1}(\cdot)$ and $g_{i,0}(\cdot)$ become linear functions and the derivatives of $g_{i,1}^{-1}(\cdot)$ and $g_{i,0}^{-1}(\cdot)$ become unity, i.e., $J_{i,1}(\cdot)=J_{i,0}(\cdot)=1$. In this case, the proposed watermark detector degenerates to Hernández's watermark detector:

$$\sum_{i=1}^{N}\left(\left|\beta_i(y_i+d_i)\right|^{c_i}-\left|\beta_i(y_i-d_i)\right|^{c_i}\right) \underset{H_0}{\overset{H_1}{\gtrless}} 0 \tag{16}$$

Further, if the pdf of the DCT coefficients is modeled as a simple Gaussian distribution, i.e., $c_i=2$, (16) becomes:

$$\sum_{i=1}^{N} y_i d_i \underset{H_0}{\overset{H_1}{\gtrless}} 0 \tag{17}$$

which is the correlation detection algorithm used by Zeng and Liu [14]. Therefore, we can see that both watermark detectors of Hernández's and Zeng's are special cases of the proposed DCT domain watermark detection framework.

B. HVS in the DCT Domain

In the DCT domain watermarking, there are two major characteristics of the HVS, *frequency selectivity* and *contrast sensitivity* [16].

In the DCT domain, *frequency selectivity* characteristic allows high frequency DCT coefficients to have greater distortion without being perceived. This characteristic is also adopted in JPEG compression. JPEG uses Table 1 as the default quantization table that allows coarser quantization in higher frequency DCT coefficients. Therefore, we can use Table 1 as a reference to design the watermark amplitude, by allowing the watermark amplitude at each frequency to be proportional to the entry value in Table 1 at the corresponding frequency.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

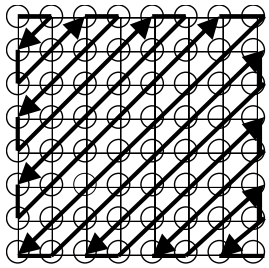
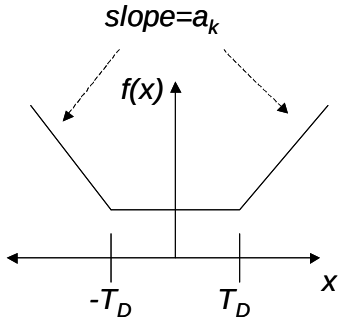


Table 1. THE DEFAULT JPEG QUANTIZATION TABLE

The *contrast sensitivity* characteristic states that the larger the absolute value of the DCT coefficient, the more the DCT coefficient can be altered without being perceived. The relationship between the DCT coefficient and the watermark amplitude is as follows. If the embedder considers embedding a watermark bit in a DCT coefficient with value x, the watermark amplitude w is proportional to the absolute value of x. If the absolute value of the DCT coefficient is too small, we can still alter the DCT coefficient to a fixed amount while keeping the watermark imperceptible, i.e., $f_k(x) = a_k \cdot \max(T_D, |x|)$, where a_k is the proportionality multiplier, T_D is the fixed amplitude used when the DCT coefficient is too small, and k is the frequency index in the zigzag order as in Table 1. Including the frequency selectivity factor

described in the last paragraph, that is, a_k being proportional to the entry in Table 1, the watermark amplitude $f_k(x)$ becomes:



$$f_k(x) = a_k \cdot \max(T_D, |x|) = e_k \alpha_D \cdot \max(T_D, |x|) \qquad (18)$$

