Histogram Preserving QIM-Based Watermarking

Chang-Tsun Li[1] and *Yue Li*[2]

[1]Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK
c-t.li@warwick.ac.uk

[2]College of Software, Nankai University, Tianjin, China
liyue80@nankai.edu.cn

Abstract

Quantisation-based watermarking schemes are known to be strongly robust to noise and widely applied for image authentication. However, such schemes tend to create specific patterns in the histogram of the watermarked image, thus opening security gap for histogram analysis and attack. As such, a secure quantisation-based watermarking scheme should be *histogram preserving*. That is to say the original statistical characteristics of the histogram should be preserved in the watermarked image. In this paper, we propose a histogram preserving quantisation index modulation (HPQIM)-based watermarking, which can preserve the original statistical characteristics of the histogram of the original image after watermark embedding. This scheme originates from quantisation index modulation (QIM)-based watermarking scheme, but highly increases the security of QIM-based schemes. Meanwhile, the proposed HPQIM-based watermarking scheme can also achieve a lower embedding distortion than the QIM-based watermarking schemes.

1. Background of QIM-Based Watermarking Schemes

Data embedding is an enabling technique for digital watermarking [1, 2, ctli 2005, ctli 2006, 27] and steganography [10, 20]. Among the many schemes reported in

the literature, Quantisation Index Modulation (QIM) – based data embedding can strike good balance among embedding capacity, robustness and embedding distortion [3, 4, 5, 9, 12, 16, 19, 24]. The general idea of the QIM-based watermarking scheme is to apply two complementary quantisation functions, Q_0 and Q_1, each with a unique set of indices/quantisers, to quantise the signal. To embed a bit of watermark message (either '0' or '1'), one of two quantisation functions is selected according to the value of the watermark bit for quantising the signal. Since there is no intersection between the sets of the output values of the two quantisation functions, the recipient can unambiguously extract the watermark by identifying the index of output set that the quantised value belongs to. The mathematical models for QIM-based watermark embedding and extraction are as following:

Embedding: $$x = \begin{cases} Q_0(s) & , \text{ if } w = 0 \\ Q_1(s) & , \text{ if } w = 1 \end{cases} \quad (1)$$

Extraction: $$w' = \begin{cases} 0 & , \text{ if } x \in Q_0(s) \\ 1 & , \text{ if } x \in Q_1(s) \end{cases} \quad (2)$$

where Q_0 and Q_1 are the two quantisation functions, s and x are the input and output of the quantisation functions on the embedding side, w and w' are the watermark embedded in and extracted from the signal s, respectively. The most commonly used QIM techniques are called scalar QIM [3,4], which quantise the signal s with a uniform quantisation step in the 1-dimension space. The two quantisation functions, Q_0 and Q_1, modify the values to the nearest odd/even

multiples of the quantisation steps. To achieve low distortion, a Distortion-Compensation QIM (DC-QIM) method [3,4] has been developed, with the quantisation function formulated as follows:

$$\text{Embedding:} \qquad x = \begin{cases} Q_0(\alpha s) + (1-\alpha)s & , \text{ if } w = 0 \\ Q_1(\alpha s) + (1-\alpha)s & , \text{ if } w = 1 \end{cases} \tag{3}$$

$$\text{Extraction:} \qquad w' = \begin{cases} 0 & , \text{ if } (x-(1-\alpha)s \in Q_0(s) \\ 1 & , \text{ if } (x-(1-\alpha)s \in Q_1(s) \end{cases} \tag{4}$$

where α is the parameter for distortion compensation. In Equation (3), a smaller α indicates a strong distortion compensation and vice verse. We can see that the traditional scalar QIM functions (i.e, Equation (1) and (2)) are special cases of CD-QIM functions when α of Equation (3) and (4) is set to 1. That mean no distortion is compensated by the scheme. When $\alpha = 0$, $x = s$, watermark embedding does not take place, thus no distortion is introduced (i.e. the distortion is compensated in full).

QIM-based watermarking techniques are frequently applied in conjunction with image compression techniques, such as JPEG [8] and JPEG2000 [26]. JPEG is a widely used image compression technique in the DCT domain while JPEG2000 is a more recently developed image compression technique in the wavelet transform domain. Ho *et al*. [9] developed a watermarking technique by applying two quantisation functions, Q_0 and Q_1 to quantise the DCT coefficients of images for embedding the watermark. Lin and Chang [16] proposed a technique, which randomly selects DCT coefficients and quantising them to either odd or even

multiples of a JPEG-50 quantisation step based on the value of the watermark, where 50 is quality factor of JPEG compression. Fridrich et al. [5] also use the QIM-based watermarking scheme to embed a hashed signature of the image into one third of the DCT coefficients after JPEG quantisation. Kundur and Hatzinakos [12] integrate the QIM-based watermarking into JPEG2000, and quantise the discrete wavelet transform (DWT) coefficients based on the watermark. More recent applications can be found in [2, 15, 21, 22**]**

