Permutation-Based Signature Generation for Spread-Spectrum Video Watermarking

Hiroshi ITO[a] and Tadashi KASEZAWA[b], Members

SUMMARY Generation of secure signatures suitable for spread-spectrum video watermarking is proposed. The method embeds a message, which is a two-dimensional binary pattern, into a three-dimensional volume, such as video, by addition of a signature. The message can be a mark or a logo indicating the copyright information. The signature is generated by shuffling or permuting random matrices along the third or time axis so that the message is extracted when they are accumulated after demodulation by the correct key. In this way, a message is hidden in the signature having equal probability of decoding any variation of the message, where the key is used to determine which one to extract. Security of the proposed method, stemming from the permutation, is evaluated as resistance to blind estimation of secret information. The matrix-based permutation allows the message to survive the spatial down-sampling without sacrificing the security. The downside of the proposed method is that it needs more data or frames to decode a reliable information compared to the conventional spread-spectrum modulation. However this is minimized by segmenting the matrices and applying permutation to sub-matrices independently. Message detectability is theoretically analyzed. Superiority of our method in terms of robustness to blind message estimation and down-sampling is verified experimentally.

1. Introduction

Permutation does not change the first order statistics of data and hence has been used in steganography to make the hidden information less visible to the third party. Mittelholzer [1] showed that a stego vector generated by the following equation achieves perfect secrecy in terms of mutual information,

\[ y = x + \Pi_f k, \] (1)

where \( x \) and \( k \) are cover and key vectors respectively and \( \Pi_f \) is a permutation matrix indexed by \( \phi \). Blaño et al. [2] proposed a steganographic scheme with a slightly different setting of

\[ y = \Pi_f x, \] (2)

where the stego vector is a direct permutation of the cover signal.

Although security of general information hiding systems varies from application to application, invisibility of secret information is a universal requirement of all such systems. In steganography [3], [4], the goal is simple enough to state that the existence itself of hidden information must not be noticed. Due to this simplicity, techniques derived from the minimum impact embedding [5] have been established in these decades as practical steganographic methods, where the impact is locally computed based on visibility of the fact of embedding or changing signal values [6], [7].

Things are complicated for the case of watermarking since its security requirements are far more diverse [8], [9]. In watermarking it is usually allowed to make public the fact that contents have watermarks. What is to be avoided are then unauthorized embedding, detection, and removal of them [8]. Stirmark [10] has been a classical set of tools to remove watermarks, by noise addition, filtering, compression, and giving geometric distortions so that the receiver fails to retrieve the embedded information. Later more sophisticated methods have been proposed to directly extract secret information from stego signals with no knowledge of the key. Independent component analysis (ICA) has been used to separate signatures depending on their statistical difference to the cover [11], [12]. An EM algorithm in [13] blindly recovers the message from noise-looking spread-spectrum signatures based on their uneven probability distribution in the embedding domain. Once the signature is extracted, it is easy to remove the watermark or even to make a fake watermarked content. Message recovery is clearly a direct threat of unauthorized message detection.

We employ the permutation modulation in generating a signature to alleviate the above threats, particularly the last mentioned blind message estimation. A method using permutation for watermarking has been disclosed in [1]. They generated a signature by permuting elements of a random vector, called carrier, depending on the message. We extend the permutation unit to a two dimensional array or a matrix and the carrier to a three-dimensional volume, so we permute the plains in a volume along the remaining third axis. In this volume we hide an arbitrary binary matrix of the same size as the permutation unit. Assuming that the carrier is made of independent, identically distributed (i.i.d.) random variables, any signatures carrying different messages or using different keys can not be distinguished statistically from the original carrier so the blind message recovery would fail. Another motivation of extending the permutation unit is its robustness to spatial down-sampling,
which is often performed for video to adjust the frame to a particular screen. Since all the elements or pixels in a permutation unit are modulated in the same way, spatial decimation of the stego signal only leads to decimation of the message directly. Hence the message survives as long as it is spatially redundant which is usually the case for a copyright mark or a logo. Embedding a two dimensional pattern in a video has been proposed in [14] with no consideration on security issues.

Rest of the paper is organized as follows. In Sect. 2, methods of encoding and decoding are provided followed by short investigation in message secrecy. Then message detectability is theoretically analyzed compared to the conventional spread-spectrum watermarking. It is shown that the proposed method needs more data to recover the message. In Sect. 3, message segmentation is proposed to alleviate this data expansion problem. Synchronization between keys and segments is also discussed. Simulation results are shown in Sect. 4, where robustness to blind message estimation and down-sampling is evaluated together with message detectability and performance of the proposed key synchronization scheme. Practical implementation is discussed in Sect. 5. Section 6 concludes the paper.

