MODIFIED LEVEL TRANSFORMATION FOR BIT INVERSION IN WATERMARKING

Tadahiko Kimoto

Toyo University, Japan, kimoto@eng.toyo.ac.jp

ABSTRACT

The transformation function proposed in a previous paper performs both inverting a bit chosen in a multi-level signal and minimizing the resultant change of level. The remaining level changes, however, get larger in the intervals including the ends of the source level range than in the middle intervals.

The principal object of the present paper is to remove this difference in transformation performance among the intervals and thus to obtain the same properties over all the source range. To avoid the ends of the level range, a translated and extended coordinate system is proposed. The transformation is then modified and formulated so as to involve the translating of levels.

Furthermore, a function of varying the output levels in a random manner is added to the modified transformation. Consequently, the output levels can spread over the whole dynamic range. The effects of the random variations are evaluated through a stochastic analysis.

***Index Terms*—** Bit inversion, level transformation, random variation, watermarking

1. INTRODUCTION

Inverting chosen bits of signals is a basic operation for data hiding[1]. This operation works particularly effectively in a scheme of watermarking in image bit-planes[2].

Such bit inversion causes level change in a signal. In our previous paper[3] we have presented a level transformation both for inverting a chosen bit of a source signal and for minimizing the level change caused by the inverted bit. Also, we proposed a method for achieving secure data hiding with this transformation while keeping level changes below given limits.

Although this level transformation yields the output levels least different from input levels, the performances differ for the end intervals and the middle intervals in the dynamic range of input levels. This difference limits the use of the level transformation in watermarking. We refer to this transformation as *the direct transformation.*

In the present paper, we modify the level transformation so that it has the same properties in every intervals throughout the dynamic range. The modified transformation, then, is able to be applied to any input level in the same manner. We also evaluate its properties in the whole input range.

The paper is organized as follows. In Section **2**, according to [3], we explain the principle for minimizing changes caused by inverted bits in signal levels. We then demonstrate the difference between the characteristics of remaining level changes in the end intervals and the middle intervals of the input range. This difference is removed by using a level translation in Section **3**. We modify the level transformation and formulate it so that it involves the level translation. In Section **4**, we add a function of varying transformed levels in a random manner, which is desirable in data hiding, to the above modified transformation with a method similar to that applied to the direct transformation [4]. The entire transformation is also analyzed in a stochastic manner. We then evaluate the effects of random variation in the transformed levels. Finally, we conclude the paper in Section **5**.

2. LEVEL TRANSFORMATION FOR BIT INVERSION

2.1. Inverting a bit of a signal

Inverting one of bits of a signal changes the signal level by the amount determined by the bit. Each signal is supposed to be uniformly quantized to M bits and expressed in natural binary. The bits of a signal are numbered from 0 at the least significant bit (LSB) to $M-1$ at the most significant bit (MSB). For a given signal, specifying a value of k, suppose that its kth bit is inverted ($0 \le k \le M-1$). The level, then, increases or decreases by 2^k.

2.2. Minimizing level changes

The level change caused by an inverted bit can be minimized by altering the other bits appropriately. The principle for transforming the entire bit-pattern by the smallest level change is illustrated in Fig. 1. Let $b(v,\ k)$ denote a binary value of the kth bit of an M-bit level v, and $\overline{b(v,\ k)}$ its binary inverse ($0 \le k \le M-1$). Suppose that the kth bit of a level v is to be inverted. Then, the inverting of the bit is performed by transforming the entire bit-pattern to one of those patterns of level v' which satisfy $b(v',\ k) = \overline{b(v,\ k)}$. Furthermore, the one of them closest to v can be chosen to achieve the smallest level change from v to v'.

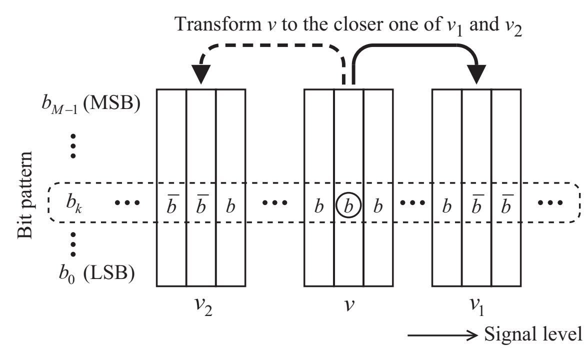

Fig. 1. Transforming of an entire bit-pattern.