One of the important factors is the undetectability of the hidden data [1, 11, 18, 19, 23]. However, QIM-based watermarking schemes have been proved to be vulnerable to histogram analysis [6, 18, 19] because they tend to create abnormal and noticeable patterns in the histograms of watermarked images, which draw the attacker's attention. Figure 1(a) and (c) demonstrate the original image of Lena and its QIM watermarked version generated with Equation (1) and a quantisation step of 4. Figure 1(b) and (d) illustrate the histograms of the original image of Lena and the watermarked version. From the histogram of the watermarked Lena image in Figure 1(d), we can see that the frequencies of the occurrence of most intensity values become zero, leaving gaps between quantisers. The gaps in the histogram are caused by the quantisation operation formulated in Equation (1). As a result, the attacker can detect the quantisation functions from the histogram, as demonstrated in Figure 1(d). Knowing the quantisation functions, the attacker can then extract the embedded watermark or destroy it [24]. In scalar QIM, if the odd multiples of the quantisation steps represent watermark bit '0', then the even multiples of the quantisation steps

represent ‘1’ and vice verse. Thus by investigating this mapping relationship between the multiples of quantisation steps and the watermark bit, the steganalyst can set up two matching tables. In one table, the odd multiples of the quantisation steps represent watermark ‘0’ and even multiples of the quantisation steps represent watermark ‘1’, while in another, the opposite applies. Since one of the two matching tables is the correct one used in QIM-based watermarking, therefore the steganalyst can extract two watermarks by using the two matching tables, knowing that one of them is the embedded one. Furthermore, the steganalyst can easily destroy the embedded watermark by randomly changing the quantised values in the watermarked image to neighbouring values in the histogram.

As proved by Vila-Forcen et al. [25], the gaps also exist in the histogram of the DC-QIM watermarked signal, as demonstrated in Figure 2. Therefore DC-QIM-based watermarking schemes are also insecure against histogram analysis. In Figure 2, a non-uniform quantisation step sequence controlled by the parameter ρ is applied. The shaded areas cover the intensity values that are present in the DC-QIM watermarked signal. The greater the value of α of Equation (3) is, the narrower the shaded areas. The white dots represent the quantisers and the black dots represent the centres of dead zones. In the histogram, a dead zone is an area in which intensity values will never appear in the watermarked signal. According to Vila-Forcen, the dead zone ensures the robustness and low extraction error rate in DC-QIM. A wider dead zone leads to greater robustness against noise and lower extraction error rate. Conversely, a narrower dead zone results in weaker robustness and higher extraction

error rate. However, due to the existence of the dead zones, the gaps, which attract attacker's attention, still exist in the histogram of DC-QIM watermarked images. As a result, by observing the gaps in the histogram, the attacker can identify the quantisers at the centroid of the shaded areas separated by the gaps. Knowing the quantisers, the attacker can either extract the embedded watermark by using the look-up tables or destroy the watermark by randomly changing the quantised values to the values in the neighbouring shaded areas. Hence, a key security requirement of the QIM- and DC-QIM-based schemes is to fill the gaps in the histogram. Furthermore, as discussed in [17, 24], the histogram of the watermarked image should not follow any particular distribution, in order to prevent the attacker from analysing the statistical features of the watermarking schemes. Therefore, a secure QIM-based watermarking scheme should be histogram preserving. In this paper, we propose an *adaptive QIM*-based watermarking scheme, *HPQIM-based watermarking*, which preserves the statistical characteristics of the original histogram of the image after embedding the watermark, in order to enhance the security of the watermarking scheme.

2. HPQIM-Based Watermarking Scheme

In this section, we propose the HPQIM-based watermarking scheme. In Section 2.1 we propose the algorithm of generating an adaptive random integer number sequence for masking the gaps created by traditional QIM-based embedding schemes in the histogram and the watermark embedding and extraction algorithms. In Section 2.2, we will discuss the special rules for embedding, extraction and random integer

number generation, when the random number in the sequence happens to be half of the quantisation step.

2.1. HPQIM Embedding and Extraction Algorithms

The proposed HPQIM-based watermarking scheme aims to achieve the following three objectives:

- The gaps in the histogram created by QIM embedding must be filled;
- The statistical characteristic of the histogram of the original image should be preserved in the stego version;
- Lower embedding distortion than QIM and DC-QIM should be achieved.

We achieve these objectives by embedding the secret data according to a random integer number sequence R, which is adaptively generated according to the local probability density function (pdf) of the occurrences of intensity values under the control of a secret key shared by the embedding and extraction sides. The dynamic range of the random integer numbers is $[0, \Delta\text{-}1]$, where Δ is the integer-valued quantisation step. The random integer number sequence is of the same size as the image so that every pixel in the image is assigned a random integer number and the pixel can be adaptively modified according to this random number after quantisation. The basic idea is as follows. Given the original image I of $X \times Y$ pixels, the secret key K and the quantisation step Δ, the histogram H of the original image I is created and divided into S segments, with each segment spanning Δ intensity levels. Then for each bin/intensity i, its corresponding *within-segment probability* p_i is calculated