2. Principles

2.1 Carrier Modulation and Message Detection

Let \( \mathbf{m} = (m_0, \ldots, m_{N-1})^t \) and \( \mathbf{k} = (k_0, \ldots, k_{M-1})^t \) be message and key signals both being binary vectors of length \( N \) and \( M \) respectively taking values in \( \{-1, 1\} \). Superscript \( t \) represents the transpose operation. In case of video, \( N \) represents the number of pixels in a frame and \( M \) is the number of frames. More natural representation of \( \mathbf{m} \) would be by using a matrix but we denote it as a vector for notational convenience. In encoding, a carrier is generated as a \( N \times M \) matrix consisting of i.i.d. real random variables of zero-mean, given by

\[
\mathbf{U} = [\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{M-1}],
\]

where \( \mathbf{u}_j = (u_{0,j}, \ldots, u_{N-1,j})^t \) and \( u_{ij} \in \mathcal{R} \). Modulation of the carrier is a simple column permutation of \( \mathbf{U} \), producing a signature matrix of

\[
\mathbf{V} = [\mathbf{u}_{\varphi(0)}, \mathbf{u}_{\varphi(1)}, \ldots, \mathbf{u}_{\varphi(M-1)}],
\]

where \( \varphi(j) \in \{0, \ldots, M - 1\} \) and \( \varphi(k) \neq \varphi(l) \) for \( k \neq l \). The permutation is carried out so that the number of vectors that satisfies the following inequality is increased.

\[
k_j \mathbf{v}_j^t \mathbf{m} = \sum_{i=0}^{N-1} k_{j} m_{i} v_{ij} > 0,
\]

where \( \mathbf{v}_j \) is the \( j \)-th column vector of \( \mathbf{V} \). This gives a positive bias to \( k_j m_i v_{ij} \) and will be used to recover the message \( \mathbf{m} \). Since \( \mathbf{V} \) is a column permutation of \( \mathbf{U} \), embedding does not change the distribution of variables in the carrier. Finally \( \mathbf{V} \) is added to the cover matrix \( \mathbf{X} \) yielding a stego signal.

\[
Y = X + \mathbf{V}
\]

which is sent to the receiver. In the sequel, we assume that \( \mathbf{m}, \mathbf{k}, \mathbf{U} \) (or \( \mathbf{V} \)) and \( X \) are all uncorrelated. The proposed watermarking system together with nomenclature used above is depicted in Fig. 1.

The receiver shares the key vector \( \mathbf{k} \) and recovers the message by computing

\[
\hat{m}_j = \text{sgn} \left( \sum_{i=0}^{M-1} k_j v_{ij} \right)
\]

\[
= \text{sgn} \left( \sum_{i=0}^{M-1} k_j x_{ij} + \sum_{j=0}^{M-1} k_j y_{ij} \right)
\]

where \( \text{sgn}(x) \) returns \( \pm 1 \) depending on the sign of \( x \). The first summation in the last expression of Eq. (7) has no bias or is zero in average. But the second summation tends to be positive for \( m_i = 1 \) and negative for \( m_i = -1 \) because of the property imposed by inequality (5), by which \( k_j m_i v_{ij} \) has been shifted toward the positive side. This positive bias would be strengthened by the summation over \( j \). As a result, \( \hat{m}_i = m_i \) comes to hold with high probability for a large number of observations. Detectability will be analyzed in more detail in the following section.

A naive modulation algorithm which transforms \( \mathbf{U} \) into \( \mathbf{V} \) is given in the following.

**Algorithm 1: [Signature Generation]**

**Step 1** Set \( i, \varphi(i) \leftarrow i \) and \( j \leftarrow 0 \).

**Step 2** If \( j = M \), stop.

**Step 3** If \( k_j u_{\varphi(i)j}^t \mathbf{m} > 0 \), then set \( j \leftarrow j + 1 \) and go to Step 2.

**Step 4** Set \( l \leftarrow j + 1 \).

**Step 5** If \( l = M \), then set \( j \leftarrow j + 1 \) and go to Step 2.

**Step 6** If \( k_j u_{\varphi(l)j}^t \mathbf{m} > 0 \), then set \( \varphi(j) \leftarrow l \), \( \varphi(l) \leftarrow j \), \( j \leftarrow j + 1 \) and go to Step 2.

