Multimed Tools Appl
DOI 10.1007/s11042-009-0440-3

A more secure steganography based on adaptive pixel-value differencing scheme

Weiqi Luo · Fangjun Huang · Jiwu Huang

Abstract Pixel-value differencing (PVD) based steganography is one of popular approaches for secret data hiding in the spatial domain. However, based on extensive experiments, we find that some statistical artifacts will be inevitably introduced even with a low embedding capacity in most existing PVD-based algorithms. In this paper, we first analyze the common limitations of the original PVD and its modified versions, and then propose a more secure steganography based on a content adaptive scheme. In our method, a cover image is first partitioned into small squares. Each square is then rotated by a random degree of 0, 90, 180 or 270. The resulting image is then divided into non-overlapping embedding units with three consecutive pixels, and the middle one is used for data embedding. The number of embedded bits is dependent on the differences among the three pixels. To preserve the local statistical features, the sort order of the three pixel values will remain the same after data hiding. Furthermore, the new method can first use sharper edge regions for data hiding adaptively, while preserving other smoother regions by adjusting a parameter. The experimental results evaluated on a large image database show that our method achieves much better security compared with the previous PVD-based methods.

Keywords Adaptive data hiding · Pixel-value differencing (PVD) · Security

W. Luo · F. Huang · J. Huang (✉)
School of Information Science and Technology, Sun Yat-Sen University,
Guangzhou, 510275, Peoples' Republic of China
e-mail: isshjw@sysu.edu.cn

W. Luo
Key Lab of Network Security and Cryptology, Fujian Normal University,
Fuzhou, 350007, Peoples' Republic of China
e-mail: weiqi.luo@yahoo.com

Published online: 09 January 2010

1 Introduction

Steganography is a technique for covert communication. It aims to send a secret message by means of embedding them in multimedia such as digital image imperceptibly. Two important properties, undetectability and embedding capacity, should be considered when designing a steganographic algorithm. There is a tradeoff between the two properties. Usually, the more bits you embed into the cover image, the more detectable traces will be introduced in the stego image, which makes it possible to be attacked by some steganalysis algorithms. In many applications, the most important requirement for steganography is the undetectability. That is, the stego images should be visually and statistically similar to the corresponding cover images.

Based on the embedding domains, image steganographic algorithms can be classified into two types, that is those embedding in the spatial domain, such as LSB (Least Significant Bit) based [1, 20] approaches and PVD-based [22, 25] approaches, and those embedding in the transform domain, such as F5 [21] and outguess [12]. In this paper, we focus on an adaptive and secure steganography in the spatial domain.

The LSB-based steganography is one of famous approaches in the spatial domain, in which the least significant bits of a cover image that along a pseudo-random route are changed according to the secret bit stream to be embedded. Those methods regard all pixels within an image can tolerate equal amounts of changes without causing visual artifacts to an observer. However, this is not true especially for the images with more smoother and/or regular regions.

Based on the fact that our human vision is sensitive to slight changes in the smooth regions, while can tolerate more severe changes in the edge regions, the PVD-based methods have been proposed to enhance the embedding capacity without introducing obvious visual artifacts into stego images. In PVD-based schemes, the number of embedded bits is determined by the difference between the pixel and its neighbor. The larger the difference amount is, the more secret bits can be embedded. Usually, PVD-based approaches can achieve more imperceptible results compared with those typical LSB-based approaches with the same embedding capacity. However, based on extensive experiments and analysis, we find that most existing PVD-based algorithms perform bad to resist some statistical analysis even with a low embedding capacity, e.g. 10% bpp (bit per pixel). Some possible reasons that cause the weak security performances are shown as follows:

- Only horizontal differences (i.e. vertical edges) that along a fixed travel route, such as in the raster scanning order, are used for data hiding. Therefore, tracking the pixels in each separate embedding unit become possible for detectors.
- A non-adaptive quantization for the difference of pixel values in the embedding unit will introduce undesired steps into the PVD histogram. Moreover, the sort relationships of most adjacent pixels are inevitably destroyed after data hiding, which may alter the local statistical features of cover images.
- The embedding positions (pixel-pairs) in the existing PVD-based methods are determined by a secret key just as it did in the LSB-based schemes, which means that those methods still embed a lot of secret bits into smooth regions in a cover image even the differences between adjacent pixels equal to zero (flat regions), while many available edge regions have not been fully exploited.

To overcome the limitations as mentioned above, we propose a novel steganography scheme to enhance the security as well as the visual quality of the stego images. Compared with the previous PVD-based approaches, our proposed method can preserve the sort order of the pixel values in all the embedding units. What is more, the new method can make full use of the content edges adaptively according to the relationship between the edge regions and the size of the message to be embedded, namely first uses sharper edges for data hiding, and then uses the smoother ones, until all the message are embedded completely. The experimental results evaluated on around 4,000 natural images using three special and four universal steganalysis algorithms show the effectiveness of our proposed method.

The rest of the paper is arranged as follows. Section 2 analyzes the properties and limitations in the existing PVD-based approaches, and then discusses some improvement strategies. Section 3 describes the details of our proposed algorithm, including the data embedding and data extraction. Section 4 shows the experimental results and discussions. Finally, the conclusion and future work will be given in Section 5.

2 Analysis on PVD-based steganography

2.1 Overview of PVD-based steganography

In the original PVD scheme proposed by Wu and Tsai [22], the procedure of data embedding is shown as follows.

- **Step 1** A cover image is first rearranged as a row vector by running thought all rows in a raster scanning manner. The vector is then divided into non-overlapping 1-by-2 embedding units. For each unit, say $[g_i, g_{i+1}]$, a difference d is calculated by $d = g_{i+1} - g_i$, where $g_i, g_{i+1} \in [0, \ldots, 255]$.
- **Step 2** And then the absolute difference $|d|$, where $|d| \in [0, \ldots, 255]$, is classified into a number of contiguous ranges denoted as R_i, where $i = 1, 2, \ldots n$. The lower bound, upper bounds and the width of region R_i are denoted as l_i, u_i and w_i, respectively. A typical setting of the regions is that [0 7], [8 15], [16 31], [32 63], [64 127] and [128 255]. Assuming $|d|$ belongs to region R_k, here $k \in [1, 2 \ldots 6]$.
- **Step 3** Determine the number of embedded bits by

$$n = \lfloor log_2(w_k) \rfloor$$

  then select the next sub-stream with n bits from the secret message, and transfer them into a decimal value b.
- **Step 4** Calculate the new difference d' by

$$d' = \begin{cases} l_k + b & for \quad d \geq 0 \\ -(l_k + b) & for \quad d < 0 \end{cases}$$

  Then the new gray values (g'_i, g'_{i+1}) are computed by

$$(g'_i, g'_{i+1}) = \begin{cases} \left(g_i - \left\lceil \frac{d' - d}{2} \right\rceil, g_{i+1} + \left\lfloor \frac{d' - d}{2} \right\rfloor\right) & d \in odd \\ \left(g_i - \left\lfloor \frac{d' - d}{2} \right\rfloor, g_{i+1} + \left\lceil \frac{d' - d}{2} \right\rceil\right) & d \in even \end{cases}$$

If g'_i or g'_{i+1} is out of the range [0,255], then the candidate embedding unit is marked as abandoned one. Note that such unused units are few in most natural images and are very easy to detect.

It can be proven that the new absolute difference $|d'| = |g'_{i+1} - g'_i|$ in the stego image will fall into the same region R_k as the difference $|d| = |g_{i+1} - g_i|$ in the cover image. So in data extraction, if $|d'| \in R_k$, the embedded value can be extracted correctly by $b = |d'| - l_k$.

In [25], the authors shows that a fixed region R_i employed in the original PVD scheme (**Step 2**) will introduce some unusual steps in the histogram of pixel value differences (PVD histogram) for all the embedding units, which can be used to expose the presence of hidden message and further to estimate the size of hidden bits. To make the steganography immune to PVD histogram based analysis, the authors employ random regions instead of the fixed regions when data hiding. The experimental results in [25] show that the new approach can effectively eliminate those undesired steps.

