A Progressive Quality Hiding Strategy Based on Equivalence Partitions of Hiding Units

Shaohui Liu, Hongxun Yao, Shengping Zhang, and Wen Gao

VILAB, School of Computer Science and Technology, Harbin Institute of Technology, 150001 Harbin, P.R. China
shliu@hit.edu.cn

Abstract. Many sophisticated schemes are springing up recently with better characteristics, such as higher capacity and better security. However, if we tune the size of the secret message progressively, most methods do not provide the progressive quality characteristic which means that the relationship between the quality of stego image and the size of the secret message could be represented by a smooth curve without any jump points. This paper designs a novel hiding strategy based on an equivalence relation, which not only provides the progressive quality characteristic but also enhances remarkably the quality of stego image without sacrificing the security and capacity compared with original steganography schemes. In the proposed strategy, all hiding units can be partitioned into equivalence classes according to a constructed equivalence relation based on the capacity of hiding units. Following that, the hiding procedure is performed in predefined order in equivalence classes as the traditional steganography scheme. Because of considering the relation between the length of message and capacity, the hiding method using proposed hiding strategy outperforms the original approaches when embedding same message. Experimental results indicate that the proposed strategy gains up to 4.0 dB over existing hiding schemes.

Keywords: Data hiding, Equivalence class partition, Hiding strategy, Progressive Quality hiding.

1 Introduction

Steganography, sometimes also called as data hiding, is the art of how to hide message into another public signal (cover signal) without perceptual distortion. When the public signal is send to the receiver, the hidden message is transmitted secretly and can be extracted by an authorized receiver. Steganography is a branch of information hiding like watermarking; hence it shares some common foundations with watermarking. For example, they have the similar components. But it should be noted that robust watermarking schemes focus heavily on the robustness of algorithms and fragile watermarking schemes focus heavily on the fragility of watermarks. Whereas, the main requirement of steganography is undetectability[1]. According to the definitions in [1], watermarking is defined as the practice of imperceptibly altering a cover signal to embed a message about that cover signal. And steganography is

Y.Q. Shi (Ed.): Transactions on DHMS VI, LNCS 6730, pp. 58–82, 2011.

defined as the practice of undetectably altering a cover signal to embed a secret message. Generally speaking, undetectability may be more challengeable than imperceptibility. Hence, undetectability first requires imperceptibility for some aspects. Moreover, from the form of transmitting secret message, steganography shares some similar features with cryptography. However, there also exist some differences between steganography and cryptography. The most essential difference is that steganography not only can hide secret message (maybe cipher text) but also can hide the existence of secret communication. Thus only authorized receiver can know which media may have the transmitted message and then extract the message, and other people do not know the existence of communication. Hence, for security, steganography seems better than cryptography. In steganograhpy, host signals are limited to be all kinds of multimedia, such as image, video and audio, and recently other signals, such as document, executable program, TCP/IP packet, NTFS (New Technology File System which is the standard file system of Windows NT) and etc also can be used as host signals. But cryptography can be used in all signals. Steganography and data hiding are basically equivalent. The small difference is that, steganography requires the security, especially the capability of resisting steganalysis, but data hiding does not. Nevertheless, the data hiding method will become more competitive if it can resist the common steganalysis. In this paper, these two terms are used equivalently except specified clearly.

Table 1. The taxonomy of data hiding

Criteria for classification	classes	Explanation and typical algorithms
1. Whether the algorithm is adaptive to the message size	Message-size-independent hiding	Once the algorithm is selected, then the hiding procedure is fixed, such as [2,6,7]
	Message-size-dependent hiding	According to the message size, one can calculate the appropriate parameters and choose the optimum hiding strategy[3]
2. Whether the algorithm is invertible	Invertible hiding	The host image can be exactly recovered from the stego-image after extracting the hidden data[4]
	Non-invertible hiding	The host image does not recovered from the stego image[2,3,5-15]
3. Whether the algorithm considers the HVS properties	HVS-based hiding	The hiding algorithm accords with HVS[9-11]
	Non-HVS-based hiding	The hiding algorithm does not accords with HVS[2-8]
4. Whether the hiding unit is separable	hiding with separable hiding units	The host image can be grouped into different hiding units where they do not overlap each other[9,10,13,14,18]
	hiding with non-separable hiding units	The host image do not group into different hiding units because there exists some overlapped area in some hiding units[8,11]
5. Whether the hiding capacity of hiding units is fixed	hiding with fixed capacity hiding unit	The host image has fixed hiding capacity[2,9,10]
	hiding with variable capacity hiding unit	The different area of the host image may have different hiding capacity[3,4,6,7]