**Step 7** \( l \leftarrow l + 1 \) and go to Step 5.

Condition in Eq. (5) is checked for \( \mathbf{u}_{\varphi(i)} \) in Step 3. If it is satisfied, no index exchange is performed. If it is not, an index \( l(> j) \) for which \( k_j u_{\varphi(i)j}^t \mathbf{m} > 0 \) is looked for and indices are exchanged in Step 6 if found. Sometimes we may fail to find such \( l \). In that case, we leave the index \( j \) unchanged (Step 5). Note that, for an arbitrary \( l \), we have \( k_j u_{\varphi(l)j}^t \mathbf{m} > 0 \) with probability \( 1/2 \). Hence probability of consecutive fails in Step 3 and Step 6 would be \( 1/2^{M-j} \), meaning that the failure only happens toward the end of the encoding. The algorithm computes \( \varphi(j) \). Once \( \varphi(j) \) is given, \( \mathbf{V} \) is computed.
from \( U \) by Eq. (4).

2.2 Key and Message Secrecy

Note that uncorrelated random variables contain all frequency components. Hence a column vector of the carrier is made of combination of all possible patterns of the message. It is understood that the modulation is performed so that a specific pattern or a message is boosted when accumulating vectors with changing signs depending on the key. This means that any message can be decoded from the same signature if we use a different key. This is the Vernam cipher. Assuming that choosing a key is completely random, given a stego \( Y \), we have probabilities as

\[
P(m|Y) = \frac{1}{2^N} = P(m),
\]

(8)

for arbitrary \( m \) and therefore we have the mutual information as

\[
I(M; Y) = H(M) - H(M|Y) = 0,
\]

(9)

meaning that knowing \( Y \) reveals nothing about the message \( M \).

Above discussion does not guarantee that \( V \) is not predicted from \( Y \). We further assume in the rest of the paper that \( X \) and \( U \) (and therefore \( V \)) are both Gaussian so that ICA algorithms would fail to separate \( V \) from \( Y \).

2.3 Analysis of Detectability

Let \( x_{ij} \sim \mathcal{N}(0, \sigma_x^2) \) and \( v_{ij} \sim \mathcal{N}(0, \sigma_v^2) \), respectively. In this section, we are interested in statistics of the quantity,

\[
d_i = \sum_{j=0}^{M-1} k_j m_i v_{ij}
\]

(10)

We first derive the mean of \( d_i \). The mean of the first term in Eq. (10) is zero since \( m \) and \( X \) are uncorrelated, i.e.,

\[
E\left(\sum_{j=0}^{M-1} k_j m_i x_{ij}\right) = 0,
\]

(11)

with \( E(x) \) representing the mean of \( x \). For the second term of Eq. (10), first note that for arbitrary \( j \in \{0, \ldots, M-1\} \), distribution of the quantity \( \sum_{i=0}^{N-1} m_i v_{ij} \) is given by \( \mathcal{N}(0, N\sigma_v^2) \). Since \( k_j = \pm 1 \) and from Eq. (5), we see that \( k_j v_j m \) follows the half normal distribution, whose mean is given by

\[
E(k_j v_j m) = E\left(\sum_{i=0}^{N-1} k_j m_i v_{ij}\right)
\]

\[
= \sqrt{N}\sigma_v \sqrt{\frac{2}{\pi}} = \sigma_v \sqrt{\frac{2N}{\pi}}.
\]

(12)

It is straightforward to see that the mean of its sum over \( j \) is given by

\[
E\left(\sum_{j=0}^{M-1} \sum_{i=0}^{N-1} k_j m_i v_{ij}\right) = \sigma_v M \sqrt{\frac{2N}{\pi}}.
\]

(13)

Now we omit the sum over \( i \) which gives

\[
E\left(\sum_{j=0}^{M-1} k_j m_i v_{ji}\right) = m_i E\left(\sum_{j=0}^{M-1} k_j v_{ji}\right)
\]

\[
= \frac{1}{N}\sigma_v M \sqrt{\frac{2N}{\pi}} = \sigma_v M \sqrt{\frac{2}{\pi N}}.
\]

(14)

Using Eqs. (11) and (14), we obtain

\[
E(d_i) = E\left(\sum_{j=0}^{M-1} k_j m_i x_{ij}\right) + E\left(\sum_{j=0}^{M-1} k_j m_i v_{ij}\right)
\]

\[
= \sigma_v M \sqrt{\frac{2}{\pi N}}.
\]

(15)

showing that \( d_i \) has some positive value in average.