To improve the embedding capacity, a new PVD-based steganography combined with LSB technique is proposed by Wu et al. [23]. In this method, **Steps 1** and **2** are the same as it did in the original PVD scheme. After that, if R_k belongs to higher level, namely $|d| > 15$, then embeds secret bits using the original method. Otherwise if $|d|$ belongs to lower level, namely $|d| \leq 15$, embeds three bits into g_i and g_{i+1} using 3-LSB replacement, respectively. Assume that the pixel values after data hiding are denoted as g'_i and g'_{i+1}, then calculate $|d'| = |g'_{i+1} - g'_i|$, if the new difference $|d'|$ is not belongs to lower level, then readjust them by

$$(g'_i, g'_{i+1}) = \begin{cases} (g_i - 8, g_{i+1} + 8) & g'_i \geq g'_{i+1} \\ (g_i + 8, g_{i+1} - 8) & g'_i < g'_{i+1} \end{cases}$$

In a latest work [24], it divides the pixel differences into three levels: lower-level, middle-level and high-level. Similar to Wu's approach [23], k secret bits are embedded into g_i and g_{i+1}, respectively, where $k = l, m, h$, if $|d|$ belongs to lower-level, middle-level and high-level, respectively. After that the method applies the modified LSB approach [1, 5] to the resulting pixels pair and gets the new pixel pair (g'_i, g'_{i+1}). If the new difference $|d'| = |g'_{i+1} - g'_i|$ belongs to different levels, readjust them into the following forms:

$$(g'_i, g'_{i+1} \pm 2^k) \quad or \quad (g'_i \pm 2^k, g'_{i+1})$$

Finally selects the better choice of the new values g'_i, g'_{i+1} which satisfies the conditions that the new difference $|d'|$ and the original one belong to the same level and the value of $|g'_i - g_i| + |g'_{i+1} - g_{i+1}|$ is the smallest among the candidates. Compared with [23], the new method can provide stego image with larger embedding capacity as well as higher objective quality.

2.2 Properties of PVD-based steganography

As described in Subsection 2.1, the main idea of the existing PVD-based approaches is that they first divide a cover image into non-overlapping embedding units with two consecutive pixels in a raster scanning manner, and then deal with the embedding units separately in a pseudo-random order. For a given embedding unit, the

difference is first calculated, and then the difference is classified into one of several regions (random or fixed). Usually, pixel pairs located at the edge regions (with larger difference) are embedded more secret bits than those located at smooth regions (with smaller differences). The embedding strategies may be different according to different steganographic approaches and/or the regions that the differences belong to. In order to guarantee the validity of data extraction, the difference in each embedding unit must belong to the same region after data hiding. Otherwise, those approaches need to readjust them into new ones or marked them as unused units.

Based on the characteristics of HVS (human visual system), the original PVD approaches [22] can embed more secret bits into an image with fewer visual artifacts. Up to now, several modified approaches i.e. [23, 24] have been proposed to enhance the embedding capacity and/or improve the objective quality of the stego images. However, just a few works [25] address the security issues of the embedding schemes.

In the following Subsection 2.3, we will first analyze the common limitations in the existing PVD-based approaches, and then propose a novel and secure scheme based on the modified PVD with more adaptability to the image contents in the next Section 3.

2.3 Limitations analysis and strategies

In order to enhance the visual quality and the security of stego images, the three following limitations in the existing PVD-based approaches should be carefully investigated and have been improved in our proposed method.

- In the existing PVD-based methods, a raster scanning order is typically employed for dividing the embedding units, which means that only vertical edges can be used. At the same time, such fixed division will leave some revealing clues for detectors. For instance, we can analyze some statistical artifacts in the PVD histogram to expose the stego images and estimate the size of hidden data [25]. To overcome the limitation, our proposed method first divides the cover images into non-overlapping squares, and rotates them by a degree of 0, 90, 180 or 270 randomly. In such a way, both horizontal and vertical edges in images can be used. What is more, tracking the embedding units from the stego images becomes impossible if without the rotation key, which makes the detection more difficult.
- Most PVD-based approaches employ fixed contiguous ranges to classify the differences between the pixel values in the embedding unit, which will lead to undesired steps appearing at the PVD histogram. In [25], the authors show that we can use such un-natural effects to estimate the size of hidden message. To defeat the PVD histogram based steganalysis, the authors in [25] introduce a pseudo-random dithering to the fixed ranges. The experimental results show that the improved approach can avoid the occurrence of the undesire steps effectively. However, we will show that later such method is also vulnerable to some other statistical analysis. To enhance the security, we divide a cover image into non-overlapping embedding units with three consecutive pixels, and the middle one is used for data embedding. The number of embedded bits is depended on the relationships of the three pixel values and an optional threshold T. Besides that, the sort relationship of the pixels in each embedding unit will be well preserved after data hiding.

– Based on the characteristics of HVS and our extensive experiments, embedding secret data into the edge regions can not also produce more visual indistinguishable results but also enhance the robustness against the statistical analysis. Therefore we should make full use of those edge regions as far as possible. Typical PVD-based approaches can embed more secret bits into busy areas. However, lots of smooth regions will also be contaminated after data embedding even the difference of the two consecutive pixel values is zero, for instance, 6 bits are embedded using the method [23], while many available edges regions have not been fully exploited. To overcome the limitation, we employ an optional threshold T to measure the edge strength for each embedding unit. When performing data embedding, our method can first estimate whether the embedding units whose strength larger than T are enough for a given secret message. If not, the method decreases the threshold T to $T-1$ to release more spaces, until all the secret message can be embedded completely. In such a way, most smooth/falt regions in cover images can be well preserved. In this paper, T decreases from 32 to 1. Therefore, only 5 ($2^5 = 32$) bits additional information is needed. Please note that the threshold T is dependent on image contents and the secret message to be embedded. Thus we need the threshold T as side information for subsequent data extraction. In practice, we can embed T into a preset region that does not be used for data hiding using some existing steganographic approaches, such as the simple LSB replacement.

3 Proposed method

3.1 Data embedding

– **Step 1** The cover image is first divided into non-overlapping squares with size $Bz \times Bz$, where Bz is a multiple of 3, in this paper, $Bz = 6$. For each square, we rotate it by a random degree of 0, 90, 180 and 270 based on a secret key key_1, which produces a random degree sequence with size of $[\frac{M}{Bz}][\frac{N}{Bz}]$, where $M \times N$ is the size of the cover image.
– **Step 2** The resulting image is rearranged as a row vector running through all the rows in a raster scanning manner as it did in the paper [22]. And then the vector is partitioned into non-overlapping embedding units with three consecutive pixels, say $[g_i, g_{i+1}, g_{i+2}]$.
– **Step 3** For a given embedding unit, calculate two differences of the adjacent pixel values by

$$d_1 = g_{i+1} - g_i, d_2 = g_{i+2} - g_{i+1}$$

If $|d_1| \leq T$ and $|d_2| \leq T$,[1] then skip to the next embedding unit in a pseudo-random travel order based on a secret key key_2.
Otherwise, if $|d_1| > T$ or $|d_2| > T$, which means that the middle pixel g_{i+1} is

[1] T is a threshold, the initialize setting of T is 32 in this paper.

located at the content edges whose strength degree is larger than T, then g_{i+1} can be used for data embedding:

Determine the range of centre pixel in the embedding unit, denoted as $range_{g'_{i+1}}$, which satisfies that the sort order of g_i, g'_{i+1}, g_{i+2} in the stego is the same as the sort order of g_i, g_{i+1}, g_{i+2} in the cover, also the relationships between d_1, d_2 (d'_1, d'_2) and T are preserved after data hiding. For instance, if $g_i \geq g_{i+1} \geq g_{i+2}$ in the cover, then we have $g_i \geq g'_{i+1} \geq g_{i+2}$ in the stego, if $|d_1| = |g_{i+1} - g_i| > T$, then we keep $|d'_1| = |g'_{i+1} - g_i| > T$.