Due to having the similar function with cryptography and hiding the fact of communication, steganography has broad applications in modern digital signal processing and security communications. In the most recent years, many steganography methods are invented [2-15]. We can classify them into different classes according to different criteria. For example, in Table1, we give five criteria and some related typical hiding algorithms. Note that some criteria may overlap with each other, such as the third and fifth criteria. According to the fifth criteria, those algorithms can be mainly grouped into two classes. The first class is to hide constant bits for each hiding unit like the LSB(Least-Significant-Bit) hiding algorithm[2,5,6,7] which is based on manipulating the LSB planes by directly replacing the LSBs of the host-image with the message bits. The hiding based predictive coding [8] also belongs to the first class. From the hiding processing, we can deduce that the first class hiding does not consider any visual characteristics of host images. However, according to the characteristics of the human visual perception capability we know that areas with different local characteristics can tolerate different amounts of changes. Thus, the low activity areas should be hidden with the less secret message than the high activity areas. Moreover, the first class hiding algorithms mentioned above provide potentially the stegnalysis the chance to use the changes of low activity areas to detect the existence of steganography. The second class of hiding algorithm considers the HVS (Human Visual System), namely the local characteristics of images. In fact, researches have noticed that HVS is very crucial to design steganography system [9-14]. This type of algorithms hides different number of bits in hiding units with different local characteristics. It is obvious that the second type algorithm is superior to the first algorithm. For example, Liu et al.[15] proposed a variable LSB hiding scheme based on Minimizing the Distortion in Equivalence Class (is abbreviated to MDEC), where pixels are grouped into different equivalence class according to their luminance value, and then LSB hiding and an optimal adjustment process are used to hide the message bits. Although it only considers the characteristics of one single pixel, it still exhibits evident superiority to the first class of algorithm. However, the second class of algorithm also has disadvantages. Because the second type algorithm has considered the HVS, most of them hide information bits into the hiding unit by a sequential hiding way (otherwise, the HVS is hard to be incorporated into the hiding system). Hence, even though the length of the message bits is much less than the capacity, the hiding scheme still leads to a perceptual distortion in the embedded area. Moreover, the distortion and some other statistical properties of stego-images may be used by steganalysis to crack the steganography[16,17]. In fact, many existing hiding methods are closely related to the size of message. Taking Direct Simple LSB hiding algorithm in Fig1 as an example, suppose the host signal be an image, where one must determine first using how many LSB planes to hide data, and in extracting side, one should know the number of LSB planes used in hiding side. Moreover, when the bits of the message to be embedded is larger than the bits of 2 LSB planes but less than the bits of 3LSB planes, Direct Simple LSB hiding must use 3 LSB planes of host-image to hide the message. It is obvious that distortion induced by 3 LSB planes hiding is definitely larger than that caused by 2 LSB planes hiding. Because the number of the used LSB planes equals to ceil(the number of the bits of message/the number of the bits of one LSB planes) in Direct Simple Hiding algorithm, when the bits of secret message increases such that the required LSB planes changes from 2 to

3, all pixels in stego images can be classified into two classes, some pixels carry 3 message bits, and other pixels do not carry any message bits, there exists some perceptual-quality jump as in Fig.1. Whether can we remedy this phenomenon? In fact, we find that the Variable LSB hiding is a valid measure to deal with the quality-jump phenomenon of Direct Simple LSB Hiding algorithm from Fig.1. And the VariableLSB+MDEC hiding can further improve the performance of Variable LSB hiding. However, it is only valid for LSB hiding. In general cases, how to design an optimal progressive quality hiding algorithm to remedy this phenomenon? Another criterion for taxonomy is the fourth rule where hiding algorithms are classified into two classes according to the separability of hiding units. Generally speaking, under different classification criteria, there exist different measures to improve the performance. For data hiding with fixed capacity hiding unit, one can use many measures, such as, OPAP(Optimal Pixel Adjustment Process)[6], dynamic programming[7], MDEC[15] and so on to enhance the performance. However, for those data hiding with variable capacity hiding unit, whether do any appropriate measures improve the performance of them?