Next we are interested in the variance of \( d_i \). Let \( \text{Var}(x) \) represent the variance of \( x \). Since \( X \) and \( V \) are uncorrelated and \( k_j, m_i \in \{-1, 1\} \), we get

\[
\text{Var}(d_i) = \text{Var}\left(\sum_{j=0}^{M-1} k_j m_i x_{ij}\right)
\]

\[
+ \text{Var}\left(\sum_{j=0}^{M-1} k_j m_i v_{ij}\right)
\]

\[
= \text{Var}\left(\sum_{j=0}^{M-1} x_{ij}\right) + \text{Var}\left(\sum_{j=0}^{M-1} v_{ij}\right)
\]

\[
= M(\sigma_x^2 + \sigma_v^2),
\]

(16)

where we used Eq. (11). Using the standard deviation, \( \text{Std}(x) = \sqrt{\text{Var}(x)} \), instead, we have

\[
\text{Std}(d_i) = \sqrt{M(\sigma_x^2 + \sigma_v^2)}.
\]

(17)

When \( M \) is large enough, we can approximate the distribution of \( d_i \) by Gaussian. Then probability of error in decision with Eq. (7) is given by

\[
P_e = 1 - \text{erf}\left(\frac{E(d_i)}{\text{Std}(d_i)}\right)
\]

\[
= 1 - \text{erf}\left(\sqrt{\frac{2M}{\pi N}} \frac{\sigma_v}{\sqrt{\sigma_x^2 + \sigma_v^2}}\right)
\]

\[
= 1 - \text{erf}\left(\sqrt{\frac{2M}{\pi N}} \frac{1}{\sqrt{1 + \gamma^2}}\right),
\]

(18)

where \( \gamma^2 = \frac{\sigma_x^2}{\sigma_v^2} \) is the cover-to-signature (C/S) ratio and \( \text{erf}(\cdot) \) is the error function.

2.4 Corresponding Spread Spectrum Modulation

Permutation is not the only way to embed message \( m \) in
matrix $X$. Indeed conventional spread spectrum (SS) modulation does the same thing with far less distortion. Here we consider a SS version of watermarking, given by the following equation,

$$y_{ij} = x_{ij} + \sigma_v k_j m_i,$$

where $\sigma_v^2$ is the signature variance. Detection of the message is implemented in a similar way to Eq. (7) by

$$\hat{m}_i = \text{sgn} \left( \sum_{j=0}^{M-1} k_j x_{ij} + M \sigma_v m_i \right).$$

(20)

Define as in Eq. (10),

$$d_i = \sum_{j=0}^{M-1} m_j k_j y_{ij} = \sum_{j=0}^{M-1} m_j k_j x_{ij} + M \sigma_v,$$

then we have

$$\mathbb{E}(d_i) = \mathbb{E} \left( \sum_{j=0}^{M-1} k_j m_i x_{ij} + (M \sigma_v) \right) = M \sigma_v$$

(22)

and

$$\text{Var}(d_i) = \text{Var} \left( \sum_{j=0}^{M-1} k_j m_i x_{ij} \right) + \text{Var}(M \sigma_v) = M \sigma_v^2,$$

(23)

where we used uncorrelatedness of $k$, $m$ and $X$. The error probability is given by

$$P_e = 1 - \text{erf} \left( \frac{M \sigma_v}{\sqrt{2 \pi N} \sigma_x} \right) = 1 - \text{erf} \left( \frac{\sqrt{M} \sigma_v}{\sigma_x} \right) = 1 - \text{erf} \left( \frac{\sqrt{M}}{\gamma} \right).$$

(24)

Now we compare the proposed method with SS modulation. Instead of using Eqs. (18) and (24), we focus on the expectation versus standard deviation ($\mu/\sigma$) ratio of $d_i$ given by

$$\frac{\mu}{\sigma} = \sqrt{\frac{2M}{\pi N} \frac{\sigma_{ve}^2}{\sigma_x^2 + \sigma_{ve}^2}}$$

(25)

and

$$\frac{\mu}{\sigma} = \sqrt{\frac{M \sigma_{vss}^2}{\sigma_x^2}} = \frac{\sqrt{M}}{\gamma},$$

(26)

respectively, where we used suffixes “ex” and “ss” to discriminate permutation and SS modulations. Figure 2 plots them in terms of $\gamma$ for two values of $N = 8$ and $N = 64$. The graph shows decreased $\mu/\sigma$ ratio or increased error probability in permutation modulation for the same $\gamma$ compared to the SS modulation. This is more pronounced for a large $N$. By equating $(\mu/\sigma)_{ex} = (\mu/\sigma)_{ss}$, we can obtain correspondence in signature strengths resulting in equal detectability as

$$\frac{\sigma_{ve}^2}{\sigma_{vss}^2} = \frac{\pi N}{\gamma^2} > \frac{\pi N}{2},$$