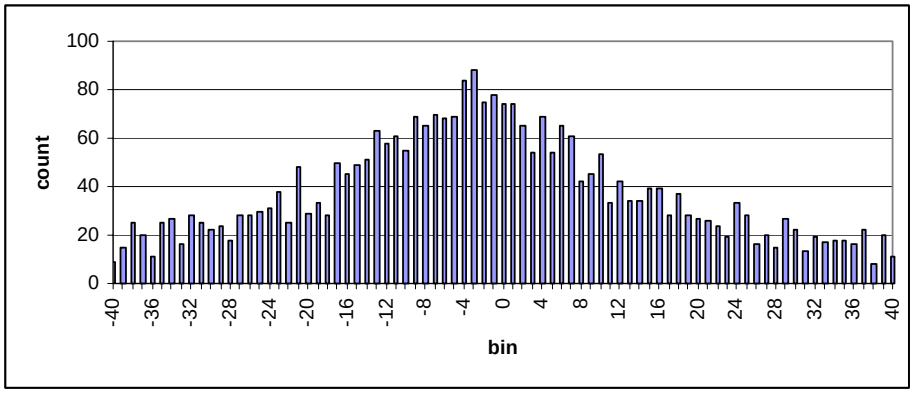
where e_k is the entry in Table 1. Applications that allow more image distortion can use larger values of T_D and α_D. Note that the choice of a_k should not be greater than one since the one-to-one and onto conditions of the functions $g_{i,1}(x)$ and $g_{i,0}(x)$ would be violated. In addition to the HVS model we presented, other DCT domain HVS models can be incorporated into the proposed framework as well.

C. Experiments

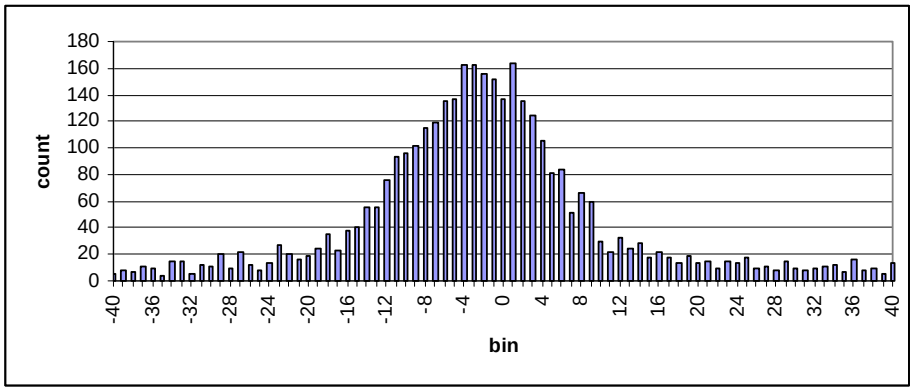
We compare three watermark detection approaches—Hernández's [19] as in (16), Zeng's [14] as in (17), and the proposed as in (15). Watermark embedding and detection are performed in 8×8 blocks of the original and watermarked images respectively. These three approaches use the same watermarked images generated by (2) and (18) to test their detection performance. We use $T_D = 10$ and $\alpha_D = 0.025$ in the experiments. The watermarked images have an average PSNR of 32.2dB. With $\alpha_D = 0.025$, we can use DCT coefficients up to the 14th ac coefficient to satisfy the one-to-one and onto conditions. As to detector parameter estimation, they all use the received watermarked image to get the statistics of the noise. In addition to estimating the noise statistics, we also need to estimate d_i in (16) and (17) for Hernández's and Zeng's detectors. We use $f_k(y_i)$ as the estimate of d_i.

The test images, both the original and the watermark, are the same as in Section III. Each 4-bit pixel of the watermark is embedded in 64 pseudo-randomly selected DCT blocks of the original image that form a group. These groups of blocks are non-overlapping with each other. We use the first 10 ac coefficients of each DCT block to embed the watermark. At each DCT block, the most significant bit (MSB) of the watermark pixel is embedded in the first ac coefficient; the second MSB is embedded in the second ac coefficient, …, and the least significant bit (LSB) is embedded in the fourth ac coefficient; the MSB is again embedded in the fifth ac coefficients, and so on until the second MSB is embedded the third time in the 10^{th} ac coefficients. Each first or second MSB of the watermark is embedded for $N = 64 \times 3$ times while each third MSB or LSB is embedded for $N = 64 \times 2$ times.