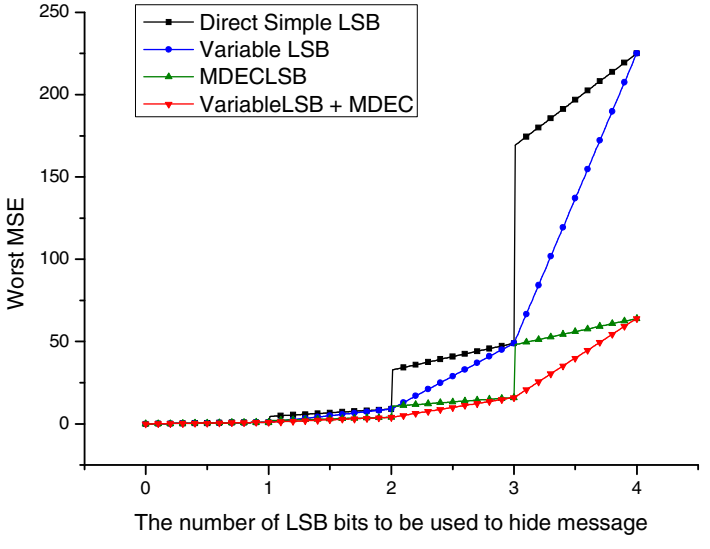


Fig. 1. The quality-jump phenomenon of LSB hiding in [15], where Direct Simple LSB means that the message is embedded into host image by LSB plane substitution where #the used LSB planes equals to ceil(the number of the bits of message/the number of the bits of one LSB planes), hence it is possible that some pixels do not carry any message bits; Variable LSB means that the message is embedded into host image LSB bitplanes by LSB bitplanes until all message bits have been embedded into host image, hence each pixel should carry some message bits except that #the used LSB planes is one; MDEC LSB means MDEC hiding strategy is used to adjust the pixel value of stego image of Direct Simple LSB; VariableLSB + MDEC means that MDEC hiding strategy is used to adjust the pixel value of stego image of Variable LSB

In this work we mainly consider the aforementioned two problems: the first is how to remedy the quality-jump phenomenon; and the second is how to improve the performance of data hiding with variable capacity hiding units. Via analyzing these two questions creatively, we find that there is a novel way to deal with two questions, where it not only decreases distortion but also eliminates the quality-jump phenomenon. The proposed hiding strategy is based on equivalence partitions of hiding units; it can significantly improve the quality of stego image without sacrificing the security and the hiding capacity. When embedding the same message bits, the performance of our novel strategy is superior to the classical methods. And more important, the proposed method can hide secret data in a progressive way until all secret data have been hidden completely or the host image's hiding capacity has been exhausted. Consequently, it is a progressive quality hiding method. This is a favorable feature in practical applications because users do not need to change the hiding method or select host signal or tune some parameters to carry the secret message like the manipulation of determining how many LSB planes should be used to carry secret message in aforementioned LSB hiding.

The rest of this paper is organized as follows. Section 2 presents the general model of proposed hiding strategy. In section3, a specific example about how to apply the hiding strategy proposed in section2 is conducted. And the experimental results are presented in section 4. In section 5, we give a short discussion about steganalysis. And this paper is concluded in section 6.

2 The Proposed Hiding Strategy

From the introduction, we know that the quality-jump phenomenon exists widely, and the distortion induced by hiding may be enough large to be used in steganalysis even the size of secret message is far less than the capacity of host image. In this section, we propose an adaptive hiding strategy which can be used in all hiding schemes with hiding unit partition mechanism (where all hiding units in host signal are independent of each other) to improve their performance. The hiding procedure is shown in Fig.2. Following, we give the general model of proposed hiding strategy.