(27)

where $\gamma_{vss}^2$ is the C/S ratio of the SS modulation. For small distortion with $\gamma_{ss}^2 \gg \pi N/2$, we have

$$\frac{\sigma_{ve}^2}{\sigma_{vss}^2} \approx \frac{\pi N}{2}$$

(28)

meaning that the signature variance must be $\pi N/2$ times larger for the proposed method. On the other hand, if we keep the C/S ratio constant, we obtain

$$\frac{(\mu/\sigma)_{ex}}{(\mu/\sigma)_{ss}} = \sqrt{\frac{2}{\pi N} \frac{\gamma}{\sqrt{1 + \gamma}}} < \sqrt{\frac{2}{\pi N}},$$

(29)

and

$$\frac{(\mu/\sigma)_{ex}}{(\mu/\sigma)_{ss}} \approx \sqrt{\frac{2}{\pi N}},$$

(30)

for the small distortion case. Since $\mu/\sigma$ ratio is proportional
to $\sqrt{M}$, Eq. (30) indicates that we need $\pi N/2$ times more samples for the permutation modulation to get the same level detectability.

3. Message Segmentation

3.1 Motivation

As shown in the previous section, the number of observations required for decoding grows in proportion to the message size. This is also explained by that expressible ability of column vectors increases exponentially for the value of $N$ and its correlation to any specific message decreases linearly. This expansion will severely narrow the application since the observation is practically finite and it limits the size of the message.

In this section, we propose a method where the message is segmented in small blocks and each of them is given a unique permutation key. Required observations decrease according to the block size in exchange of higher decoding complexity since a mechanism of spatial synchronization between segments and keys is newly required. The proposed scheme includes an extreme case where the maximum segmentation is applied until the size of a block reaches a single pixel. This keeps the observations at the minimum but makes synchronization more strict up to the pixel-wise accuracy.

A major motivation to use block sizes larger than a single pixel, besides relaxed spatial synchronization, is its robustness to low-pass filtering or sub-sampling applied to the stego signal. This is so when the message is slowly changing. Because the key is constant in blocks, message signals embedded in the signature will not change rapidly as far as they are in the same block and will survive spatial decimation. Therefore down-sampling will only lead to detection of small sized messages if we ignore boundary pixels. Note that this spatial redundancy is lost when the single pixel segmentation is used since the sign of adjacent signals change randomly due to different keys. It will be shown in Sect. 4 that 2 $\times$ 2 and 4 $\times$ 4 blocks have gains in robustness to 2 to 1 and 4 to 1 decimation over the single pixel segmentation. It is stressed that modulating the lower frequency space (low DCT coefficients for example) of a block with a single key is vulnerable to blind message estimation.

3.2 Method

We split the message $m$ and corresponding rows of $U$ into $L$ segments of size $N/L$ as written by

$$m = \begin{bmatrix} m_0 \\ \vdots \\ m_{L-1} \end{bmatrix}$$

and

$$U = \begin{bmatrix} u_{0,0} & u_{0,1} & \cdots & u_{0,M-1} \\ u_{1,0} & u_{1,1} & \cdots & u_{1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{L-1,0} & u_{L-1,1} & \cdots & u_{L-1,M-1} \end{bmatrix}.$$  (32)

Then each segment is assigned an independent key vector $k_i = (k_{i,0}, k_{i,1}, \cdots, k_{i,M-1})$ ($l = 0, \cdots, L-1$). We need to partition the signature matrix $V$ according to $U$.

$$V = \begin{bmatrix} v_{0,0} & v_{0,1} & \cdots & v_{0,M-1} \\ v_{1,0} & v_{1,1} & \cdots & v_{1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ v_{L-1,0} & v_{L-1,1} & \cdots & v_{L-1,M-1} \end{bmatrix}.$$  (33)

Sub-vectors of $V$ is generated from the carrier matrix by

$$v_{l,j} = u_{l,\phi(l,j)}.$$  (34)

where $\phi(l,j) \in \{0, 1, \cdots, M-1\}$ and $\phi(l,j) \neq \phi(l,i)$ for $j \neq i$. The index $\phi(l,j)$ is determined such that

$$k_{i,j}v_{l,j}m_l > 0.$$  (35)

In decoding we compute

$$\hat{m_l} = \text{sgn}(Y_lk_l),$$  (36)

where $\text{sgn}(\cdot)$ applies element-wise. $Y_l$ is a sub-matrix of $Y$ given by

$$Y_l = [ y_{l,0} \ y_{l,1} \ \cdots \ y_{l,M-1} ].$$  (37)

Now the whole process proceeds in parallel on the segment-basis and hence expansion of required observations is restricted to the segment size.