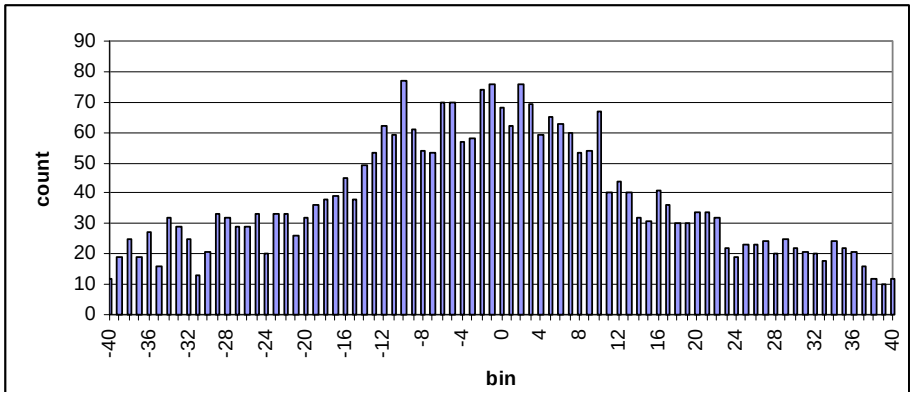
In Figure 7, histograms of the 5^{th} ac coefficients of the received watermarked images are shown. It is clear from the figure that the generalized Gaussian distribution can well model the pdf of the DCT coefficients of the received watermarked images.



(a)



(b)



(c)

Figure 7. HISTOGRAM OF THE 5th AC COEFFICIENTS OF IMAGES: (a) BRIDGE; (b) LENA; (c) MANDRILL

At each frequency, we use different generalized Gaussian distribution parameters to describe the pdf of the DCT coefficients. The parameters include standard deviation σ and c, assuming all ac coefficients

have mean zero. With 10 ac coefficients used for watermarking, we need to estimate 20 parameters in total. Detailed algorithm used to obtain those parameters is shown in Section VII-B.

We show the performance of the watermark detectors in Figure 8 and Figure 9. In Figure 8, we show the BER of the retrieved watermark with respect to the original watermark at various quality factors of JPEG compression. In Figure 9, we show the PSNR of the retrieved watermark with respect to the original watermark at various quality factors. It is shown in Figure 8 and Figure 9 that the proposed DCT domain watermark detector performs the best among the three approaches since we have the best modeling of the pdf of the DCT coefficients. Compared Hernández's with Zeng's, the former is better in the BER performance since Hernández's approach has the generalized Gaussian modeling instead of the simple Gaussian modeling (but does not have the terms $J_{i,1}(\cdot)$ and $J_{i,0}(\cdot)$ as we do).