No loss of generality, suppose the length of message to be hidden into a hiding unit in host media belongs to the range $R = [r_l, r_u]$, where $r_l, r_u \in Z$ and $r_l > 0$. And the embedding/extracting procedure is denoted by function of $EmbeddingFun() / ExtractingFun()$. The output of $EmbeddingFun()$ is the stego-unit (denoted by $StegUnit_i$) with an input unit (denoted by $InputUnit_i$) of hidden message. The output of $ExtractingFun()$ is the extracted message after extracting procedure is performed over a stego-unit. Then, a general hiding algorithm can be described as:

$$StegUnit_i = EmbeddingFun(InputUnit_i, message_i, key), i = 1, \cdots, n \quad (1)$$

Where the three parameters of $InputUnit_i, message_i$ and key indicate the hiding unit, message bits and key separately, and i is the index of the unit. $StegUnit_i$ is

the output after $message_i$ is hidden into $InputUnit_i$ by $EmbeddingFun()$. The total hiding procedure is finished until all the data have been hidden or all the hiding units have been used. And the corresponding extraction procedure is:

$$message_i = ExtractingFun(StegUnit_i, key), i = 1, \cdots, n \tag{2}$$

Given a set H in which each element is a hiding unit of cover signal, then the general hiding strategy can be divided into three steps.

1. Constructing an equivalence relation ~on H, generally speaking, the equivalence relation ~ can be the capacity of a hiding unit;

2. Establishing the set of equivalence classes based on the equivalence relation ~. And the set of all equivalence classes of H forms a partition of H;

3. Hiding message bits equivalence-class by equivalence-class. In each equivalence class, user can permute the embedding order of hiding units to improve the security of the steganography scheme.

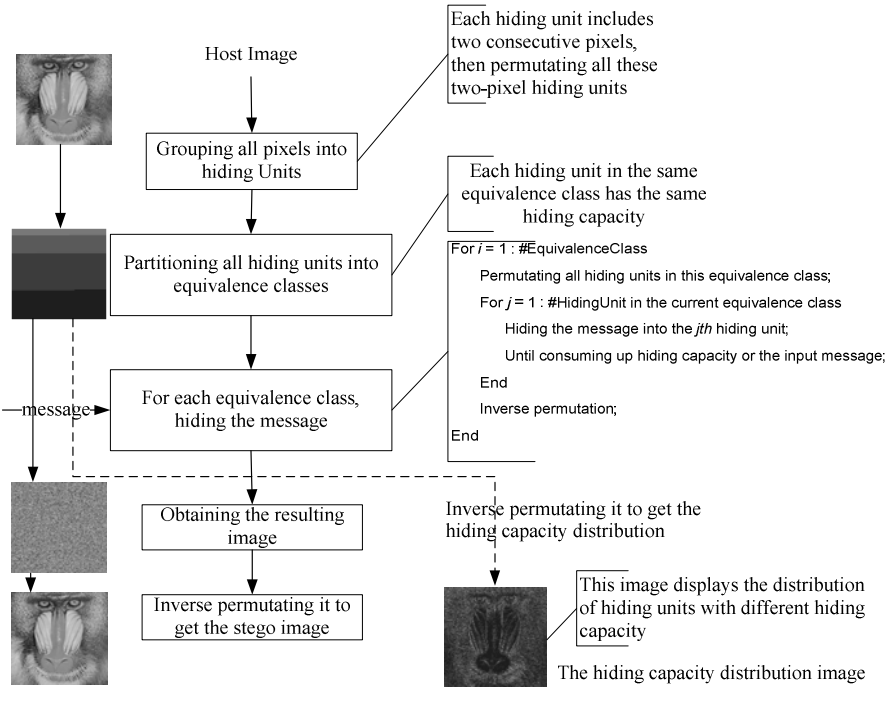


Fig. 2. The hiding procedure. It is noted that the hiding capacity distribution image is obtained by inverse permutating the grouped hiding capacity image, and these two images are obtained by 60× *CapacityofHidingUnit* for better visual presentation, for example, suppose the capacity of one hiding unit is 2bits, then the corresponding gray value of hiding capacity distribution image unit is 60× 2=120(more details in Fig. 4). Hence, in the hiding capacity distribution image, the darker the color is, the less the hiding capacity is.