For a given value of k, the available level v' closest to each level v is determined. This relationship between v and v' is shown in Fig. 2. Here, let $\Delta_k = 2^{k-1}$. For a level v in the interval $[0,\ \Delta_k - 1]$, those levels available for bit inversion exist only on the upper side of v because of the end of M-bit levels. Accordingly, all the source levels in the interval are transformed to the smallest one of the available levels, $2\Delta_k$. Similar end effects occur for the levels in the interval $[2^M - \Delta_k,\ 2^M - 1]$. In this paper we refer to these two intervals as the *end intervals*. The relationship between v and v' in the end intervals are shown by the broken lines in Fig. 2.

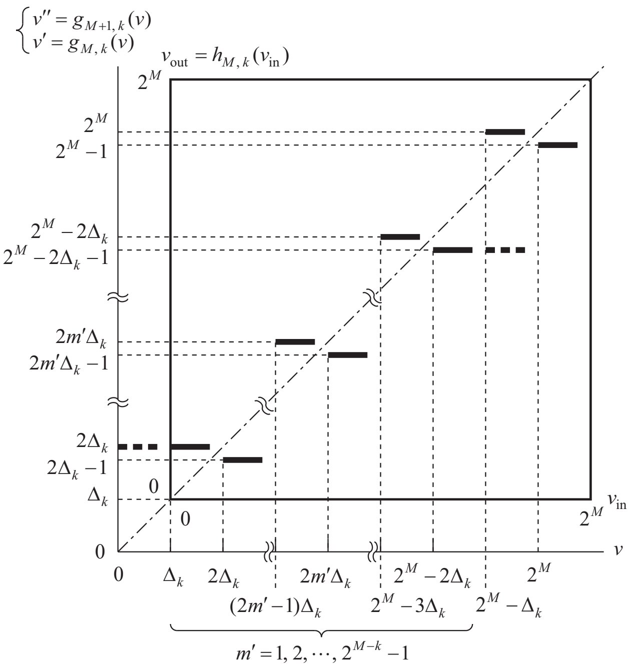

Fig. 2. Level transformation for bit inversion.

The whole relationship between a level v in the range $[0,\ 2^M - 1]$ and the corresponding level v' can be expressed as a level transformation from v to v'[3]. We denote the transformation function by $g_{M,\ k}(v)$ for given M and k.

The remaining level difference by the level transformation is also a function of the source levels. Figure 3 shows the difference of the transformed level $v' = g_{M,\ k}(v)$ and a source level v in the whole source range. The absolute magnitude of the difference ranges from 1 to Δ_k for the source levels v in the interval $[\Delta_k,\ 2^M - \Delta_k - 1]$. In the two end intervals, meanwhile, the level differences have the magnitude greater than Δ_k.

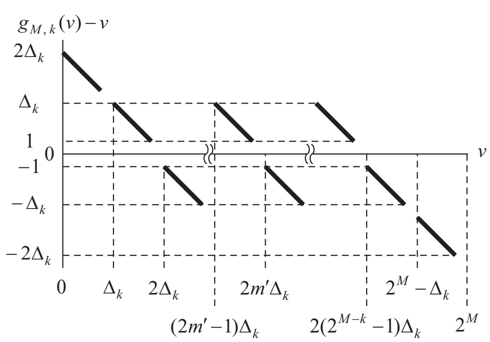

Fig. 3. Level differences caused by $v' = g_{M,\ k}(v)$ ($m' \in [1,\ 2^{M-k} - 1]$).

3. MODIFYING LEVEL TRANSFORMATION

3.1. Translating levels

The M-bit end effects can be avoided by translating the coordinate system. For given M and k, first, to remove the upper end effects, we use the function $g_{M+1,\ k}(v)$ instead of $g_{M,\ k}(v)$. Second, to avoid the lower end effects, we translate both the coordinates by Δ_k. These operations are expressed in the relation

$$v_{\text{out}} + \Delta_k = g_{M+1,\ k}(v_{\text{in}} + \Delta_k), \tag{1}$$

where v_{in} and v_{out} are the translated coordinates. The coordinate system with v_{in}- and v_{out}-axis is shown in Fig. 2.

In the new coordinate system the level transformation from v_{in} to v_{out} is expressed by the transformation function $h_{M,\ k}$ of v_{in}, which is defined by the relation

$$\begin{aligned} v_{\text{out}} &= h_{M,\ k}(v_{\text{in}}) \\ &= \begin{cases} (2m+1)\Delta_k, & \text{for } 2m\Delta_k \le v_{\text{in}} \le (2m+1)\Delta_k - 1 \\ (2m+1)\Delta_k - 1, & \text{for } (2m+1)\Delta_k \le v_{\text{in}} \le (2m+2)\Delta_k - 1 \end{cases} \end{aligned} \tag{2}$$

where $m = 0,\ 1, \cdots,\ 2^{M-k} - 1$. Thus, this transformation maps 2^M consecutive source levels into 2^{M-k+1} discrete output levels.