according to the following formula

$$p_i = \frac{h_i}{\sum_{k=s\cdot\Delta}^{(s+1)\cdot\Delta-1} h_k} \tag{5}$$

where h_i is the number of the occurrences of intensity value i in I and s ($s \in [0, S\text{-}1]$) is the index of the segment, to which i belongs. The reason we call p_i *within-segment probability* is because it indicates intensity i's occurring frequency *locally* relative to the histogram segment to which it belongs, rather than *globally* relative to the entire histogram. Now let $I_{x,y}$ be the intensity of the (x, y)th pixel of image I. The random integer number sequence R, $R = \{R_{1,1}, R_{1,2}, ..., R_{1,Y}, ..., R_{X,1}, R_{X,2}, ..., R_{X,Y}\}$, $0 \le R_{x,y} \le \Delta$ is generated as follows. For each pixel (x, y), find s such that $s \cdot \Delta \le I_{x,y} \le (s+1) \cdot \Delta$ and generate $R_{x,y}$ according to the Monte Carlo algorithm [7] to ensure that the distribution of $R_{x,y}$ follows the probability of $\{p_{s\cdot\Delta}, p_{s\cdot\Delta+1}, p_{s\cdot\Delta+2}, \cdots, p_{(s+1)\cdot\Delta-1}\}$ calculated according to Eq. (5). That is to say the probability that $R_{x,y} = 0$ is $p_{s\cdot\Delta}$, $R_{x,y} = 1$ is $p_{s\cdot\Delta+1}$, and so on. This random integer number sequence, R, is then used for modifying the quantised values during the embedding process, aiming to cover the gap in the histogram. Let $w, w = \{w_{x,y} \mid x \in [0, X-1], y \in [0, Y-1]\}$, denote the binary secret data sequence of the same size as the original image I. In the embedding process, each secret bit $w_{x,y}$ is first involved in the quantisation of the corresponding pixel $I_{x,y}$ using the QIM method, as formulated in Eq. (1), to produce $I^q_{x,y}$. Then the quantised value $I^q_{x,y}$ is modified by either adding $R_{x,y}$ or subtracting $R_{x,y}$, depending on which operation produces a result closer to the original value $I_{x,y}$, in order to produce the final stego-image $I'_{x,y}$. That is

to say

$$I'_{x,y} = \begin{cases} I^q_{x,y} + R_{x,y} & \text{, if } \left|I_{x,y} - (I^q_{x,y} + R_{x,y})\right| < \left|I_{x,y} - (I^q_{x,y} - R_{x,y})\right| \\ I^q_{x,y} - R_{x,y} & \text{, otherwise} \end{cases} \quad (6)$$

The data embedding procedures are listed in Algorithm 1. Since *R* follows the pdf of the histogram of the original image, the adaptive modification leads to a preserved histogram after HPQIM embedding. Meanwhile, the adaptive modification also achieves a lower embedding distortion because, in addition to the quantisers, intensity values other than those quantisers are allowed in the watermarked version.

When the stego-image is received, the recipient needs to regenerate the same random integer number sequence *R* according to the histogram of the stego-mage I'. For each pixel $I'_{x,y}$, $I'_{x,y}+R_{x,y}$ and $I'_{x,y}-R_{x,y}$ are compared with both $Q_i\left(I'_{x,y}\right), i \in \{0,1\}$, respectively, according to the following equation

$$D_i = \min\left\{\left|(I'_{x,y}-R_{x,y}) - Q_i(I'_{x,y})\right|, \left|(I'_{x,y}+R_{x,y}) - Q_i(I'_{x,y})\right|\right\}, \quad (7)$$

$i \in \{0,1\}$ and Q_i is defined in Eq. (1). If $D_1 < D_0$, 1 is deemed as the hidden bit, i.e., $w_{x,y} = 1$. Otherwise, if $D_1 > D_0$, $w_{x,y} = 0$. This is because on the embedding side, $I_{x,y}$ is quantised to the closest quantiser $I^q_{x,y}$ using Eq. (1) and $I^q_{x,y}$ is further modified to $I'_{x,y}$ that minimise its distance from $I_{x,y}$ according to Eq. (6). Note that the case when $D_1 = D_0$ needs special attention as we will discuss later. We will explain how to deal with this special case in Section 2.2. The secret data extraction procedures are

listed in Algorithm 2.