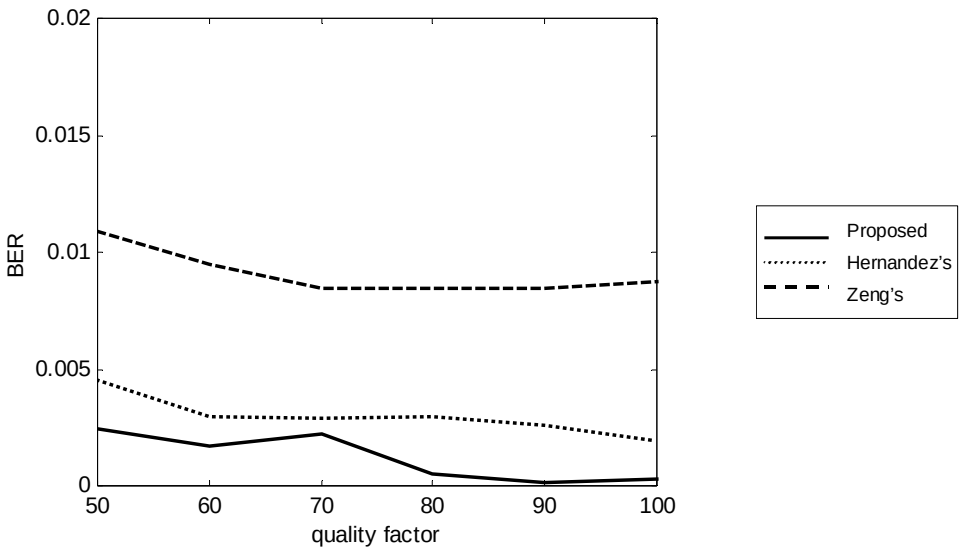


Figure 8. BER PERFORMANCE OF THREE DCT DOMAIN WATERMARK APPROACHES AT VARIOUS QUALITY FACTORS OF COMPRESSION

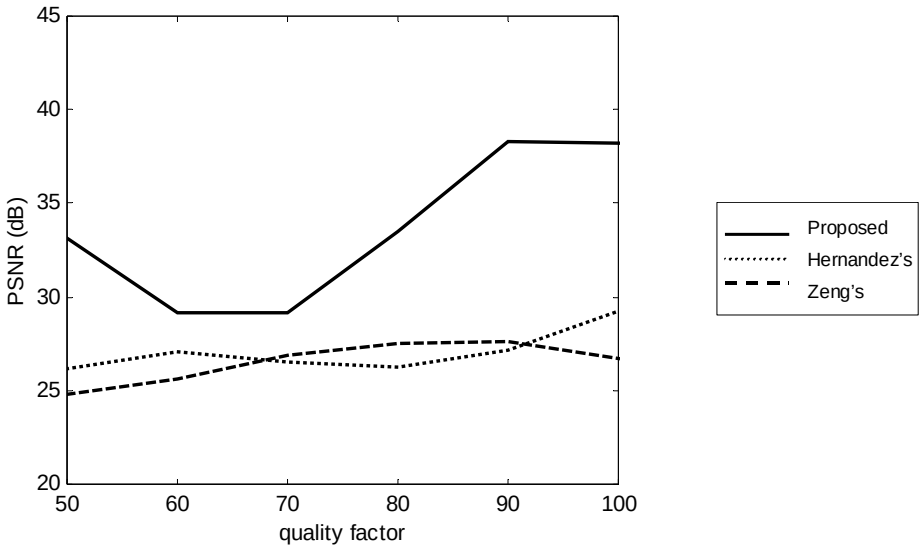


Figure 9. PSNR PERFORMANCE OF THREE DCT DOMAIN WATERMARK APPROACHES AT VARIOUS QUALITY FACTORS OF COMPRESSION

We also show in Figure 10 that it is reasonable to use the corrupted watermarked image to estimate the pdf of the noise (with original image and compression artifacts). It is shown that the BER performances using the corrupted watermarked image and using the real noise to estimate the pdf of the noise are comparable. Thus, we use the received watermarked image to estimate the pdf of the noise for (15). Note that this experiment is only to validate the use of received watermarked image to estimate the pdf of the noise. The proposed watermark detector does not use the real noise since the real noise is not available to the watermark detector.

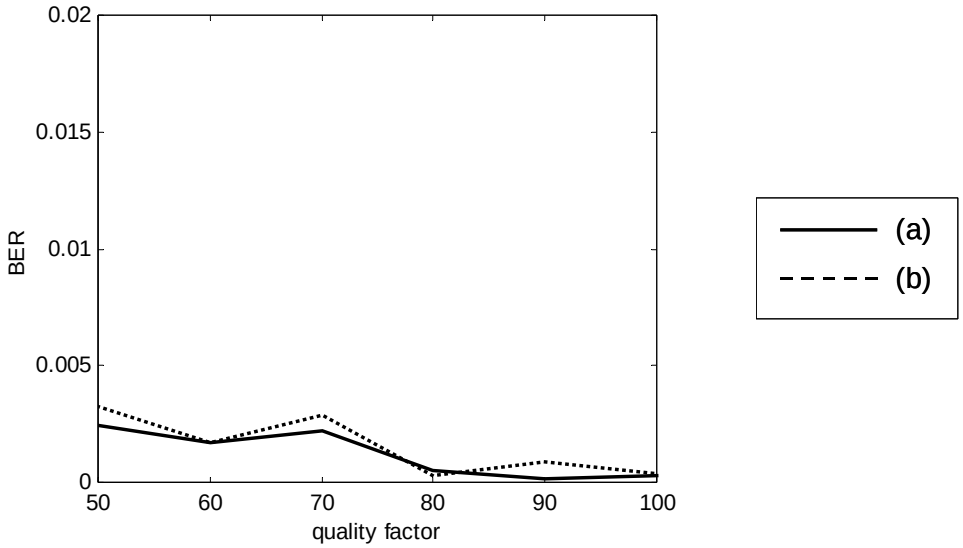


Figure 10. BER PERFORMANCE OF DCT DOMAIN WATERMARK DETECTORS USING (a) THE CORRUPTED WATERMARKED IMAGE AND (b) THE REAL NOISE