3.3 Spatial Synchronization

The same key must be used in decoding as used in embedding. This introduces the need of a synchronization mechanism when segmentation is used. We propose the following decoding scheme where only some candidate keys must be identified in decoding with no strict synchronization.

For decoding the $i$-th element of the $l$-th message segment, we first decide on the index of the key to be used by

$$j = \arg \max_{k \in \mathcal{N}(l,i)} |Y_lk_l|.$$  (38)

where $\mathcal{N}(l,i)$ is a set of candidate keys and $(\cdot)_i$ indicates the $i$-th element of a vector. Then the message is decoded using this key by

$$\hat{\text{sgn}}(Y_lk_l)_i.$$  (39)

The set $\mathcal{N}(l,i)$ should be determined so as to contain the correct key with high probability. Usually the set consists of the most probable key and its spatially neighboring keys. In the extreme case, it can be the whole set of keys used in embedding. As shown in the next section, however, both the errors and decoding time increase as the size of the set becomes large. It is thus preferable to keep its size at the
minimum.

4. Simulation Results

In this section, the message is assumed to be a two-dimensional binary image. The cover and the signature are both sets of images or frames imitating video and have been generated from i.i.d. random variables having normal distribution of \( N(0, \sigma_v^2) \) and \( N(0, \sigma_c^2) \) respectively. The C/S ratio is defined in decibel by

\[
C/S = 10 \log_{10} \frac{\sigma_c^2}{\sigma_v^2} = 10 \log_{10} \sigma_c^2, \tag{40}
\]

where we used \( \sigma_v^2 = 1 \) in the last equality and will use this value hereafter.

4.1 Message Secrecy

Message secrecy of the permutation modulation has been evaluated for \( \sigma_x = 0 \) (\( C/S = -\infty \)) and varying values of \( N \) and \( M \). No segmentation is applied in this section. Results have been compared with corresponding SS modulation of equivalent detectability. The C/S ratio of the SS modulation used in the simulation is summarized in Table 1.

Figure 3 shows decoded messages for the permutation modulation with \( N = 32 \times 32 = 1024 \) and \( M = 20000 \). The message has been generated as an image with a white square of a quarter size of the image located in the center of a black background. The image in (a) on the left is a detection result from an unmodulated carrier signal. This is random since no signature has been embedded yet. The image in (b) on the center is a detection result from a modulated carrier and the correct key. Although some random noise accompanies, the message is clearly visible showing that the embedded information is recoverable. The image in (c), which resulted when using the blind message recovery in [13], is random again indicating that it failed to extract the hidden message. Figure 4 shows corresponding images from the SS modulation. We notice that the blind recovery successfully extracted the hidden message as shown in (c).

![Fig. 3 Decoded messages for permutation modulation](image1)

![Fig. 4 Decoded messages for SS modulation](image2)

Note that the sign of the recovery is arbitrary.

We compared them in term of error probability of blind recovery estimation for various values of \( N \) and \( M \). Figure 5 plots the estimation error for increasing number of observations. For the SS modulation, the message has been recovered with less than 1000 observations for \( N = 4 \times 4 \). As \( N \) goes higher, or embedding strength gets weaker, the blind recovery becomes more difficult, but even with \( N = 32 \times 32 \), the message has been recovered with 20000 observations. On the contrary, for the permutation modulation, we see that the estimation error always stays close to 0.5 indicating that nothing about the message has been extracted for any tested values of \( N \) and \( M \).

4.2 Message Detection from Noisy Signals

Next we verified that the message is detected from a stego signal with non-zero cover data. Segmentation has also been introduced using \( N/L = 1, 4 \) and 16 corresponding to blocks of \( 1 \times 1 \), \( 2 \times 2 \) and \( 4 \times 4 \) respectively. A signature of size \( 256 \times 256 \) has been added to the cover with \( C/S = 10 \)dB. Perfect synchronization is assumed and hence the keys are correctly matched in encoding and decoding. Figure 6 shows decoded images in different segmentation sizes for \( M = 16, 64 \) and 256. It is obvious that the images are clearer for the \( 1 \times 1 \) segment than for the other larger segments. This has been expected from Sect. 2.3. We need 4 and 16 times more frames for the \( 2 \times 2 \) and \( 4 \times 4 \) segments to get a comparable image to the single pixel segmentation.