Figure 3 demonstrates an example of the working of the HPQIM algorithm. Assuming the two sets of quantisers are $Q_0 = \{0,8,16,\ldots,248\}$ and $Q_1 = \{4,12,20,\ldots,252\}$ with the quantisation step Δ equal to 4, the original intensity value of the pixel $I_{x,y}$ is 7, secret bit $w_{x,y}$ '0' is to be embedded into the pixel and the corresponding random number $R_{x,y}$ is 3. To embed '0', the original intensity value, 7 ($I_{x,y}$), is quantised to 8 ($I^q_{x,y}$) because it is the nearest quantiser in Q_0. This quantisation is demonstrated in Figure 3(a). After the modification, the output $I'_{x,y}$ is either $Q_0(I_{x,y}) - R_{x,y}$ = 8 - 3 = **5** or $Q_0(I_{x,y}) + R_{x,y}$ = 8 + 3 = **11**. Since $|5-7| < |11-7|$, changing the original intensity value 7 ($I_{x,y}$) to 5 inflicts lower distortion, so the output $I'_{x,y}$ is set to 5. This adaptive modification process is demonstrated in Figure 3(b). Given the same secret key as that used in the embedding side, when the recipient obtains the stego-image I', he/she is able to figure out that $I'_{x,y}$ with an intensity value of 5 has a corresponding random number of $R_{x,y}$ equal to 3, and $I'^1_{x,y} = I'_{x,y} - R_{x,y}$ = 5 - 3 = **2** and $I'^2_{x,y} = I'_{x,y} + R_{x,y}$ = 5 + 3 = **8**. Now to determine whether $I'_{x,y}$ is carrying '0' or '1', both quantiser sets, $Q_0 = \{0,8,16,\ldots,248\}$ and $Q_1 = \{4,12,20,\ldots,252\}$, are searched. The quantiser in Q_0 nearest to 5 (i.e., $I'_{x,y}$) is 8, so

$$D_0 = \min\left\{\left|(I'_{x,y} - R_{x,y}) - Q_0(I'_{x,y})\right|, \left|(I'_{x,y} + R_{x,y}) - Q_0(I'_{x,y})\right|\right\} = \min\{|2-8|, (|8-8|)\} = 0$$

and the quantiser in Q_1 nearest to 5 is 4, so

$$D_1 = \min\left\{\left|(I'_{x,y} - R_{x,y}) - Q_1(I'_{x,y})\right|, \left|(I'_{x,y} + R_{x,y}) - Q_1(I'_{x,y})\right|\right\} = \min\{|2-4|, (|8-4|)\} = 2$$

Since $D_0 < D_1$, the recipient knows that the embedded secret is '0'. This extraction

process is demonstrated in Figure 3(c).

2.2. Special Rules for Embedding, Extraction and Random Sequence Generation

The algorithms presented in Section 2.1 are for the general cases. According to the analysis of HPQIM, a special case, where in the random number $R_{x,y}$ is half of the quantisation step (i.e., $R_{x,y} = \Delta/2$), may cause ambiguity, thus entailing special care. We discuss this special case in the following section.

2.2.1. Special Rules for Embedding & Extraction

Unfortunately, a special situation when $R_{x,y}$ = Δ/2 will lead to an ambiguity as explained below. Using the same example in Figure 3 except that $R_{x,y}$ = Δ/2 = 2, $I_{x,y}$ = 7 is quantised to 8 (i.e., $I^q_{x,y}$), then adaptively modified to 6 (i.e., the final watermarked version $I'_{x,y} = I^q_{x,y} - R_{x,y} = 8 - 2 = 6$). When the recipient receives the image with the pixel value $I'_{x,y}$ = 6, and he/she calculates $Q_0(6) = 8$ and $Q_1(6) = 4$ and regenerates the random number $R_{x,y}$ (= 2) using the shared secret key K and. According to Equation (7) and (8), he/she gets

$$\begin{aligned} D_0 &= \min\left\{\left|(I'_{x,y} - R_{x,y}) - Q_0(I'_{x,y})\right|, \left|(I'_{x,y} - R_{x,y}) + Q_0(I'_{x,y})\right|\right\} \\ &= \min\left\{|6-2-8|, |6+2-8|\right\} = 0 \end{aligned}$$

and

$$\begin{aligned} D_1 &= \min\left\{\left|(I'_{x,y} - R_{x,y}) - Q_1(I'_{x,y})\right|, \left|(I'_{x,y} + R_{x,y}) - Q_1(I'_{x,y})\right|\right\} \\ &= \min\left\{|6-2-4|, |6+2-4|\right\} = 0 \end{aligned}$$

This leads to an ambiguous case since $D_0 = D_1$. This problem arises only when the random number $R_{x,y} = \Delta/2$. The ambiguity may occur when embedding 0 and 1

when the special case is encountered, however we only need to deal with case associated with either value. The way we deal with this special case is as follows. When this special situation is encountered, if the watermark bit $w_{x,y}$ = 0, then the intensity value is not further changed after quantisation, i.e., we let $I'_{x,y} = I^q_{x,y}$. By doing so on the embedding side, when the extraction side encounter the special case that $R_{x,y} = \Delta/2$ and $I'_{x,y} = Q_0(I'_{x,y})$, the algorithm will know that the embedded watermark bit is 0. On the other hand, when the extraction side encounters the case that $R_{x,y} = \Delta/2$ and $I'_{x,y} \neq Q_0(I'_{x,y})$, 1 will be taken as the embedded watermark bit. The complete data embedding and extraction algorithms are as follows.