In this paper, the capacity of one hiding unit which indicates the length of binary message that can be hidden into the hiding unit is used as an equivalence relation. Then, based on this equivalence relation, the equivalence class is established. In this paper, all hiding units are partitioned into $r_u - r_l + 1$ equivalence classes according to equation (3). Under this setting, It is obvious that hiding units in $EquiClass_0$ have the least capacity.

$$\begin{aligned} EquiClass_i = \{ & InputUnit_m \mid Length(ExtractingFun(\\ & EmbeddingFun(InputUnit_m, message_m, key), key)) \\ & \equiv i+1 (\mathrm{mod}\ r_u - r_l + 1), m = 1, \cdots, n\} \end{aligned} \tag{3}$$

Where $Length(message)$ denotes the length of message.

Then hiding procedure can be executed equivalence-class by equivalence-class, namely firstly hiding message bits in all hiding units in the equivalent class $EquiClass_0$, and then hiding message bits in all hiding units in $EquiClass_1$, $EquiClass_2$ and so on. The specific hiding procedure can be written as equation (4).

$$StegUnit_{i,j} = EmbeddingFun(InputUnit_{i,j}, message_{i,j}, key) \tag{4}$$

where $InputUnit_{i,j} \in EquiClass_i$, and $StegUnit_{i,j}$ represents the j^{th} hiding unit in the i^{th} equivalence class, and $message_{i,j}$ represents the hidden message in $InputUnit_{i,j}$.

3 An Example of Application about the Hiding Strategy

We know adaptive hiding algorithms take advantage of texture characteristics of images, hence this kind of hiding algorithms have better performance than hiding algorithms based on LSB. Difference between neighboring elements in an image is just one of texture characteristics, similarly, differences between different entities, for example, pixel values, histogram bin values, transform coefficients energy and so on, were extensively used to hide message in information hiding community. Among these differences, the pixel-difference is very simple and commonly used to design hiding algorithms [9,10,12,13]. In this paper, we select the pixel-difference-based hiding methods [9,10] as the target hiding algorithms, and then apply the proposed hiding strategy to them to illustrate the hiding strategy proposed in section 2. In fact, the proposed hiding strategy can be used to all hiding algorithms whose hiding units

are independent of each other, such as [14]. Following, we give the details about how to apply the strategy. The hiding procedure is shown in Fig.2.

Suppose two adjacent pixels p_i and p_{i+1} in one gray image, theirs gray value are g_i and g_{i+1} respectively, and the difference $d = g_{i+1} - g_i$. Obviously, if the range of gray value in an image is [0,255], then d belongs to [-255,255]. And then this interval is divided into sub-intervals R_k, which satisfies following conditions: $R_k = [l_k, u_k], k = 1, \cdots, n$, where $l_1 = 0$, $u_k = 255$, and the width of each sub-interval is the integer power of 2. Then the absolute difference $|d|$ is quantized into some sub-interval. For the sake of convenience, suppose $|d|$ falls into the k^{th} sub-interval $R_k = [l_k, u_k]$, then defining the length of information bits to be hidden in this difference is $\log_2(u_k - l_k + 1)$ bit. In this paper, we firstly partition all hiding units into equivalence classes as equation (3). And then we embed the secret message as following procedure (which is the same as Wu's work [9]) in a way of equivalence-class by equivalence-class. Suppose the decimal value of $\log_2(u_k - l_k + 1)$ is b, then we can obtain the new difference value as:

$$d' = \begin{cases} l_k + b, & d \geq 0 \\ -(l_k + b), & d < 0 \end{cases} \tag{5}$$

And the resulting pixel values g_i' and g_{i+1}' after hiding information can be calculated by:

$$(g_i', g_{i+1}') = \begin{cases} (g_i - \lceil m/2 \rceil, g_{i+1} + \lfloor m/2 \rfloor) & d\ is\ odd \\ (g_i - \lfloor m/2 \rfloor, g_{i+1} + \lceil m/2 \rceil) & d\ is\ even \end{cases} \tag{6}$$

where $m = d' - d$. When extracting information, firstly processing those hiding units in the first equivalent class $EquiClass_0$, we recalculate the difference $d' = g_{i+1}' - g_i'$. If $|d'| \in R_k$, then the hidden information can be extracted by:

$$b = \begin{cases} d' - l_k, d' \geq 0 \\ -d' - l_k, d' < 0 \end{cases} \tag{7}$$

As the same with hiding procedure, the decimal value b can be expanded into binary string with length of $\log_2(u_k - l_k + 1)$.