Comparing the proposed pixel domain watermark and DCT domain watermark methods (Figure 11), we found that the DCT domain watermark method has better performance when the compression is more serious and a little worse when the compression is less serious. This shows that the transform domain watermark methods such as the DCT domain watermark method are in general more robust than the pixel domain watermark methods.

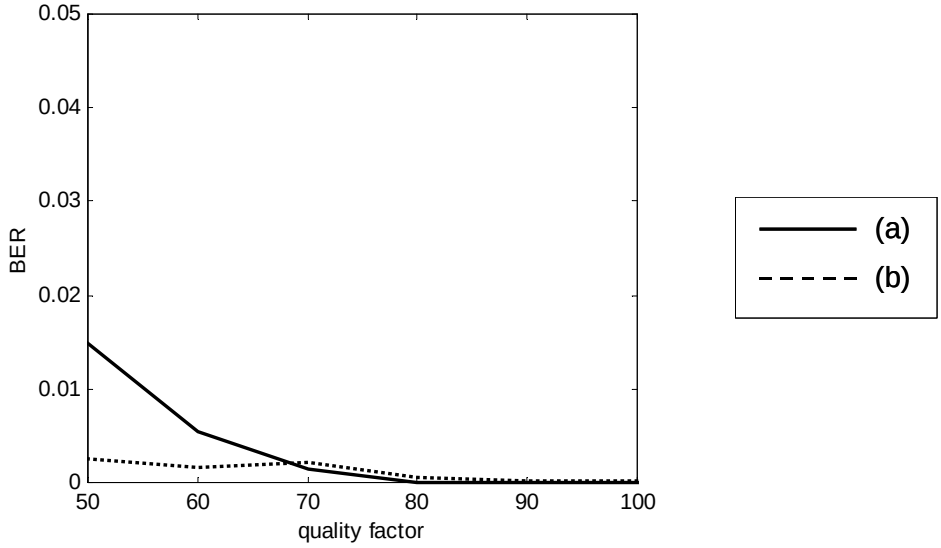


Figure 11. BER PERFORMANCE COMPARISON OF THE PROPOSED WATERMARK DETECTORS: (a) PIXEL DOMAIN WATERMARK; (b) DCT DOMAIN WATERMARK

V. CONCLUSION

In this paper, we provided a general framework for public watermark detection and presented an optimal watermark detector in this framework. Other watermark detectors were shown to be special cases of this framework. The proposed watermark detector models the pdf of the noise, which comes from the original image and perhaps some distortions, better than other detectors.

To show that better modeling of the noise results in better watermark detection performance, we presented the results in both the pixel domain and the DCT domain. In the pixel domain watermark system, we modeled the pdf of the noise as a Gaussian mixture with a well-chosen number of components. In the DCT domain watermark system, we modeled the pdf of the noise as a generalized Gaussian distribution. It was shown that the proposed watermark detectors in both the pixel domain and the DCT domain have better performance in terms of the retrieved watermark BER and PSNR. Comparing the proposed watermark methods in the pixel domain and the DCT domain, we found the proposed DCT domain watermark method performs better in general. This complies with the argument that the transform domain watermark methods are more robust than the pixel domain watermark methods.

Even though better modeling of the pdf of the noise can improve the watermark detection performance, it is also important that we should not over fit the histogram of the received watermarked image to obtain the pdf of the noise. In both the proposed pixel domain and DCT domain watermarking experiments, the detection performance is the best when the number of parameters is between 10 and 20. If the number of parameters is too big, the pdf modeling over fits the received watermarked image. On the other hand, if the number of parameters is too small, the pdf modeling cannot describe the pdf of the noise precisely enough either.