The range can be obtained according to the following four cases:

- **Case 1** $g_{i+1} < g_i$ and $g_{i+1} < g_{i+2}$,

  case 1.1: if $|d_1| > T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[0, \ldots, min\left(g_i - T - 1, g_{i+2} - T - 1\right)\right]$$

  case 1.2: if $|d_1| > T$ & $|d_2| \leq T$

  $$range_{g'_{i+1}} = \left[max\left(g_{i+2} - T, 0\right), \ldots, min\left(g_i - T - 1, g_{i+2} - 1\right)\right]$$

  case 1.3: if $|d_1| \leq T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[max\left(g_i - T, 0\right), \ldots, min\left(g_{i+2} - T - 1, g_i - 1\right)\right]$$

- **Case 2** $g_{i+1} > g_i$ and $g_{i+1} > g_{i+2}$,

  case 2.1: if $|d_1| > T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[max\left(g_i + T + 1, g_{i+2} + T + 1\right), \ldots, 255\right]$$

  case 2.2: if $|d_1| > T$ & $|d_2| \leq T$

  $$range_{g'_{i+1}} = \left[max\left(g_i + T + 1, g_{i+2} + 1\right), \ldots, min\left(g_{i+2} + T, 255\right)\right]$$

  case 2.3: if $|d_1| \leq T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[max\left(g_{i+2} + T + 1, g_i + 1\right), \ldots, min\left(g_i + T, 255\right)\right]$$

- **Case 3** $g_i \geq g_{i+1} \geq g_{i+2}$

  case 3.1: if $|d_1| > T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[g_{i+2} + T + 1, \ldots, g_i - T - 1\right]$$

  case 3.2: if $|d_1| > T$ & $|d_2| \leq T$

  $$range_{g'_{i+1}} = \left[g_{i+2}, \ldots, min\left(g_{i+2} + T, g_i - T - 1\right)\right]$$

  case 3.3: if $|d_1| \leq T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[max\left(g_{i+2} + T + 1, g_i - T\right), \ldots, g_i\right]$$

- **Case 4** $g_i \leq g_{i+1} \leq g_{i+2}$

  case 4.1: if $|d_1| > T$ & $|d_2| > T$

  $$range_{g'_{i+1}} = \left[g_i + T + 1, \ldots, g_{i+2} - T - 1\right]$$

case 4.2: if $|d_1| > T \ \& \ |d_2| \le T$

$$range_{g'_{i+1}} = \left[max\left(g_{i+2} - T, g_i + T + 1\right), \ldots, g_{i+2}\right]$$

case 4.3: if $|d_1| \le T \ \& \ |d_2| > T$

$$range_{g'_{i+1}} = \left[g_i, \ldots, min\left(g_{i+2} - T - 1, g_i + T\right)\right]$$

Please note that the range $range_{g'_{i+1}}$ is independent of the original pixel value g_{i+1} in all cases. And what's more, it can be proven that the sort order and the relationships as mentioned previously will be preserved for all $g'_{i+1} \in range_{g'_{i+1}}$, which is very important to guarantees the validity of data extraction. Please refer to Appendix for more details.

– **Step 4** Calculate

$$n = min\left(\left\lfloor log_2 \left| range_{g'_{i+1}} \right| \right\rfloor, k\right)$$

where $|range_{g'_{i+1}}|$ denotes the number of the elements in the set $range'_{g'_{i+1}}$, k denotes the maximal number of embedded bits into a single pixel g_{i+1}, we limit $k \le 4$. In this work, we set $k = 4$. We can change it to adjust the embedding capacity and visual quality.
If $n > 0$, covert n secret bits into a decimal value b, and calculate

$$g'_{i+1} = \arg\min_e \left\{ \left| e - g_{i+1} \right| \middle| \left| e - g_i \right| \equiv b \left(mod \ 2^n\right), e \in range_{g'_{i+1}} \right\}$$

– **Step 5** Select the next embedding unit based on the key key_2, and repeat above operations from **Step 3**, until all the message are embedded.
If all the available embedding units have been traveled while the secret message has not been embedded completely, decrease the threshold T to $T - 1$, and restart the operations from **Step 2**. If $T = 1$ and the message can not be embedded completely, the cover image has not enough space to embed the given secret message.

– **Step 6** Re-rotate the resulting image based on the block size $Bz \times Bz$ and the secret key key_1. Finally, we obtain the stego image.

For instance, we are dealing with an embedding unit $[g_i, g_{i+1}, g_{i+2}] = [40, 70, 50]$. Assuming that the threshold $T = 10$ and the secret bit stream is $0011001\ldots_{(2)}$. We firstly calculate the differences $d_1 = g_{i+1} - g_i = 30$, $d_2 = g_{i+2} - g_{i+1} = -20$, then we have $g_{i+1} > g_i$ and $g_{i+1} > g_{i+2}$, so **Case 2** is considered. And since $|d_1| > T \ \& \ |d_2| > T$, the range of the centre pixel values can be determined by (**case 2.1**)

$$range_{g'_{i+1}} = \left[max\left(g_i + T + 1, g_{i+2} + T + 1\right), \ldots, 255\right] = [61, \ldots, 255]$$

Then calculate $n = min(\lfloor log_2 195 \rfloor, 4) = 4$ and extract 4 bits from the secret message stream, i.e. $0011_{(2)}$, and then converted it into decimal value $b = 3$. Finally the value of g_{i+1} after data embedding becomes

$$g'_{i+1} = \arg\min_e \left\{ |e - 70| \, | \, |e - 40| \equiv 3 \left(mod \ 2^4\right), e \in [61, \ldots, 255] \right\} = 75$$

It can be checked that the sort order and the relationships of the values $g_i = 40$, $g'_{i+1} = 75$, $g_{i+2} = 50$ and $T = 10$ remain the same, namely

$$g'_{i+1} > g_i, g'_{i+1} > g_{i+2}, \left|d'_1\right| = \left|g'_{i+1} - g_i\right| > T, \left|d'_2\right| = \left|g_{i+2} - g'_{i+1}\right| > T$$

3.2 Data extraction

We first divide the stego image into $Bz \times Bz$ blocks which are then rotated by random degrees based on the secret key key_1, and obtain non-overlapping embedding units with three consecutive pixels just as **Steps 1** and **2** in data embedding. We then travel all the embedding units in a pseudo-random order based on the secret key key_2.

For a given unit, say $[g_i, g'_{i+1}, g_{i+2}]$,[2] we execute the following operations:

First calculate the differences of the adjacent pixel values by

$$d'_1 = g'_{i+1} - g_i, d'_2 = g_{i+2} - g'_{i+1}$$

If $|d'_1| \leq T$ and $|d'_2| \leq T$,[3] then skip to next embedding unit. Otherwise determine the range of g'_{i+1} according to the four cases just as shown in the process in data embedding, also denoted as $range_{g'_{i+1}}$.[4]

Then determine the number of the embedded bits in pixel g'_{i+1} by

$$n = min\left(\left\lfloor log_2 \left| range_{g'_{i+1}} \right| \right\rfloor, k\right)$$

where $k = 4$

And then the embedded value b in decimal representation is

$$b \equiv \left| g'_{i+1} - g_i \right| (mod\ 2^n)$$

Finally convert the decimal value b into the binary representation and obtain the hidden message bits.

For instance, we want to extract secret bits from an embedding unit $[g_i, g'_{i+1}, g_{i+2}] = [40, 75, 50]$ with a threshold $T = 10$. First, calculate the differences $d'_1 = 75 - 40 = 35, d'_2 = 50 - 75 = -25$, then we get $g'_{i+1} > g_i, g'_{i+1} > g_{i+2}$, hence **Case 2** is considered. Since $|d'_1| = 35 > T, |d'_2| = 25 > T$, the range of the centre pixel g'_{i+1} is determined by (**case 2.1**)

$$range_{g'_{i+1}} = \left[max\left(g_i + T + 1, g_{i+2} + T + 1\right), \ldots, 255\right] = [61, 255]$$

So the number of embedded bits is

$$n = min\left(\left\lfloor log_2 \left| range_{g'_{i+1}} \right| \right\rfloor, k\right) = min\left(\left\lfloor log_2 195 \right\rfloor, 4\right) = 4$$

Finally we obtain the secret bits in the decimal representation by

$$b = 3 \equiv 74 - 40 (mod\ 2^4)$$

namely $0011_{(2)}$

[2] Note that only the centre one has been changed after data hiding.

[3] The threshold T can be extracted in a preset region in the stego.

[4] It can be proven that such region is the same before and after data embedding using our embedding scheme. Please refer to Appendix for more details.

4 Experimental results and discussions

In this Section, we present some experimental results to show the effectiveness of our proposed method compared with the existing PVD-based approaches as described in Section 2.1, including the original PVD scheme [22], the improved version of PVD (**IPVD** [25]), the modified PVD scheme combing with LSB approach (**PVD-LSB** [23]) and its enhanced version called **Adaptive-Edge** [24] for short.