Algorithm 1. Data embedding algorithm for HPQIM

Input: original image I, secret data w

Output: stego- image I'

1. Generate the random integer number sequence R using Algorithm 3 under the control of a secret key K.
2. For each input pixel $I_{x,y}$,

   2.1. Generate its quantised counterpart $I^q_{x,y}$ according to Eq. (1) and $w_{x,y}$.

   2.2. If $R_{x,y} = \Delta/2$ and the watermark $w_{x,y}$ = 0 (the special case)

$$I'_{x,y} = I^q_{x,y}$$

   else

$$I'_{x,y} = \begin{cases} I^q_{x,y} + R_{x,y} & \text{, if } \left|I_{x,y} - (I^q_{x,y} + R_{x,y})\right| < \left|I_{x,y} - (I^q_{x,y} - R_{x,y})\right| \\ I^q_{x,y} - R_{x,y} & \text{, otherwise} \end{cases}$$

Algorithm 2. Data extraction algorithm for HPQIM

Input: stego-image I', secret key K, quantisation step Δ

Output: secret data w

1. Generate the random integer number sequence R using Algorithm 3 under the

control of a secret key *K*

2. For each pixel $I'_{x,y}$

2.1. $D_i = \min\left\{\left|(I'_{x,y} - R_{x,y}) - Q_i(I'_{x,y})\right|, \left|(I'_{x,y} + R_{x,y}) - Q_i(I'_{x,y})\right|\right\}, \quad i \in \{0,1\}$

2.2. If $R_{x,y} = \Delta/2$ and $I'_{x,y} = Q_0(I'_{x,y})$ (the special case)

$w_{x,y} = 0$

Else if $R_{x,y} = \Delta/2$ and $I'_{x,y} \neq Q_0(I'_{x,y})$ (the special case)

$w_{x,y} = 1$

Else if $D_1 < D_0$

$w_{x,y} = 1$

Else

$w_{x,y} = 0$;

2.2.2. Special Rules for Random Sequence Generation

Undoubtedly, the special embedding rule will *increase* the probabilities of the occurrence the quantisers of Q_0 and *reduce* the probabilities of the occurrence of $q_0 + \Delta/2$, where q_0 is any quantiser of Q_0, in the watermarked image I' because when $R_{x,y} = \Delta/2$ and $w_{x,y} = 0$, the quantised intensity values $I^q_{x,y}$ is taken as the final watermarked value $I'_{x,y}$ without further modification using Equation (6). As a result, the histogram H' of the original image I' will be significantly different from the histogram H of the watermarked image I. Fortunately, assuming that there are equal numbers of watermark bits taking value 0 and 1, it can be expected the

probability of occurrence of $q_0 + \Delta/2,\ \forall q_0 \in Q_0$ is *halved* due to the handling of the special case and the loss of $q_0 + \Delta/2,\ \forall q_0 \in Q_0$ is gained by and the loss of $q_0,\ \forall q_0 \in Q_0$. This can be compensated for, after the within-segment probability of each intensity has been calculated according to Equation (5), by reducing the *within-segment probability* of $p_{q_0},\ \forall q_0 \in Q_0$ by enforcing

$$p_{q_0} := \max\left(0, p_{q_0} - p_{q_0+\Delta/2}\right),\ \forall q_0 \in Q_0 \tag{8}$$

and *then* doubling the $p_{q_0+\Delta/2},\ \forall q_0 \in Q_0$, i.e.,

$$p_{q_0+\Delta/2} := 2 \cdot p_{q_0+\Delta/2} \tag{9}$$

Note that ":=" in equation (8) and (9) stands for "is replaced with", which is equivalent to the *assignment* operator in many programming languages, and we also need to set a lower bound of p_{q_0} at 0 because we cannot have a negative probability. The algorithm for generating the random integer number sequence *R* is presented in Algorithm 3.

Algorithm 3. Algorithm for generating the adaptive random number sequence for HPQIM scheme

---

Input: Image *I* of $X \times Y$ pixels, secret key *K*, quantisation step Δ
Output: Random integer number sequence *R*

1. Generate the histogram *H* of image *I*
2. Divide *H* into *S* segments, each spanning Δ intensity levels
3. For each bin/intensity *i*, calculate its corresponding *within-segment probability* p_i using Equation (5)
4. If Δ is even,

   $p_{q_0} := \max\left(0, p_{q_0} - p_{q_0+\Delta/2}\right),\ \forall q_0 \in Q_0$

   $p_{q_0+\Delta/2} := 2 \cdot p_{q_0+\Delta/2}\ ,\ \forall q_0 \in Q_0$

5. Generate the random integer number sequence

$R = \{R_{x,y} \mid x \in [0, X-1], y \in [0, Y-1], 0 \le R_{x,y} \le \Delta\}$ under the control of a secret key K such that the probability of $R_{x,y} = r$ is $p_{I_{x,y}+r}$, $r \in [0, \Delta]$