To conclude, the success of the proposed optimal watermark detector results from two considerations: 1) the detection framework considers the watermark designed based on the human visual system and 2) the watermark detector has better modeling of the probability distribution of the noise.

Our future work includes better modeling of the noise in other domains, such as the discrete wavelet transform domain. In addition, we want to improve the modeling of the pdf with side information. Further, we can extend the concept of better modeling of the pdf of the noise to other watermark problems. One example is the one-bit watermark detection problem in which one tries to, instead of retrieving the watermark, determine the presence of a preset watermark. Existing solutions assume the noise to be Gaussian [25]. By better modeling the noise, we will be able to determine the presence of the watermark with higher reliability. Another example is the watermark capacity problem in which one tries to estimate how many watermark bits can be embedded into an original image. Conventional analysis assumes the noise to be Gaussian when calculating the capacity of the original image [26]. The work done in [27] suggests an iterative method to find the optimal watermark attack to calculate the image capacity for watermarking. With better modeling of the noise and the attacks, we will be able to estimate the capacity more precisely.

VI. REFERENCE

[1] M. Wu, E. Tang, and B. Liu, "Data Hiding in Digital Binary Image". *Proceedings of the IEEE Intl. Conf. on Multimedia and Expo. (ICME'00),* New York, July 2000.

[2] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking Digital Image and Video Data". *IEEE Signal Processing Magazine*, September 2000, 20-46.

[3] F. Hartung, and M. Kutter, "Multimedia Watermarking Techniques". *Proceedings of the IEEE*, 87(7), 1079-1107.

[4] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual Watermarks for Digital Images and Video". *Proceedings of the IEEE*, 87(7), 1108-1126.

[5] I. J. Cox, M. L. Miller, and A. L. McKellips, "Watermarking as Communications with Side Information". *Proceedings of the IEEE*, 87(7), 1127-1141.

[6] J. R. Hernández and F. Pérez-González, "Statistical Analysis of Watermarking Schemes for Copyright Protection of Images". *Proceedings of the IEEE*, 87(7), 1142-1166.

[7] G. Voyatzis and I. Pitas, "The Use of Watermarks in the Protection of Digital Multimedia Products". *Proceedings of the IEEE*, 87(7), 1197-1207.

[8] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia Data-Embedding and Watermarking Techniques". *Proceedings of the IEEE*, 86(6), 1064-1087.

[9] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia". *IEEE Transactions on Image Processing*, 6(12), 1997, 1673-1687.

[10] A. Z. Tirkel, R. G. Schyndel, and C. F. Osborne, "A Digital Watermark". *Proceeding of the ICIP'94*, 1994, 86–90.

[11] T. Kalker, A.J.E.M. Janssen, "Analysis of Watermark Detection using SPOMF". *Proceedings of the ICIP'99*, 1999, 316-319.

[12] M. Barni, F. Bartolini, V. Cappellini, A. Piva, "A DCT-domain System for Robust Image Watermarking". *Signal Processing*, 66, 1998, 357-372.

[13] J. J. Eggers, J. K. Su; B. Girod, "Robustness of a Blind Image Watermarking Scheme". *Proceedings of the ICIP2000*, 17-20.

[14] W. Zeng and B. Liu, "A Statistical Watermark Detection Technique Without Using Original Images for Resolving Rightful Ownerships of Digital Images". *IEEE Transactions on Image Processing*, 8(11), 1534-1548.

[15] J. R. Hernández, F. Pérez-González, J. M. Rodriguez, and G. Nieto, "Performance Analysis of a 2-D Multipulse Amplitude Modulation Scheme for Data Hiding and Watermarking of Still Images". *IEEE Journal on Selected Areas in Communications*, 16(4), 510-524.

[16] Andrew B. Watson, Digital Images and Human Vision, Massachusetts Institute of Technology 1993.

[17] E. Sayrol, J. Vidal, S. Cabanillas, and S. Santamaria, "Optimum Watermark Detection in Color Images". *Proceedings of the ICIP1999*, 231-235.