Meanwhile, we also apply the proposed strategy to the algorithm of Zhang [10], where the different place with Wu's algorithm is the partition of sub-interval is not fixed. Zhang dynamically partitions the interval [0, 255] into sub-intervals by a predefined random parameter $\beta \in [0,1]$. Suppose the original sub-interval is R_k, then the new dynamically generated sub-interval R_k' is obtained by:

$$R_k' = (l_k', u_k') = (l_k + \lfloor \beta w_k \rfloor, u_k + \lfloor \beta w_{k+1} \rfloor) \tag{8}$$

where $l_0' = 0, u_n' = 255$, $w_k = u_k - l_k + 1$ indicates the width of the k^{th} sub-interval. Note that the new resulting difference is calculated as:

$$d' = \begin{cases} \underset{e \in R_k', \mathrm{mod}(e, w_k) = b}{\arg\min} (|e - d|) & d > 0 \\ -\left(\underset{e \in R_k', \mathrm{mod}(e, w_k) = -b}{\arg\min} (|e - d|) \right) & d < 0 \end{cases} \tag{9}$$

where b is the same as Wu's method. If $0 \le |d| \le u_0'$, then

$$d' = \underset{-u_0' \le e \le u_0', \mathrm{mod}(e, w_0) = b}{\arg\min} (|e - d|) \tag{10}$$

In extracting side,

$$b = \begin{cases} \mathrm{mod}(d', w_0) & 0 \le |d'| \le u_0' \\ \mathrm{mod}(d', w_k) & l_k' \le |d'| \le u_k' \end{cases} \tag{11}$$

It is noted that our proposed method allows users to choose different hiding schemes for different equivalent class hiding units for improving the hiding performance. For example, for the equivalent class with 2 bits hiding capacity, then one can use the method [5] to hide 2 bits in one hiding units with two pixels. In addition, we can also use the permutation transform to increase the security of hiding schemes in each equivalent class without sacrificing the other performance, for example the capability of resisting steganalysis. The (g), (h) and (i) in Fig. 5 in the next section show the fact, where the hiding position is scattered into the whole image, however, the existing hiding schemes do not exhibit this property as (a)-(f)in Fig.5. In the following, we will give the experiments.

4 Experiments

To establish a more quantitative measure of proposed algorithm's performance, we use the peak signal-to-noise ratio (PSNR) and root-mean-square error(RMSE) metrics. Although these measures are generally not very accurate and do not take the features of HVS into consideration, they serve as most commonly used and simple rules of thumb of the quality of stego images. They are defined as:

$$PSNR = 10\log\frac{255^2}{RMSE^2} \tag{12}$$

$$RMSE = \left(\frac{1}{N_{total}}\sum_{i=1}^{N_{total}}(HostImage_i - StegoImage_i)^2\right)^{\frac{1}{2}} \tag{13}$$

Where N_{total} denotes the number of pixels in host image $HostImage$, $HostImage_i$ denotes the pixel value of the i^{th} pixel, $StegoImage_i$ denotes the stego pixel value of the i^{th} pixel of stego image $StegoImage$. At the same time, we also use the structural similarity (SSIM) index [18] to evaluate the effect of HVS. The larger the SSIM index value is, the better the quality of stego image is. In experiments, the Hiding rate of capacity is defined as:

$$Hiding\ rate = \frac{\#\,Bits\,of\ Hidden\,Message}{\#\,Hiding\ Capacity}\cdot 100\% \tag{14}$$

To verify the performance of the proposed hiding strategy, the extensive experiments were carried out. Some selected images are shown in Fig3. The host-images used in our scheme are 8-bit single channel gray-scale images with 256*256 pixels, the selected existing steganography methods include the Wu's algorithm (PVD)[9] and Zhang's algorithm (ImprovedPVD)[10]. At the same time, the proposed strategy is applied to PVD and ImprovedPVD, which are respectively denoted by Adaptive+PVD and Adaptive+ ImprovedPVD. Furthermore, the results are compared with PVD[9] and ImprovedPVD[10] for confirmation. For avoiding the effect of different specific messages, in these experiments, the randomly generated messages are used to test the performance. To make the comparison simple, in all algorithms, the range [0, 255] is divided into two types of sub-intervals. The first type includes six sub-intervals and the widths of all the sub-interval are 8,8,16,32,64 and 128. The second type includes 13 sub-intervals and the widths of these sub-intervals are 2,2,4,4,4,8,8,16,16,32,32, 64 and 64. Following, we will discuss the performance of proposed hiding strategy from capacity, stego image quality, and perceptual distortion.