Figure 7 estimates detection performance in bit error rate (BER) where a uniform white image has been used for the message and the number of pixels decoded to black has been counted as errors. We also varied the C/S ratio between \(-\infty \) and 10dB. We see that the BER decreases as the number of observations increases. Also as expected we see more errors for the larger C/S ratios. It can be verified numerically that in any case the single pixel segmentation performs better than the \( 2 \times 2 \) and \( 4 \times 4 \) segmentations.

4.3 Robustness to Sub-Sampling

Although the single pixel segments have performed best so far, this is not the case when the stego image is decimated to a smaller size. Figure 8 shows the BER when a uniform white is embedded with \( C/S = 10 \)dB and has been decimated by \( 2:1 \), \( 4:1 \) and \( 8:1 \) in both horizontal and vertical directions. Comparing them with Fig. 7(c), we see that the
BER increases rapidly for the $1 \times 1$ segments as the size becomes smaller making it come behind the $2 \times 2$ and $4 \times 4$ segments in almost all cases. The $2 \times 2$ segments are the most robust for 2:1 decimation while the $4 \times 4$ segments perform best for 4:1 and further decimation. Note that the BER reduction is partly due to effect of majority voting induced by low-pass filtering of spatially redundant information in
segments during the decimation process. This effect cannot be expected in the single pixel segments. Figure 9 depicts two decoded images from 4:1 decimated stego signals for the $1 \times 1$ and $4 \times 4$ segments. C/S ratio was set to 10dB and 256 frames have been accumulated for decoding. We see a clear difference in the contrast of decoding results.

Sampling phase of decimation can affect the decoding performance for segments larger than $2 \times 2$. Peripheral pixels in a segment are more vulnerable to sub-sampling as they are much more mixed with unrelated signals in neighboring segments than internal pixels. This effect is shown in Fig. 10 where BER curves are plotted for two sampling phases at inner and peripheral pixels of $4 \times 4$ segments. We see some gains for the inner phase over the peripheral phase in the case of 4:1 decimation. However these gains almost diminish for 8:1 decimation because the filter kernel comes to cover a large area extending a single segment.

4.4 Loose Synchronization

Finally we evaluate the effect of loose synchronization. When the synchronization is not perfect, the decoding error increases as shown in Fig. 11, where a uniform white message has been segmented into $4 \times 4$ blocks with no cover signal added ($C/S = -\infty$) and keys of adjacent 8 ($\pm 1$ neighbors) and 24 ($\pm 2$ neighbors) blocks have been used for decoding together with the true key. The degradation is not negligible. Currently keys are selected pixel-wise independently according to Eq. (38). However keys are not random. The error would decrease if spatial structures of the key assignment are utilized. We can correct the key selection if neighboring pixels have chosen different keys, for example.

5. Discussion on Practical Issues

Applying our method to real video is not immediate. Carrier selection and practical modulation schemes must be investigated. In this paper we only discuss these problems in order and present a possible approach toward practical implementation.

Some recent steganographic techniques choose as carrier the noises introduced in information-reducing processes and use the original contents to the embedder’s advantage as side information. The wet paper code in [15] modulates the quantization noise in JPEG compression so that only coefficients in half way between quantization intervals are modified. Since the original image is available only to the embedder and the embedding distortion has been minimized based on this information, it is difficult for the adversary to identify which coefficients have been modified. A similar idea is
disclosed in [16], where the sensor noise is used as a carrier. They shoot the same scene twice with different ISO settings, produce a signature that mimics the difference between two images and add it to the one with lesser noise. The resultant stego is hard to distinguish from any images taken at a single ISO value.

We would be able to borrow the carrier from those techniques described above. In both cases the carrier is the noise inherent in the video which is introduced in critical processes such as acquisitions and compressions. The carrier is modulated using information available at the time of processing but discarded afterward. Such signatures are expected to be much harder to remove than those created with no side information. In addition if the fact of embedding is not visible, the adversary would need to take a risk to degrade non-watermarked content trying to remove nonexistent watermarks unless he is sure that the content has no watermark.

In modulation we must take into account that such carriers may not be stationary. In fact, sensor noises are luminance dependent [17]. Quantization noises vary locally due to adaptive control of quantization step sizes [18]. The non-stationarity of the carrier makes the permutation extremely difficult since it involves exchange of data in different positions which may have different properties. Instead of compensating the local differences in permutation, we are tempted to simply replace the permutation by flipping of the sign of the carrier in part. This flipping is performed so that the relationship in Eq. (5) holds frequently. Sign flipping, when performed randomly, does not change the first order statistics of the carrier if its distribution is symmetric. Specific algorithms of flipping have to be studied.