3. Experiments and Discussions

Experiments have been carried out on a number of images. Without the loss of generality, the results of the experiment on the image of Lena are demonstrated. In the experiment, Δ is set from 4 to 8, and for DC-QIM, α is set to 0.7 in order to prevent extraction errors [25]. Figure 4(a) demonstrates the histogram of the original Lena image. Figure 4(b), (c) and (d) are the histograms of the QIM watermarked image, the DC-QIM watermarked image and the HPQIM watermarked image, respectively. We can observe the disappearance of gaps in the histogram of the HPQIM watermarked image from Figure 4(d). Furthermore, we also expect that the histogram of the image is preserved after the HPQIM-based watermarking (See Section 1.2). We measure the similarity between the original histogram and the histogram of the watermarked image to examine how much the original histogram has been preserved. The Kullback-Leibler divergence D_{KL} is used to measure the similarity [20]. The general discrete expression of *K-L* divergence for comparing the similarity between two curves is

$$D_{KL} = \sum_i p_i \log \frac{p_i}{p_i'}, \tag{10}$$

where p_i and p_i' are the probability of the *i*-th component of the two curves. When applied in the context of this work for measuring the similarity between the

histograms of the original and watermarked images of $X \times Y$ pixels, D_{KL} can be expressed as

$$D_{KL} = \sum_i \frac{h_i}{X \cdot Y} \log \frac{h_i}{h_i'} \ , \tag{11}$$

where h_i and h_i' are the *i*-th bins of the histogram, indicating the frequencies of the occurrence of intensity i. A lower *K-L* divergence indicates higher similarity and a higher value bespeaks more significant difference between the histograms.

Table 1 shows the *K-L* divergence between the histograms of the original Lena image and watermarked version. From Table 1, we can see that for QIM-based watermarking, the K-L divergence increases from 1.19 to 3.03 when Δ grows from 4 to 8, and for the DC-QIM-based embedding, the *K-L* divergence increases slightly, from 1.58 to 2.01. This increase is due to the fact that as Δ grows, the dead zones become wider (See Section 1.2 and Figure 2), indicating a more significant distortion in the histogram. Meanwhile, we can see that, compared to QIM, the histogram of DC-QIM is more similar to the original histogram with a lower *K-L* divergence. However, *K-L* divergence in DC-QIM is not low enough and the detectable gaps still exist in the histogram. For the histogram of the HPQIM watermarked Lena image, the *K-L* divergence is as low as 0.03 and increases to 0.09 as Δ grows. This *K-L* divergence demonstrates that the HPQIM successfully preserves the histogram of the original signal.

From Table 2 we can see that the distortion of HPQIM is also much lower than

QIM and DC-QIM in terms of PSNR. For QIM, the distortion increases from 40.66dB to 34.72dB when Δ grows from 4 to 8. Compared to QIM, the performance of DC-QIM in terms of embedding distortion is slightly improved, from 40.71dB to 34.85dB, as Δ increases from 4 to 8. In contrast, there is a great improvement in HPQIM. When Δ is 4, the distortion is 41.99dB, and as Δ grows to 8, the distortion is 37.15dB, which is about 1.5dB to 2dB higher than the PSNR of the DC-QIM.

4. Conclusions

In this paper, we have analysed the security of traditional QIM-based watermarking schemes. Due to traditional QIM's vulnerability to the histogram analysis, we proposed an HPQIM-based watermarking scheme. The proposed HPQIM preserves the histogram of the original signal in the watermarked version and fills the gaps in the histogram by using an adaptive random number sequence generated according to the *local* probability density function (pdf) of the histogram of the original image. Thus, the proposed HPQIM achieves higher security. Meanwhile, the adaptive random number sequence ensures that the proposed scheme inflicts lower embedding distortion than traditional QIM-based and DC-QIM-based watermarking schemes.

Reference

[1] J.-P. Boyer, P. Duhamel, and J. Blanc-Talon, "Performance analysis of scalar DC-QIM for zero-bit watermarking," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 2, pp. 283-289, 2007.

[2] J.-P. Boyer, P. Duhamel and J. Blanc-Talon, "Scalar DC–QIM for Semifragile Authentication," *IEEE Transactions on Information Forensics and Security*, vol. 3 , no. 4, pp. 776 - 782, 2008.

[3] B. Chen and G. W. Wornell, "Quantization index modulation methods: A class of provable good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 563-593, 2001.

[4] B. Chen and G. W. Wornell, "Quantization index modulation methods for digital watermarking and information embedding of multimedia," *Journal of VLSI Signal Processing Systems*, vol. 27, no. 1-2, pp. 7-33, 2001.

[5] J. Fridrich, "Methods for tamper detection in digital images," in *Proceedings of the ACM Workshop on Multimedia and Security,* 1999, pp. 19-23.

[6] J. Fridrich, M. Goljan, and D. Hogea, "New methodology for breaking steganographic techniques for JPEGs," *Proceedings of SPIE, Electronic Imaging*, pp. 143-155, 2003.

[7] H. Gould, J. Tobochnik, and W. Christian, "An introduction to computer simulation methods: Applications to physical systems," *American Journal of Physics*, vol. 74, no. 7, pp. 652-663, 2006.

[8] S. Grgic, M. Mrak, M. Grgic, and B. Zovko-Cihlar, "Comparative study of JPEG and JPEG2000 image coders," in *Proceeding of the 17th International Conference on Applied Electromagnetics and Communications,* 2003, pp. 109-112.

[9] A. T. S. Ho and F. Shu, "A robust spread-spectrum watermarking method using two-level quantization," in *Proceedings of the IEEE International Conference on Image Processing*, Singapore, 2004, pp. 725-728.