The level transformation is no longer affected by the M-bit end effects. Hence, the level change has the magnitude in the range $[1,\ \Delta_k]$ throughout the whole source range, as shown in Fig. 4.

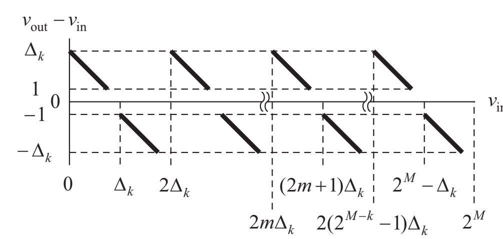

Fig. 4. Level differences caused by $v_{\text{out}} = h_{M,\ k}(v_{\text{in}})$ ($m \in [0,\ 2^{M-k} - 1]$).

3.2. Implementation of bit inversion

3.2.1. Access to object bits

Levels need to be shifted to observe the inversion of bits because the inverting of chosen bits are performed in the original coordinate system. Given values of M and k, and a signal of level v_{in} in the range $[0,\ 2^M - 1]$, the procedure for implementing the bit inversion is as follows.

Step 1: Let $v = v_{\text{in}} + \Delta_k$ modulo 2^M.

Step 2: Extract $b(v,\ k)$, which is considered the kth bit associated with the level v_{in}.

Step 3: Use the level transformation $v_{\text{out}} = h_{M,\ k}(v_{\text{in}})$ to invert $b(v,\ k)$. We then obtain v_{out} so that $b(v_{\text{out}} + \Delta_k,\ k) = \overline{b(v_{\text{in}} + \Delta_k,\ k)}$.

3.2.2. *Estimation function for level differences*

Given limits to level differences allowable at each source level, which can be determined from human perception, the bit inversion can be implemented in a manner such that actual level differences lie within the limits. If the magnitude of level difference that is to be caused by a level transformation is less than the limits at a level v, then, the level transformation for the level can be carried out.

We have proposed the use of an estimation function for level differences to compare to the limits instead of the exact differences caused by the level transformation[3]. The estimation function is set constant over the source range so that all the levels are equally likely to be transformed for the same limits. The estimation function for $g_{M,\,k}(v)$ was defined as a constant Δ_k throughout the source range except the two end intervals where the level differences exceed Δ_k. On the other hand, the estimation function for $h_{M,\,k}(v)$ is able to be defined as the constant Δ_k throughout the whole source range.

4. VARYING TRANSFORMED LEVELS

4.1. Randomizing lower bits

Adding random variations to the transformed levels makes the bit patterns different from fixed ones and consequently, makes it difficult to recognize them. Our scheme is to randomize some lowest order bits in the bit pattern of a transformed level[4]. We here formulate this scheme using the above modified level transformation.

In an M-bit pattern of the level that $h_{M,\,k}(v)$ yields, the $k-1$ lowest order bits are available to change without losing the validity of bit inversion. All of these bits of a transformed level have the same value, either 0 or 1. Accordingly, we replace the ℓ lowest order bits of a value of $h_{M,\,k}(v)$ by random bits, where $0 \leq \ell \leq k-1$. Here, the case when $\ell = 0$ means that no bit is replaced. For given M, k and ℓ, $h_{M,\,k}(v)$ and the operation of bit replacement that follows the transformation can be united to a single level transformation of two variables v and r, denoted by $h_{M,\,k,\,\ell}(v,\ r)$; the function is defined by

$$h_{M,\,k,\,\ell}(v,\ r) = \begin{cases} h_{M,\,k}(v) + r, & \text{for } 2m\Delta_k \leq v \leq (2m+1)\Delta_k - 1 \\ h_{M,\,k}(v) - r, & \text{for } (2m+1)\Delta_k \leq v \leq (2m+2)\Delta_k - 1 \end{cases} \quad (3)$$

where r is an arbitrary integer in the range $[0,\ \delta_\ell]$, and $\delta_\ell = 2^\ell - 1$.

The number of randomized bits ℓ is determined from the magnitude of level differences for each source level. The number ℓ in (3) can increase as long as the total level difference lies below a value of the estimation function described in Section **3.2.2**. The largest one of such ℓ's is expressed as a function of a source level v for a given k, which we denote by $L_k(v)$.