4.1 Image database

We employ 3855 natural images from the following three databases:

- **NRCS** We randomly downloaded 1,543 color images with TIFF (Tagged Image File Format) in high resolutions (1,500 × 2,100 or 2,100 × 1,500) from the **NRCS** photo gallery [10]. Those images are then resized to 512 × 768 or 768 × 512 using bicubic method.
- **UCID** [13] The database includes 1,338 uncompressed color image with size of 384 × 512 or 512 × 384.
- **Our Database** We collect 974 uncompressed images taken by our research group. Those images are then reduced to size of 640 × 480.

The test images include (but not limited to) landscapes, people, plants, animals and buildings. In our experiments, all the color images are converted to gray-scale images.

4.2 Subjective quality and embedding capacity

Figure 1 illustrates the binary images which show the differences between the cover image and the stego images using different steganographic schemes with the same embedding capacity of 10% bpp.

From Fig. 1b, it can be clearly seen that our proposed method embeds most secret bits along content edges, and leaves those smooth regions, such as the regions in the sky, as they are. In this way, the subjective quality of the stego images will be better since our human vision can tolerate more amounts of changes in the edge regions. While for the other PVD-based approaches as shown in Fig. 1c∼f, the positions of changed pixels are random, which means that those embedding schemes are not adaptive enough with the image contents. Though more secret bits are hidden in those edge regions using the existing PVD-based approaches, there are still a lot of bits embedded into the smooth regions even the difference of pixel pair is zero, which will lead to some noise-like blocking artifacts in the smooth regions just as illustrated in Fig. 2. Thus the subjective quality of stego images will decline in those regions.

Another important characteristic of our proposed method is that it can adaptively select content edges for data hiding according to a given embedding rate. As shown in Fig. 3a, for a lower embedding rate of 5% bpp, only the sharper edges, whose absolute differences between the adjacent pixel values are larger than $T = 32$, are used while leaving those edge regions whose differences are equal to or smaller than T as they are. When the embedding rate increases, the method can release more

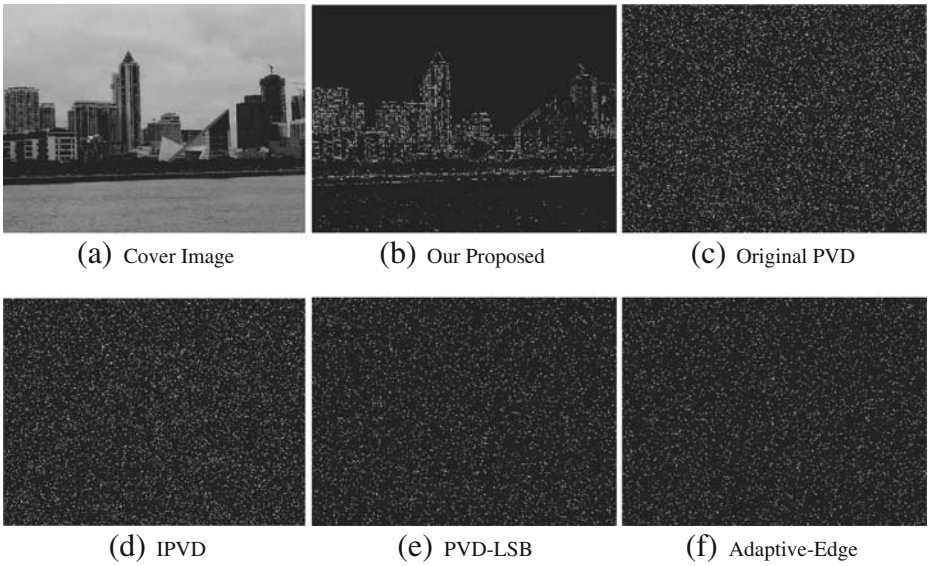
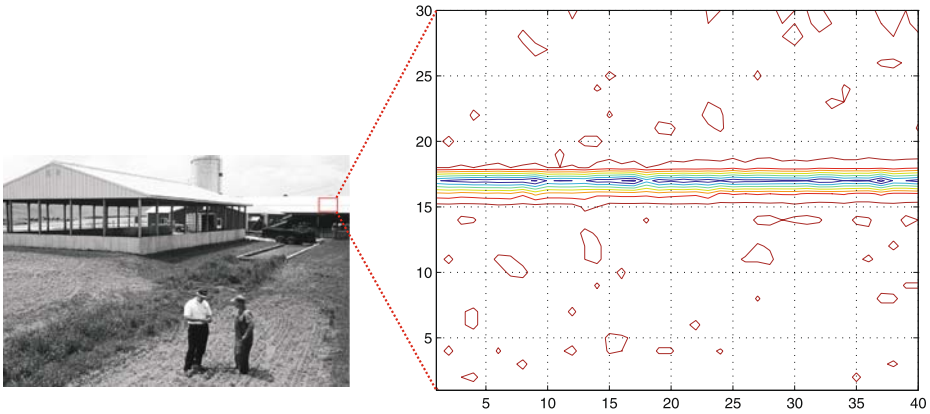


Fig. 1 (**a**) Cover image; (**b**–**f**) show the positions of those changed pixels using different steganographic schemes with the same embedding capacity of 10% bpp. The white points indicate that the pixel values in the corresponding positions are altered after data embedding

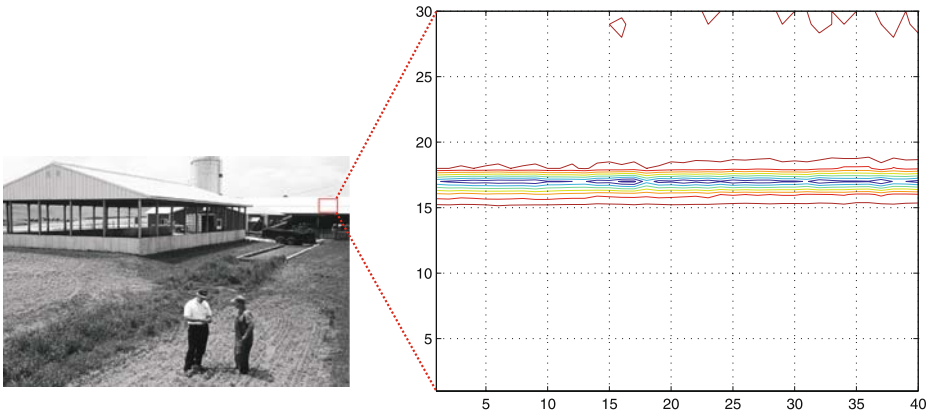
embedding units by decreasing the threshold T, as shown in Fig. 3b–d. Therefore, the embedding capacity of our method highly depends on the edge regions in an image. Usually, for those image with more edge regions, such as Baboon, more secret message can be embedded without being detected easily. For those image with many smooth regions, as shown in Fig. 4, fewer edge regions in the image can be used for secure embedding. In such a case, the available space for data hiding is small using the proposed method. Please note that the maximal embedding capacity is around 16% bpp for this image example with a threshold $T = 1$.

Based on the process of our proposed scheme, 1/3 pixels in an image have been employed for data hiding, and the maximal number of the embedded bits within a single pixel is not larger than k (please refer to **Step 4** in data embedding). Therefore, the upper bound of our embedding capacity for a given image is $k/3$ bpp, and the lower bound is zero, for an extreme instance, an image with a uniform background. In this paper, we set $k = 4$, thus 1.33 bpp is the upper bound. In most situations, we can not reach the maximum value, since we should preserve some relationships of the pixel values in the embedding unit and the threshold T as illustrated in Fig. 8 in the Appendix. Thus, the choice of cover image is very important since it will significantly influence the embedding capacity and security significantly. Usually we should select those images with more textural regions and remove those with large smooth regions such as the examples in Figs. 4a and 5.