[10] T. Ishida, K. Yamawaki, H. Noda and M. Niimi, "Performance Improvement of JPEG2000 Steganography Using QIM," in *Proc. International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Harbin, China 2008, pp. 155 - 158.

[11] A. Ker, "Locally Square Distortion and Batch Steganographic Capacity," *International Journal of Digital Crime and Forensics*, vol. 1, no. 1, pp. 29-44, 2009.

[12] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proceedings of the IEEE,* vol. 87, no. 7, pp. 1167-1180, 1999.

[13]C.-T. Li, "Reversible Watermarking Scheme with Image-independent Embedding Capacity" IEE Proceedings - Vision, Image, and Signal Processing, vol. 152, no. 6, pp. 779 - 786, 2005.

[14] Dependence for Image Authentication," Optical Engineering, vol. 45, no. 12, pp. 127001-1 ~ 127001-6, Dec. 2006.

[15] Q. Li and I. J. Cox, "Using Perceptual Models to Improve Fidelity and Provide Resistance to Volumetric Scaling for Quantization Index Modulation Watermarking," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 2, pp. 127 - 139, 2007.

[16] C. Y. Lin and S. F. Chang, "A robust image authentication method distinguishing JPEG compression from malicious manipulation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 2, pp. 153-168, 2001.

[17] S. H. Liu, H. X. Yao, and W. Gao, "Steganalysis of data hiding techniques in wavelet domain," in *Proceedings of the International Conference on Information Technology: Coding and Computing*, 2004, pp. 751-754.

[18] B. R. Matam and D. Lowe, "Watermark-Only Security Attack on DM-QIM Watermarking: Vulnerability to Guided Key Guessing," *International Journal of Digital Crime and Forensics*, vol. 2, no. 2, 2010.

[19] P. Moulin and A. Briassouli, "A stochastic QIM algorithm for robust, undetectable image watermarking," in *Proceedings of the IEEE International Conference on Image Processing*, Singapore, 2004, pp. 1173-1176.

[20] H. Noda, M. Miimi, and E. Kawaguchi, "High-performance JPEG steganography using quantization index modulation in DCT domain," *Pattern Recognition Letters*, vol. 27, no. 5, pp. 455-461, 2006.

[21] F. Perez-Gonzalez and C. Mosquera, "Quantization-Based Data Hiding Robust to Linear-Time-Invariant Filtering," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 2, pp. 137 - 152, 2008.

[22] X. Qi; S. Bialkowski and G. Brimley, "An adaptive QIM- and SVD-based digital image watermarking scheme in the wavelet domain," in *Proc. IEEE International Conference on Image Processing*, San Diego, CA, U.S.A, 2008, pp. 421 - 424

[23] K. Sullivan, Z. Bi, U. Madhow, S. Chandrasekaran and B. S. Manjunath, "Steganalysis of quantization index modulation data hiding," in *Proc. International Conference on Image Processing*, Singapore, 2004, pp. II-1165 - II-1168.

[24] P. Tsai, "Histogram-based reversible data hiding for vector quantisation-compressed images," *IET Image Processing*, vol. 3, no. 100, p. 114, 2009.

[25] J. E. Vila-Forcen, S. Voloshynovskiy, O. Koval, and T. Pun, "Performance analysis of nonuniform quantisation-based data-hiding," *Proceedings of SPIE: Security,*

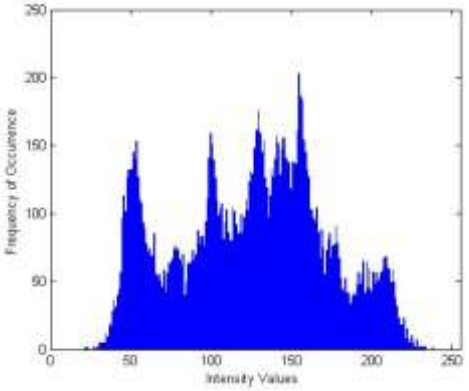
Steganography, and Watermarking of Multimedia Contents VIII, vol. 6072, pp. 354-361, 2006.

[26] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, p. xviii-xxxiv, 1992.

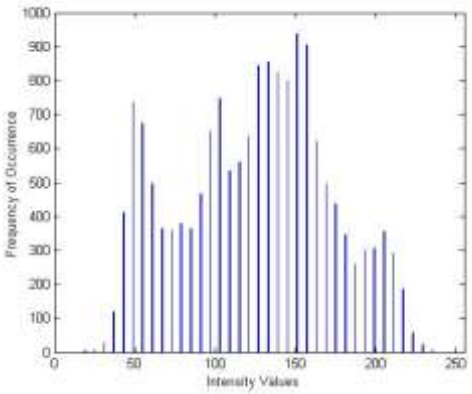
[27] Y. Yang, X.Sun, H. Yang, C.-T. Li and R. Xiao, "A Contrast-Sensitive Reversible Visible Image Watermarking Technique," IEEE Transactions on Circuits and Systems for Video Technology,vol. 19, no. 5, pp. 656 - 667, May 2009.

(a) Original Lena

(c) QIM Watermarked Lena



(b) Histogram of the original Lena



(d) Histogram of the watermarked Lena

Figure 1. The histograms of the original Lena and the watermarked Lena.