For given M and k, the level transformation $h_{M,\,k,\,L_k(v)}(v,\ r)$ is determined in the whole source range. Fig. 5 shows the characteristics of the transformation function $h_{M,\,k,\,L_k(v)}(v,\ r)$ and the values of the function $L_k(v)$ in a source range $[2m\Delta_k,\ 2(m+1)\Delta_k - 1]$ for $m = 0,\ 1,\ \cdots,\ 2^{M-k} - 1$. As Fig. 5(c) shows, the number of random bits given by $L_k(v)$ varies in the range $[0,\ k-1]$. The function $L_k(v)$ performs periodically with a period of $2\Delta_k$ levels in the whole range of v. At each v, according to the value of $L_k(v)$, the output levels vary in the range of $2^{L_k(v)}$ levels, which is depicted in the shaded areas in Fig. 5(a). So do the level differences, as shown in Fig. 5(b). Thus, this transformation maps the whole M-bit source range $[0,\ 2^M - 1]$ into the same range of output levels in a stochastic manner.

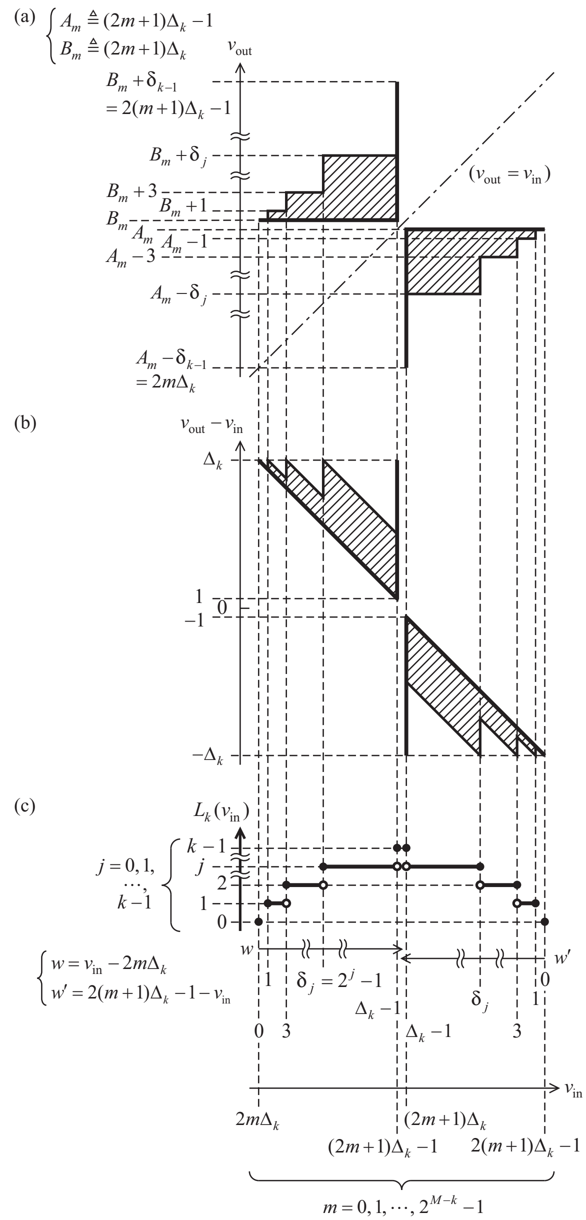

Fig. 5. Level transformation with bit randomization: (a) input-output relationship; (b) level differences; and (c) numbers of random bits.

4.2. Analysis of randomized levels

We analyze the level transformation with bit randomization and that without bit randomization, and compare them to evaluate effects of the random variation on the output levels. Here, we assume that the frequencies of source levels are distributed uniformly over the whole M-bit range $[0,\ 2^M - 1]$. Also, the random numbers r in (3) are assumed to be equally likely to be any value within the range $[0,\ 2^{L_k(v)} - 1]$ at each source level v.

4.2.1. Mean square difference

For given M and k, we average values of the square difference $(h_{M,\,k,\,L_k(v)}(v,\ r)-v)^2$ over v in the whole source range. The mean square difference, denoted by $E_{\mathrm{R}k}$, is then given by

$$E_{\mathrm{R}k}=\frac{(3\Delta_k^2+\Delta_k-1)(\Delta_k+1)}{6\Delta_k}. \tag{4}$$

Similarly, in the case without bit randomization, the square differences $(h_{M,\,k}(v)-v)^2$ are calculated. Then, the mean square difference over the whole source range, denoted by $E_{\mathrm{NR}k}$, is given by

$$E_{\mathrm{NR}k}=\frac{(2\Delta_k+1)(\Delta_k+1)}{6}. \tag{5}$$

Using (4) and (5), we obtain the ratio of the mean square differences:

$$\frac{E_{\mathrm{R}k}}{E_{\mathrm{NR}k}}=\frac{3\Delta_k^2+\Delta_k-1}{\Delta_k(2\Delta_k+1)}=\frac{3+1/\Delta_k-1/\Delta_k^2}{2+1/\Delta_k}, \tag{6}$$

which we denote by $p_{\mathrm{E}k}$. This ratio indicates the increase of level change caused by the bit randomization with k. Equation (6) reveals that $p_{\mathrm{E}k}$ approaches $3/2$ asymptotically as k increases.