If we use the sensor noise as carrier we can watermark the content in the spatial domain. Using the quantization noise instead we can do it in the compression domain. For a uncovered situation where the compression follows the spatial embedding, we expect that the robustness to down-sampling would extend without modification to compressions since lower frequencies are usually better conserved in popular compression standards [18].

Finally we give a primitive but interesting calculation concerning distortion in practical situations. Note that distortion and detectability are in the trade-off relationship in general. Assume that video signals are quantized with 8 bits and that they are distributed uniformly in the range of [0, 255]. If we add a signature of $\sigma^2_v = 1.0$ to this signal, the distortion is 48 dB in PSNR (Peak-to-peak Signal-to-Noise Ratio), which is usually almost imperceptible to the eye. The variance of the video signal is computed as $\sigma^2_v = \frac{1}{255} \int_0^{255} (x - \frac{255}{2})^2 dx \approx 5.42 \times 10^3$, which is much larger than $\sigma^2_w = 10$ used in the experiments. Using Eq. (25) we can compute the number of frames required to get comparable results in Fig. 6 as $M = 256 \times 1.5 \times 10^2 \approx 1.26 \times 10^5$ frames for the right most images of the figure. This accounts for 70 minutes for 30 frames per second video or 87 minutes for 24 frames per second cinema. Whether this is practical may depend on applications but in case that it is too many, a remedy would be to repeat embedding the same message in each frame. If the size of message is $64 \times 64$ and the video is HDTV, we can have $30 \times 16 = 480$ repetitions per frame. This would reduce the frame accumulation to $\frac{126 \times 10^5}{480} \approx 263$ frames or about 9 seconds for video and 11 seconds for cinema.

6. Conclusion

A permutation-based method of generating spread-spectrum watermarking has been proposed. Permuting random matrices along the time axis generates a signature volume containing a specific message. We can embed binary images such as emblems and logos in a video. Decoding is performed as in the usual spread-spectrum watermarking. Message secrecy has been verified as robustness to the blind message recovery algorithm. The robustness comes from the permutation modulation used in message embedding. The blind algorithm fails since the generated signature is not distinguished from the Gaussian distributed random noise. Message detectability has been discussed in comparison to the conventional spread-spectrum modulation. The proposed method needs more frames to decode a comparable image if the embedding strength is the same. Length of required observation is proportional to the message size. To alleviate this data expansion problem, we proposed a method to segment the message into smaller blocks. By this, the expansion is limited to the size of a block. We have shown that the proposed method is also robust to spatial downsampling. This comes from the fact that all pixels in a block are modulated with the same key. Hence redundancy of the message is preserved in blocks, so down-sampling only leads to detection of smaller messages. Experiments have shown that message segmentation into $2 \times 2$ or $4 \times 4$ blocks are a reasonable choice in terms of trade-off in data expansion and down-sampling resilience.

We have not considered any temporal synchronization and only assumed that it is perfect. In addition, although we proposed a scheme for spatial synchronization, it needs further improvement to reduce decoding errors. Practical implementation of our method needs further study.

References

[1] T. Mittelholzer, “An information-theoretic approach to steganography and watermarking,” Proc. the 3rd International Information Hiding Workshop, pp.1–16, 1999.
[2] F. Balado and D. Haughton, “Asymptotically Optimum Perfect Universal Steganography of Finite Memoryless Source,” IEEE Trans. Inf. Theory, vol.64, no.2, pp.1199–1216, Feb. 2018.
[3] R. Böhme, Advanced Statistical Steganalysis, Springer, 2010.
[4] J. Fridrich, Steganography in Digital Media, Cambridge University Press, 2010.
[5] J. Fridrich and T. Filler, “Practical Method for Minimizing Embedding Impact in Steganography,” Proc. SPIE Security, Steganography, and Watermarking of Multimedia Contents IX, vol.6505, Feb. 2007.
[6] T. Pevný, T. Filler, and P. Bass, “Using High-Dimensional Image Models to Perform Highly Undetectable Steganography,” Lect.
Hiroshi Ito received the B.S and M.S degrees in Electrical Engineering from Kyoto University, Japan in 1981 and 1983 respectively. In 1983, he joined Imaging Systems Laboratory, Mitsubishi Electric Corporation, Japan. From 1994 to 1995, he was a visiting research assistant at University of Maryland, MD, USA. Since 2008, he has been with College of Industrial Technology, Nihon University. His research interest involves digital signal processing and information security.

Tadashi Kasezawa received the B.S degree in communication engineering and the Ph.D. degree in information science from Tohoku University, Japan in 1983 and 1998, respectively. He is currently a professor of the Department of Computer Science, College of Engineering, Nihon University. His research interest involves signal processing, machine learning and computer vision.