Table 1 gives the number and percentage of those images in our test database that have enough space for a given embedding capacity, i.e. 5% bpp, 10% bpp, 20% bpp and 30% bpp, respectively. We can see that most images in the database can be used. In the following experiments, including the objective quality measurement and



(a) Adaptive-edge method



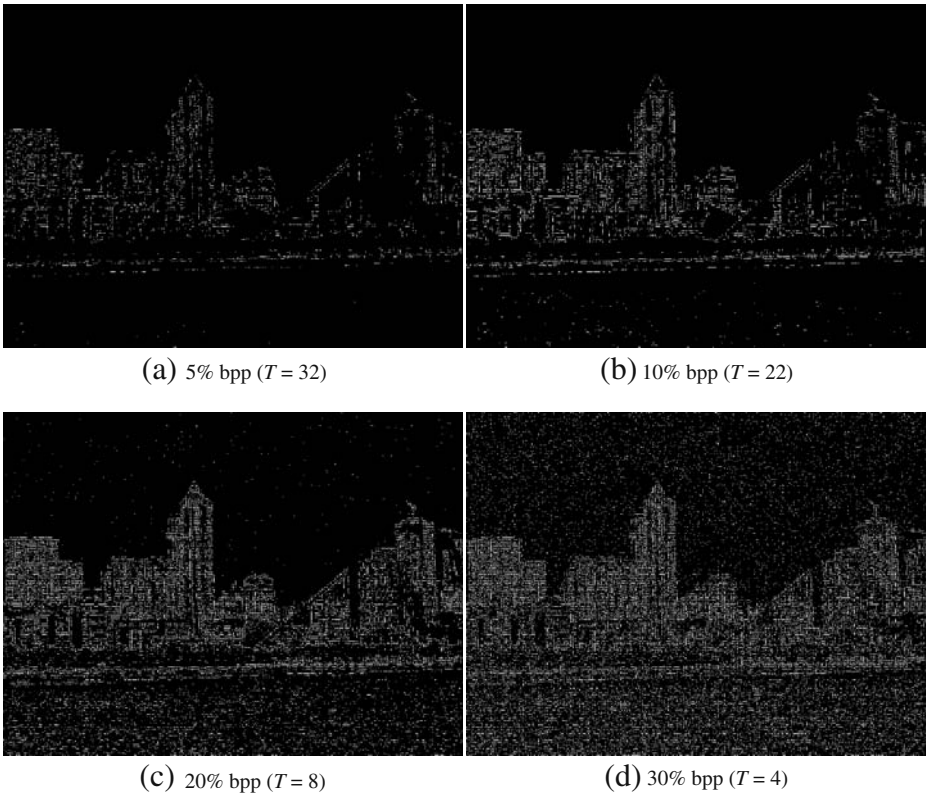
(b) Our proposed method with $T = 4$

Fig. 2 Illustration of the difference between the stego images using adaptive-edge approach [24] and our proposed method with the same embedding capacity of 70% bpp. For display purpose, we shows the contour lines of the zoom-in smooth region in the stego using Matlab with default settings. It can be clearly seen that many local minimal/maximal regions appear after using adaptive-edge approach, which will lead to noise-like blocking effects in the corresponding positions. While our method can leave those regions as they are in the cover image. Example: NRCSMI01029 in NRCS database, Zoom-in location: (96 : 125, 701 : 740)

security evaluation, we employ the same image subset as shown in the table for all the steganography approaches for a fair comparison environment.

4.3 Objective quality

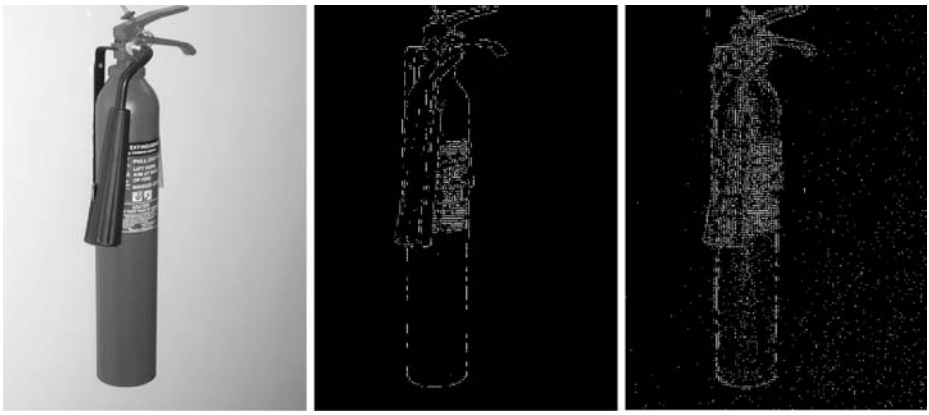
In above subsection, we show that our proposed method can embed most secret bits into edge regions adaptively, and will not produce noise-like blocking artifacts in those smooth regions. Therefore, the subjective quality of the stego images is



(a) 5% bpp ($T = 32$) (b) 10% bpp ($T = 22$)

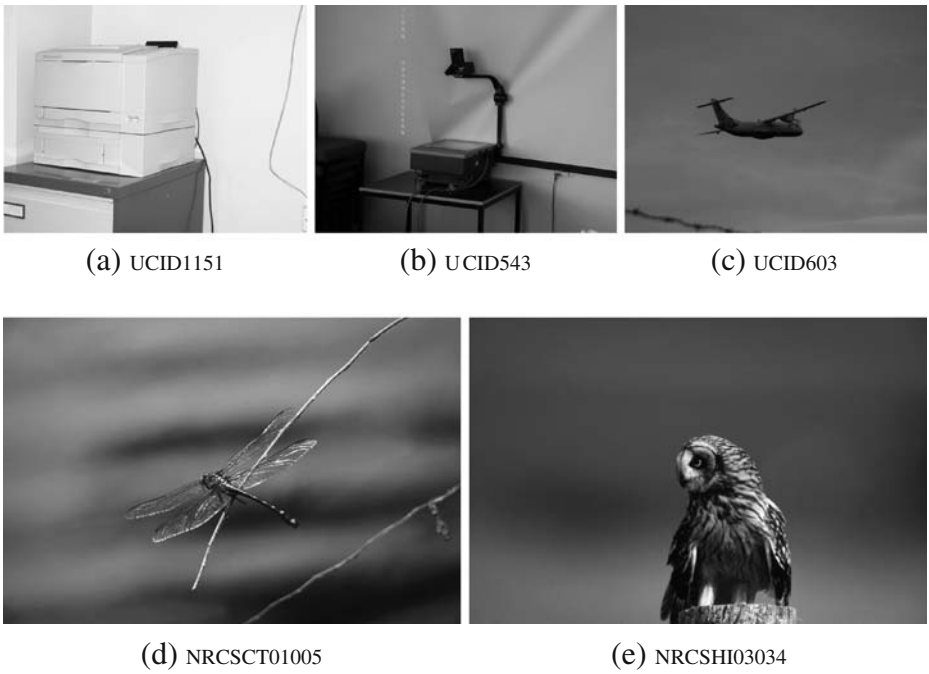
(c) 20% bpp ($T = 8$) (d) 30% bpp ($T = 4$)

Fig. 3 The positions of changed pixels using our proposed method with the embedding capacity of 5%, 10%, 20% and 30% bpp, respectively



(a) Cover Image (b) 5% bpp ($T = 17$) (c) 10% bpp ($T = 2$)

Fig. 4 An image example with a lower embedding capacity



(a) UCID1151 (b) UCID543 (c) UCID603

(d) NRCSCT01005 (e) NRCSHI03034

Fig. 5 Some examples with large smooth regions in databases

better based on the characteristics of HVS. In this subsection, we will present some experimental results about the objective quality.

The PSNR (Peak Signal to Noise Ratio) is most commonly used as a measure of quality, which defined by

$$PSNR = 10log_{10}\frac{max(x)^2}{||x' - x||^2}$$

where x is the cover image and x' is the stego image.

Table 2 gives the average PSNR of the stego images using different steganographic schemes with the embedding rates ranging from 5% bpp to 30% bpp. It is observed that, the IPVD scheme usually obtains the best image quality. Although our stego images have high quality (all over 45 dB), it is still relative lower comparing with the others, decreasing by 3.1 $\sim$ 3.9 dB with respect to the best ones.