4.2.2. Variance in level distribution

The distribution of output levels over the M-bit range is evaluated. Analyzing the input-output relationship of $h_{M,\,k,\,L_k(v)}(v,\ r)$ illustrated in Fig. 5(a), we obtain the frequency distribution of the output levels shown in Fig. 6. The variance of the relative frequencies is then given by

$$V_{\mathrm{R}M,k}=\frac{1}{2^M}\left(2-\frac{2k-1}{\Delta_k}-\frac{1}{\Delta_k^2}\right). \tag{7}$$

On the other hand, the variance of the relative frequencies of output level $h_{M,\,k}(v)$ is given by

$$V_{\mathrm{NR}M,k}=\frac{\Delta_k-1}{2^M}. \tag{8}$$

Then, using (7) and (8), we obtain the ratio of these two variances:

$$\frac{V_{\mathrm{R}M,k}}{V_{\mathrm{NR}M,k}}=\frac{2-(2k-1)/\Delta_k-1/\Delta_k^2}{\Delta_k-1}, \tag{9}$$

which we denote by $p_{\mathrm{V}k}$.

Equation (7) shows that the value of $V_{\mathrm{R}M,k}$ increases as k increases and approaches 2^{1-M} asymptotically. The ratio $p_{\mathrm{V}k}$ in (9) hence decreases as k increases. If $M=8$ and $k=3$, as an example, which is typical one in practical use[3], $p_{\mathrm{V}3}=0.23$.

5. CONCLUDING REMARKS

The proposed level transformation successfully avoids the end effects of fixed-length bits by involving the translation of coordinate axes. At the same time, the additional operation of translating input levels is required to access object bits before and after the level transformation. This operation is so simple that no substantial increase of computation loads would occur.

The effects of random variations in the transformed levels are summarized as follows:

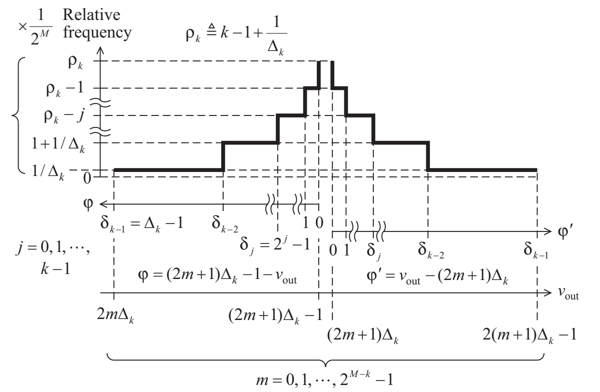

Fig. 6. Histogram of output levels obtained by a stochastic analysis.

- The randomization makes a level difference larger than its minimum for each input level. The stochastic analysis has indicated that the mean square level difference increases by a factor of at most $3/2$.
- The output levels are distributed all over the range by the stochastic randomization while both end intervals are excluded from the output range in the direct transformation including the same randomization scheme[4].

  The stochastic analysis has also demonstrated that the randomization decreases variances in the output level distribution effectively.

On the other hand, we need the following conditions to be satisfied to implement the proposed level transformation:

a) Only a single bit of a signal is chosen for inversion.
b) The other bits are available for any other processing.

These conditions allow the transformation to change the other bits so as to compensate the level change caused by the inverted bit appropriately. It can therefore be applied to any of data hiding methods that satisfy the above conditions.

6. REFERENCES

[1] B. M. Macq, ed., "Special issue on identification and protection of multimedia information," *Proc. IEEE*, vol. 87, no. 7, pp. 1059–1276, July 1999.

[2] K. Oka and K. Matsui, "Signature method into gray-scale images with embedding function," *Trance. IEICE*, vol. J80-D-II, no. 5, pp. 1186–1191, May 1997, in Japanese.

[3] T. Kimoto, "Implementation of level transformations for hiding watermarks in image bit-planes under limited level changes," in *Proc. IEEE Int. Conf. on Image Processing*, September 2005, pp. 253–256.

[4] T. Kimoto, "A sophisticated bit-conversion method for digital watermarking," in *Proc. 8th IASTED Int. Conf. on Signal and Image Processing*, August 2006, pp. 139–144.