[18] J. Vidal and E. Sayrol, "Optimum Watermark Detection and Embedding in Digital Images". *Proceedings of the ICIP1999*, 285-290.

[19] J. R. Hernández, M. Amado, and F. Pérez-González, "DCT-Domain Watermarking Techniques for Sill Images: Detector Performance Analysis and a New Structure". *IEEE Transactions on Image Processing,* 9(1), Jan. 2000, 55-68.

[20] John G. Proakis, Digital Communications, McGraw-Hill, 3rd edition, 1995.

[21] Anil K. Jain, Fundamentals of Digital Image Processing, Prentice Hall.

[22] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. R. Stat. Soc. Lond*, 39, 1977, 1-38.

[23] Thomas M. Cover, and Joy A. Thomas, Elements of Information Theory, John Wiley & Sons, Inc. 1991.

[24] Bruce Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, John Wiley & Sons, Inc. 1995.

[25] Jean-Paul Linnartz, Geert Depovere, and Ton Kalker, "On the Design of a Watermarking System: Considerations and Rationales". *Proceedings of the Information Hiding Workshop*, 1999.

[26] J. Smith , and B. Comiskey, "Modulation and Information Hiding in Images". *Proceeding of the 1996 First International Workshop on Information Hiding*, Cambridge, U.K., 1996, 207-226.

[27] J. K. Su, J. J. Eggers and B. Girod, "Analysis of Digital Watermarks Subjected to Optimum Linear Filtering and Additive Noise". *Signal Processing*, 81(6), 2001, 1141-1176.

[28] W.H. Press, S. A. Teukolsky, W.T. Vetterling, and B. P. Flannery, Numerical Recipes in C, Cambridge University Press, 2nd Ed.

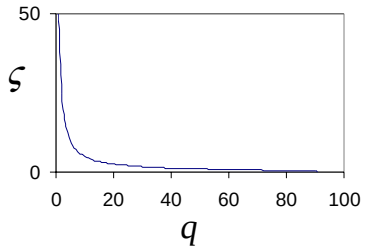
[29] Independent JPEG Group website. http://www.ijg.org/

VII. APPENDIX

A. Quality Factor of JPEG Compression

The JPEG quality factor is not standardized. We use the definition used by Independent JPEG Group (IJG) [29]. The input of the quality factor is between 0 and 100. The larger the number, the smaller the

quantization level and hence the better the quality of the quantized image. The quality factor q is first converted to a scale factor ς as follows:



$$\varsigma = \begin{cases} 50 & if \quad q < 1 \\ 50/q & if \quad 1 \le q < 50 \\ 2 - 0.02 \cdot q & if \quad 50 \le q \le 100 \\ 0 & if \quad q > 100 \end{cases} \tag{19}$$

Each entry of the default quantization table (Table 1) is multiplied by the resulting scale factor ς to form the quantization table at quality factor q.

B. Finding the Parameter of a Generalized Gaussian Distribution

We can find the parameter c of a generalized Gaussian distribution by curve-fitting the histogram of the DCT coefficients of an image. If the histogram is constructed at each frequency, we can compute a parameter that is frequency dependent; otherwise, we can compute a single parameter with no frequency dependency.

Let the histogram of the DCT coefficients be denoted as $h(y)$, where y is the value of the DCT coefficient. We can use the following formula to get the value c.

$$\Phi = \left(Y^t Y\right)^{-1} Y^t Z \tag{20}$$

where $\Phi = \begin{bmatrix} c \\ \omega \end{bmatrix}$, $Y = \begin{bmatrix} \ln|y_1| & 1 \\ \ln|y_2| & 1 \\ \vdots & 1 \\ \ln|y_m| & 1 \end{bmatrix}$, and $Z = \begin{bmatrix} \ln(-\ln(h'(y_1))) \\ \ln(-\ln(h'(y_2))) \\ \vdots \\ \ln(-\ln(h'(y_m))) \end{bmatrix}$, where $h'(y) = h(y)/h(0)$. Each index, $i = 1 \sim m$, refers to one sample of the histogram. Other ways of computing the value of c can be found in [19].