The classical PSNR quality metric does not take into account the HVS characteristics. It treats the distortions in all regions in the same way. An adaptive way

Table 1 The number and percentage of images in our test database that can be used for a given embedding capacity

	5% bpp	10% bpp	20% bpp	30% bpp
Number	3,855	3,851	3,831	3,772
Percentage (%)	100	99.99	99.38	97.85

Table 2 Average PSNR for the stego images in different cases

	Proposed	Ori PVD	IPVD	PVD-LSB	Adaptive-edge
5% bpp	52.2024	54.0047	56.1460	53.1465	54.4073
10% bpp	49.3767	50.9887	53.1188	50.1204	51.3946
20% bpp	46.7451	47.9620	50.0929	47.1044	48.3651
30% bpp	45.1625	46.1555	48.2703	45.3062	46.5598

called weighted-PSNR has been proposed to provide a better objective measure of image quality, and it has been adopted as a new quality metric in Checkmark Version 1.2 [11, 18], a famous evaluation software for watermarking technologies. The main idea of the new metric is to assign different weighting into different content regions according to the characteristics of the regions. Usually, the weighting for busy regions is smaller than that of smooth regions since our vision is sensitive to slight changes in the smooth regions, while it can tolerate more severe changes in the busy regions. The modified quality metric is shown as follows.

$$wPSNR = 10log_{10}\frac{max(x)^2}{||NVF(x' - x)||^2}$$

where NVF denotes Noise Visibility Function, which is used for calculating the weighting for all the pixels within the image based on their local variances. Please refer to [16, 17] for more details.

The experimental results are shown in Table 3. It is observed that, with our proposed method, the average wPSNR of the stego images are very close to the stegos using IPVD scheme (decreasing by $0.4 \sim 2.3$ dB), and are better than those using the original PVD, PVD-LSB and Adaptive-Edge schemes in most cases.

4.4 Security evaluation

Security is a very important issue in steganographic communication systems. One of factors that may significantly influence the system security is the embedding capacity. Usually, the more bits we embed into a cover image, the more detectable artifacts will be introduced in the stego image. If there is a detection algorithm that can guess whether a given image is cover or stego with a relative high rate than random guessing, then the steganographic system is broken. In other words, for a given embedding capacity, a steganographic method is considered more secure, if its stegos are more statistically similar to the corresponding covers and its guessing rate is closer to the random guessing than the others.

Table 3 Average weighted-PSNR for the stego images in different cases

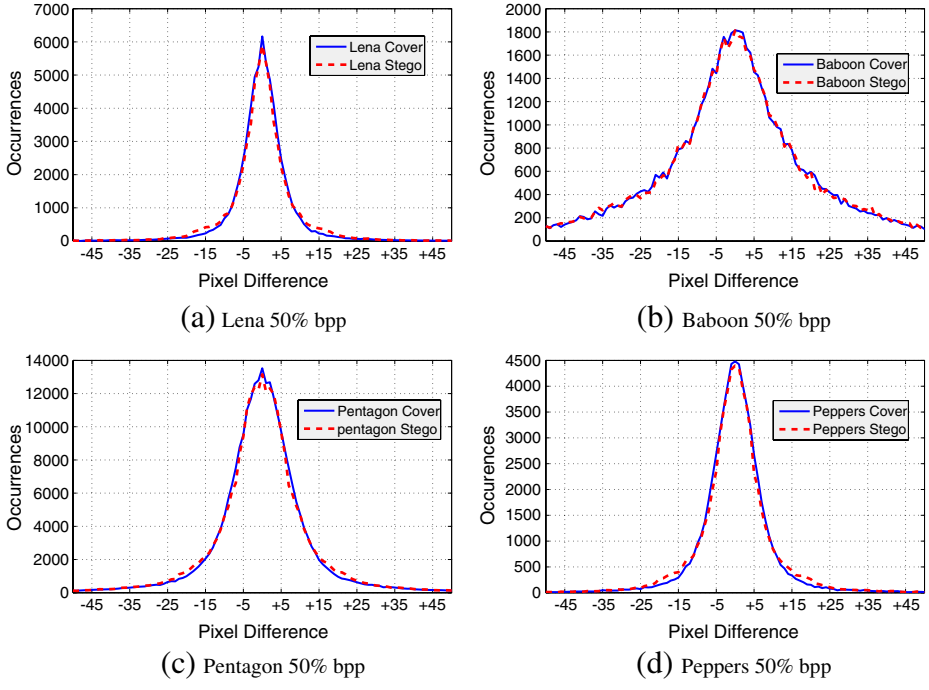
	Proposed	Ori PVD	IPVD	PVD-LSB	Adaptive-edge
5% bpp	61.3478	59.4781	61.7337	56.8888	59.1981
10% bpp	57.9387	56.4724	58.7217	53.8801	56.1852
20% bpp	54.1467	53.4691	55.7140	50.8767	53.1774
30% bpp	51.6324	51.7464	53.9810	49.1496	51.4480

In this subsection, we will evaluate the security of the proposed method and the previous PVD-based approaches in the three following parts, i.e. PVD histogram analysis, LSB matching analysis and universal steganalysis.

4.4.1 PVD histogram analysis

PVD histogram may be a potential characteristic to expose the hidden message of those stegos using the PVD based steganographic methods. In [25], Zhang and Wang showed that the original PVD scheme [22] will inevitably introduce some undesired steps in the histogram of the differences between two continuous pixels in each embedding unit due to its fixed division of embedding units and its fixed quantization steps. By detecting and analyzing such artifacts, it is possible to estimate the size of hidden message, especially when the embedding rate is high.

Since detectors can not identify the embedding units used in our proposed method without a rotation key key_1. And what is more, the number of embedded bits are mainly dependent on the relationship between a pixel value and its two touching neighbors rather than pre-determined values, it is expected that those undesired steps in the original PVD-scheme can be easily avoided in the new embedding scheme. Figure 6 shows the PVD histograms of some cover images and their corresponding stegos using our proposed method with an embedding rate of 50% bpp, respectively. It is observed that the PVD histogram can be well preserved after data hiding.



(a) Lena 50% bpp (b) Baboon 50% bpp

(c) Pentagon 50% bpp (d) Peppers 50% bpp

Fig. 6 PVD histogram differences between covers and theirs corresponding stegos with an embedding rate of 50% bpp

Table 4 shows the mean absolute difference between the PVD histograms before and after data hiding using different steganographic schemes with the embedding rates ranging from 5% bpp to 30% bpp. It is observed that the our differences are the smallest or nearly the smallest among the five methods, which means that our new scheme may have better capacity to resist the PVD histogram analysis.

4.4.2 LSB matching analysis

Since our proposed method embeds secret message by modifying pixel values in an image, it may introduce some artifacts just as the LSB matching scheme did. In this subsection, we employ two special feature sets for detecting the LSB matching to evaluate the security of our method.

- **Li-1D** [7] Calculate the calibration-based detectors (e.g. calibrated HCF COM) as the difference between adjacent pixels within an image. The experimental results in [7] shows that the method outperforms the previous calibrated HCF COM methods in [4].
- **Huang-1D** [3] Calculate the alteration rate of the number of neighborhood gray levels. Unlike the HCF COM based methods [4, 7], it detects the statistical changes of those overlapping flat blocks with 3×3 pixels in the first two bit planes after re-embedding operations.

The ROC (Receiver operating characteristic) curves for the experimental results are shown in Fig. 7. It is observed that both special feature sets will fail in detecting the five PVD-based schemes, especially for our proposed method (getting closer to the random 50% guessing).

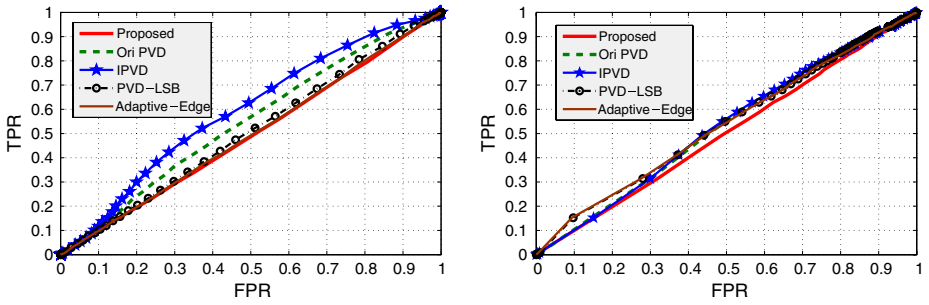
4.4.3 Universal steganalysis

In the subsection, four universal steganalytic feature sets are employed to evaluate the security of the proposed method and the existing PVD-based ones.