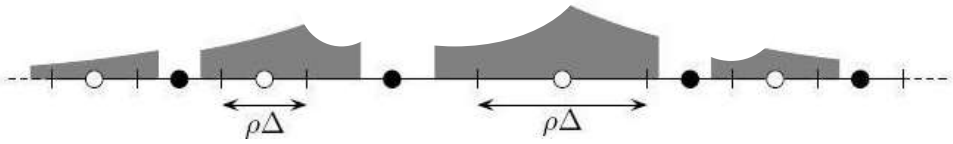
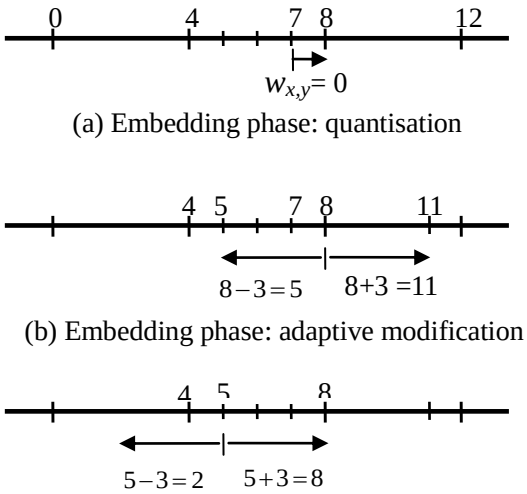


Figure 2. The histogram of the DCQIM watermarked signals.

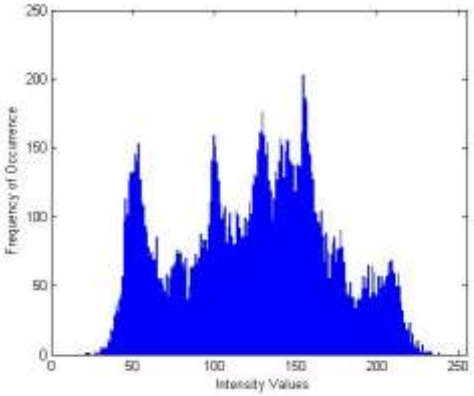


(a) Embedding phase: quantisation

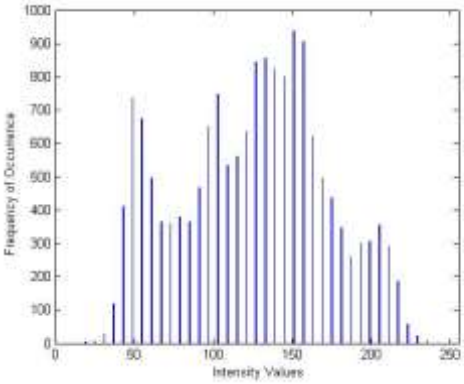
(b) Embedding phase: adaptive modification

(c) Extraction phase: extraction

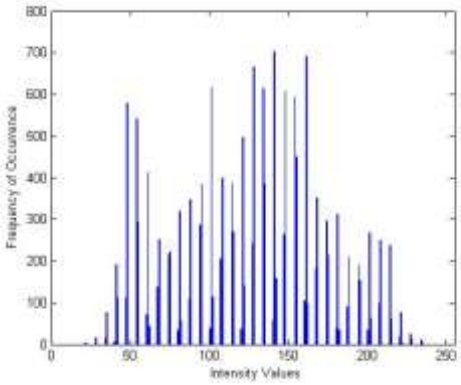
Figure 3. Demonstration of HPQIM embedding and extraction.



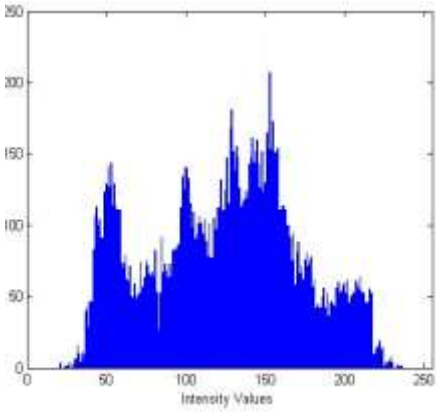
(a) The histogram of the original Lena



(b) The histogram of the QIM watermarked Lena



(c) The histogram of the DCQIM watermarked Lena



(d) The histogram of the HPQIM watermarked Lena

Figure 4. The histogram of Lena and watermarked images based on QIM, DCQIM and HPQIM.

Table 1: K-L Divergence between the histograms of the original Lena and watermarked image.

Δ	QIM	DCQIM	HPQIM
4	1.99	1.58	0.03
5	2.35	1.74	0.04
6	2.61	1.80	0.05
7	2.81	1.91	0.09
8	3.03	2.01	0.09

Table 2: Distortion of the QIM, DCQIM and HPQIM embedding (Unit: dB).

Δ	QIM	DCQIM	HPQIM
4	40.66	40.71	41.99
5	38.79	38.83	40.60
6	37.21	37.37	39.36
7	35.97	35.99	38.15
8	34.72	34.85	37.15