- **Shi-78D** [14] The statistical moments of characteristic functions of the prediction-error image, the test image and their wavelet subbands are employed to reflect the differentiation property of the associated histogram between cover and stego images. (78 Dimension)
- **Farid-72D** [2] The higher-order statistical moments taken from a multi-scale decomposition, which includes basic coefficient statistics as well as error statistics based on an optimal linear predictor, are employed to capture certain natural properties of cover images. (72 dimension)
- **Moulin-156D** [19] Features are extracted from both empirical PDF (probability density functions) moments and the normalized absolute characteristic function.

Table 4 The mean absolute difference between the PVD histograms before and after data hiding using different methods with different embedding rates

	Proposed	Ori PVD	IPVD	PVD-LSB	Adaptive-edge
5% bpp	1.2478	2.5998	1.8820	1.3751	1.2316
10% bpp	1.8770	4.9401	3.4280	2.5290	2.2628
20% bpp	2.9329	9.4669	6.3649	4.7485	4.2306
30% bpp	4.1173	13.8332	9.1084	6.8763	6.1125



(a) 30% bpp using Li-1D (b) 30% bpp using Huang-1D

Fig. 7 The ROC curves for five steganographic methods with the same embedding rate of 0.3 using two special feature sets. The x-coordinate and y-coordinate denote the false positive rate (FPR) and true positive rate (TPR), respectively

In our experiments, we follow the extraction scheme proposed in the paper [19] but without a feature selection processing. The highest statistical order is set as $N = 6$, and thus we get 156 dimensions features.

– **Li-110D** [6] Steganalytic features are extracted from the normalized histogram of the local linear transform coefficients [15] of the image. The experimental results in [6] show that these features can capture certain changes of the local textures before and after data embedding, and thus can detect the presence of hidden message effectively, especially for some adaptive steganographic algorithms, such as MBNS [26], MPB [8] and JPEG2000 BPCS [9], even with low embedding rates, for instance, 10% bpp. (110 dimension)

In a pattern recognition system, classifier is another important factor that affects the system performance besides features selection. Usually, non-linear classification schemes will produce higher detection rate. However, it may be too complex and too slow to configure the optimal parameters for those algorithms. For instance, it is empirical to select the kernel functions and cost parameters in Support Vector Machine (SVM), also a gird search is required to get the better parameters for each feature set. For simplicity and without loss of comparison, we use the Fisher Linear Discriminant (FLD) classifier, which is used in some steganalysis algorithms e.g. in [2, 6, 19], for all the feature sets as mentioned above.

For a given embedding rate, the number of cover images used in the experiments is shown in Table 1. We then create the stego images using our proposed method and the other four PVD-based approaches as mentioned previously. For each image database, we randomly choose half of cover images and their corresponding stego images for training, and the remaining images are used for testing. The detection accuracy has been averaged over 20 times by randomly splitting the training and testing sets to achieve more reliable results. Table 5 shows the average accuracy with the embedding rate of 5% bpp, 10% bpp, 20% bpp and 30% bpp, respectively.

It is observed that our proposed algorithm significantly outperforms the previous PVD-based methods in all cases. For instance, the detection accuracies are over 0.80 for all the existing ones when the embedding rate is 20% bpp, while our detection accuracy is just 0.60, much closer to the random guessing 0.50. On average, our

Table 5 The average accuracy of each feature set on FLD in different cases

Embedding capacity	Steganographic algorithms	Shi 78-D	Farid 72-D	Moulin 156-D	Li 110-D	Max. accuracy
5% bpp	Proposed	0.52	0.54	0.52	0.53	0.54*
	Original PVD	0.68	0.69	0.72	0.73	0.73
	IPVD	0.60	0.58	0.60	0.67	0.67
	PVD-LSB	0.67	0.59	0.66	0.71	0.71
	Adaptive-edge	0.64	0.56	0.60	0.67	0.67
10% bpp	Proposed	0.52	0.56	0.54	0.55	0.56*
	Original PVD	0.78	0.79	0.81	0.80	0.81
	IPVD	0.68	0.63	0.68	0.74	0.74
	PVD-LSB	0.79	0.66	0.76	0.79	0.79
	Adaptive-edge	0.73	0.61	0.69	0.75	0.75
20% bpp	Proposed	0.55	0.60	0.57	0.60	0.60*
	Original PVD	0.88	0.87	0.87	0.85	0.88
	IPVD	0.78	0.70	0.77	0.80	0.80
	PVD-LSB	0.88	0.73	0.85	0.85	0.88
	Adaptive-edge	0.82	0.67	0.79	0.82	0.82
30% bpp	Proposed	0.60	0.64	0.63	0.66	0.66*
	Original PVD	0.92	0.90	0.90	0.87	0.92
	IPVD	0.83	0.73	0.81	0.83	0.83
	PVD-LSB	0.91	0.76	0.89	0.88	0.91
	Adaptive-edge	0.86	0.70	0.84	0.85	0.86

For each steganographic algorithm, the underlined values denote the maximum accuracy among the four feature sets. For each embedding rate, the values marked with an asterisk denote the accuracy which is the closest to the random guessing (0.5) among the five steganographic algorithms

detection accuracy will drop around 20% compared with the others, which means that our method has better performance to resist the four universal steganalysis.

Based on our analysis, the main contribution to the higher security performances is that our method can select embedding position adaptively according to the relationship between content edges and the size of the secret message to be embedded. The reason is that, assuming that an image is made up of many non-overlapping small sub-images (embedding units), and then different sub-images usually have different capacity of hiding message. Based on the characteristics of HVS, it is expected that the embedding units located at the sharper regions have better hiding characteristics than those at the smoother/flat regions. By adjusting a threshold T adaptively, the proposed method can firstly use the sharper edge regions for data hiding while preserving the other embedding units. In such a way, the new method can make full use of the sharper edges in a cover image as far as possible. It is well-known that edge information (such as locations and statistical moments etc.) is highly dependent on image contents. However, a good model of natural image contents has not been seen. Therefore, the universal features are difficult in catching those changes that along content edges, especially for those images with more complicated edge regions, such as Baboon. While for most existing PVD-based schemes, the embedding position is mainly determined by a PRNG (Pseudo-Random Number Generator) without considering the relationship between the image contents and the size of the embedded message, which means that each pixel (pixel-pair) in a cover image has the same probability of being selected for data hiding. Although more secret bits can embedded into edge regions, the smoother regions or even flat regions

are inevitably contaminated by such a random selection scheme, which will lead to poorer security performances based on our extensive experiments.

5 Concluding remarks

In accordance with the limitations of the existing PVD-based schemes, an adaptive and more secure steganographic method has been studied in the paper. Quite different from typical PVD and it improved versions, whose embedding positions are mainly determined by a PRNG without considering the relationship between the image contents and the size of the message to be embedded. Our novel method can first embed secret bits at the sharper regions adaptively, while keeping those smooth/flat regions as they are. Besides that, the new method does not destroy sort relationships between the three consecutive pixels within all the embedding units. In such a way, fewer visual artifacts and detectable artifacts will be introduced. The experimental results evaluated on a large image database show the significant improvements in terms of adaptability and security compared with the existing ones.

In the future step, we will investigate whether our proposed idea can be applied to other steganographic schemes that in the frequency domain for enhancing security.

Acknowledgements This work is supported by the NSFC (60633030), 973 Program (2006CB303104), China Postdoctoral Science Foundation (20080440795), Funds of Key Lab of Fujian Province University Network Security and Cryptology (09A011) and Guangzhou Science and Technology Program (2009J1-C541-2).

Appendix

Figure 8 illustrates the four different cases according to the pixel values g_i, g_{i+1}, g_{i+2} within an embedding unit and a given threshold T. Note that g_i and g_{i+2} are fixed before and after data embedding. In this appendix, we want to show you how to determine the range of the centre pixel values $range_{g'_{i+1}}$ so that the sort order of the three consecutive pixel values and the relationships between the pixel values and T are well preserved after data embedding.

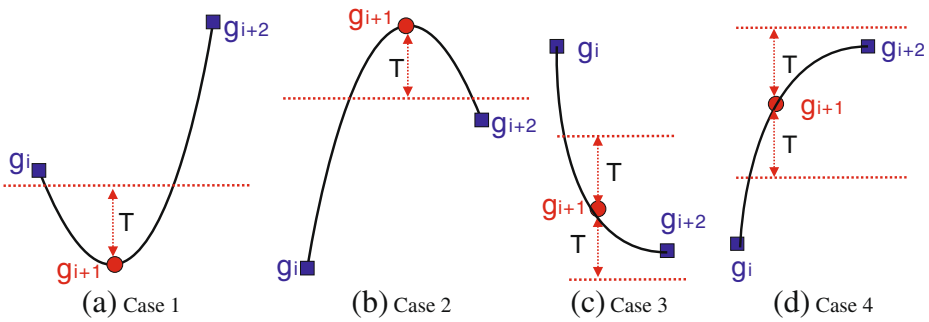


Fig. 8 Four different cases according to the values of g_i, g_{i+1}, g_{i+2}

Without loss of generality, **case 1.1** is considered. Other cases can be analyzed in a similar way. In **case 1.1**,we have the following inequations:

$$g_{i+1} < g_i, \quad g_{i+1} < g_{i+2}, \quad |d_1| > T, \quad |d_2| > T$$

Then we obtain

$$|d_1| = |g_i - g_{i+1}| = g_i - g_{i+1} > T, \quad |d_2| = |g_{i+1} - g_{i+2}| = g_{i+2} - g_{i+1} > T$$

Namely

$$g_{i+1} < g_i - T, \quad g_{i+1} < g_{i+2} - T$$

Since g_{i+1} is an integer and belongs to $[0, \ldots, 255]$, we can obtain

$$g_{i+1} \in \left[0, \ldots, min\left(g_i - T - 1, g_{i+2} - T - 1\right)\right]$$

Therefore, if we limit the centre pixel value changing in the following range when doing data embedding

$$range_{g'_{i+1}} = [0, \ldots, min(g_i - T - 1, g_{i+2} - T - 1)]$$

All the inequations in **case 1.1** will hold. And this is very important to preserve some local structure features in the cover image. However, most existing PVD-based approaches will inevitably destroy such relationships and thus make it easy to detect based on the extensive experiments.

Moreover, it can be seen that the range is constrained by a threshold T. Usually, the larger the threshold T is, the narrower the range is, and thus the fewer secret bits can be embedded using our proposed method. In practice, we can change the threshold T to adjust the embedding capacity. For the short secret message, we can just use those pixels on the sharper edges (larger than T) within an image for data hiding, while keep other smooth regions (smaller than or equal to T) unchanged. If there is not enough space for the given secret message, we will decrease the threshold T to enlarge the embedding space. In such a way, our method is more adaptive with image contents. Based on our experiments and analysis, both the visual imperceptivity and security will become much better.

Please note that since all the inequations are preserved after data embedding, we can reappear the same range in the stego image, which guarantees the validity of data extraction.

References

1. Chan CK, Cheng L-M (2004) Hiding data in images by simple LSB substitution. Pattern Recogn 37:469–474
2. Farid H (2002) Detecting hidden messages using higher-order statistical models. In: IEEE int. conf. on image processing, vol 2, pp II905–II908
3. Huang F, Li B, Huang J (2007) Attack LSB matching steganography by counting alteration rate of the number of neighbourhood gray levels. In: IEEE international conference on image processing, vol 1, pp 401–404
4. Ker AD (2005) Steganalysis of LSB matching in grayscale images. IEEE Signal Process Lett 12(6):441–444
5. Lee YK, Chen LH (2000) High capacity image steganographic model. In: Proceedings of IEE inst. elect. eng., vis. images signal process, vol 147, pp 288–294

6. Li B, Huang J, Shi Y (2008) Textural features based universal steganalysis. In: Proceedings of the SPIE on security, forensics, steganography and watermarking of multimedia, vol 6819, pp 681912
7. Li X, Zeng T, Yang B (2008), Detecting LSB matching by applying calibration technique for difference image. In: Proceedings of the 10th ACM workshop on multimedia and security, Oxford, United Kingdom, pp 133–138
8. Nguyen B, Yoon S, Lee H (2006) Multi bit plane image steganography. In: Proceedings of 5th int. workshop on digital watermarking, pp 61–70
9. Noda H, Spaulding J (2002) Bit-plane decomposition steganography combine with JPEG2000 compression. In: Proceedings of 5th int. workshop on information hiding, pp 295–309
10. NRCS Photo Gallery (2005) Available at: http://photogallery.nrcs.usda.gov/
11. Pereira S, Voloshynovskiy S, Madueno M, Marchand-Maillet S, Pun T (2001) Second generation benchmarking and application oriented evaluation. In: Information hiding workshop III, pp 340–353
12. Provos N (2001) Defending against statistical steganalysis. In: Proceedings of 10th conf. on USENIX security symposium
13. Schaefer G, Stich M (2003) Ucid: an uncompressed color image database. In: Proceedings of SPIE electronic imaging, storage and retrieval methods and applications for multimedia, vol 5307, pp 472–480
14. Shi Y, Xuan G, Zou D, Gao J, Yang C, Zhang Z, Chai P, Chen W, Chen C (2005) Image steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image, and neural network. In: IEEE int. conf. on multimedia and expo
15. Unser M (1986) Local linear transforms for texture measurements. Signal Process 11(1):61–79
16. Voloshynovskiy S, Herrigel A, Baumgaertner N, Pun T (1999) A stochastic approach to content adaptive digital image watermarking. In: Workshop on information hiding, pp 211–236
17. Voloshynovskiy S, Pereira S, Herrigel A, Baumgartner N, Pun T (2000) Generalized watermarking attack based on watermark estimation and perceptual remodulation. In: Proceedings of SPIE, vol 3971, pp 358–370
18. Voloshynovskiy S, Pereira S, Iquise V, Pun T (2001) Attack modelling: towards a second generation watermarking benchmark. Signal Process 81:1177–1214
19. Wang Y, Moulin P (2007) Optimized feature extraction for learning-based image steganalysis. IEEE Trans. on Information Forensics and Security 2(1):31–45
20. Wang RZ, Lin CF, Lin JC (2001) Image hiding by optimal LSB substitution and genetic algorithm. Pattern Recogn 34(3):671–683
21. Westfeld A (2001) F5—a steganographic algorithm high capacity despite better steganalysis. In: Proceedings of 4th int. workshop on information hiding, pp 289–302
22. Wu DC, Tsai WH (2003) A steganographic method for images by pixel-value differencing. Pattern Recogn Lett 24:1613–1626
23. Wu HC, Wu NI, Tsai CS, Hwang MS (2005) Image steganographic scheme based on pixel-value differencing and LSB replacement methods. In: Proceeding of IEE inst. elect. eng., vis. images signal process, vol 152, no 5, pp 611–615
24. Yang CH, Weng CY, Wang SJ, Sun HM (2008) Adaptive data hiding in edge areas of images with spatial LSB domain systems. IEEE Trans. on Information Forensics and Security 3(3):488–497
25. Zhang X, Wang S (2004) Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security. Pattern Recogn Lett 25:331–339
26. Zhang X, Wang S (2005) Steganography using multiple-based notational system and human vison sensitivity. IEEE Signal Process Lett 12(1):66–70

Weiqi Luo received the Ph.D degree from Sun Yat-Sen University, China, in 2008. He is currently a postdoctoral researcher in Guangdong key laboratory of information security technology, Guangzhou, China. Dr. Luo's research interests include digital multimedia forensics, steganography and steganalysis.

Fangjun Huang received the B.S. degree from Nanjing University of Science and Technology, China, in 1995, the M.S. degree and the Ph.D. degree from Huazhong University of Science and Technology, China, in 2002 and 2005, respectively. Now, he is a faculty at School of Information Science and Technology, Sun Yat-Sen University, China. From June of 2008, he has been doing his post-doctoral research in Department of Electrical and Computer Engineering, New Jersey Institute of Technology, USA. His research interests include digital forensics and multimedia security.

Jiwu Huang received the B.S. degree from Xidian University, China, in 1982, the M.S. degree from Tsinghua University, China, in 1987, and Ph.D. degree from Institute of Automation, Chinese Academy of Science in 1998. He is currently a Professor with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China. His current research interests include multimedia security and data hiding. Dr. Huang serves as a member of IEEE CAS Society Technical Committee of Multimedia Systems and Applications and the chair of IEEE CAS Society Guangzhou chapter.