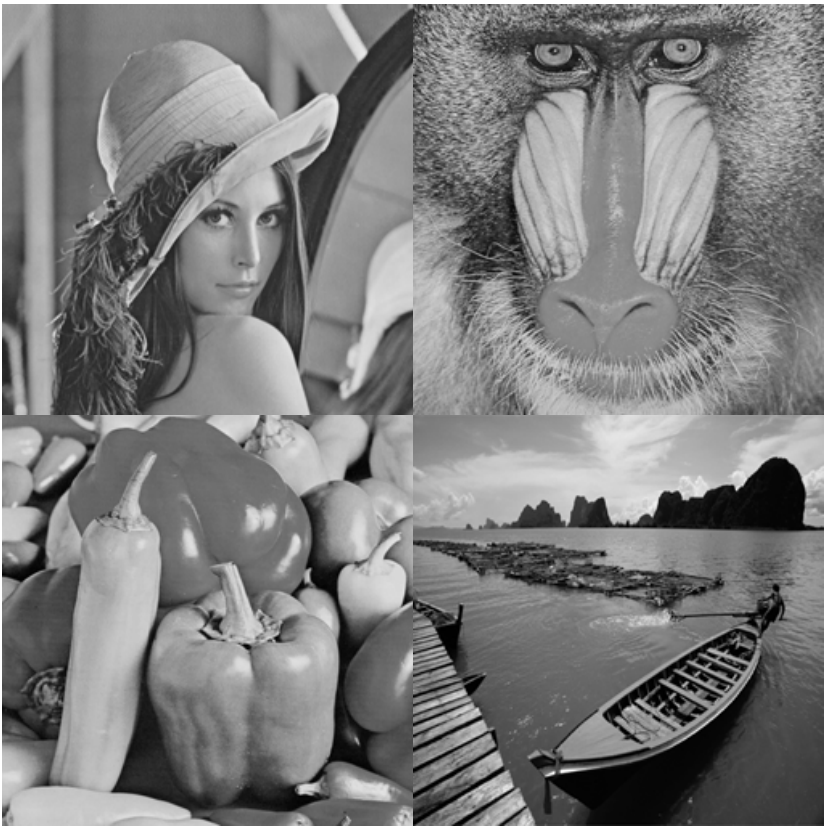


Fig. 3. The host images: from left to right and top to down, Lena, Baboon, Peppers and Boat images

4.1 Capacity

According to the introduction in previous sections, we know that the proposed hiding strategy is an adaptive strategy which can be used in all hiding schemes with hiding unit partition mechanism (where all hiding units in host signal are independent of each other). Moreover, it is just used as an auxiliary measure to improve the performance of existing hiding schemes; hence, capacities of all hiding schemes improved by using this strategy are still very similar with capacities of existing schemes. Table.2 shows the experiments result, where those values emphasized by the bold font are capacities of those improved hiding schemes by using the proposed hiding strategy. It should be noted that the capacity of AdaptivePVD is the same as the capacity of PVD because we only adjust the hiding order. However, the capacity of Adaptive+ImprovedPVD is different from ImprovedPVD because we take the random parameter β to improve the security.

Table 2. The capacity of different hiding method

Image	Hiding method	Capacity when 6 sub-intervals (byte)	Capacity when13 sub-intervals(byte)
Lena	PVD	13019	6872
	AdaptivePVD	**13019**	**6872**
	ImprovedPVD	12698	6624
	Adaptive+ImprovedPVD	**12705**	**6617**
Baboon	PVD	14108	9095
	AdaptivePVD	**14108**	**9095**
	ImprovedPVD	13395	8630
	Adaptive+ImprovedPVD	**13392**	**8621**
Boat	PVD	12101	6009
	AdaptivePVD	**12101**	**6009**
	ImprovedPVD	11699	5813
	Adaptive+ImprovedPVD	**11689**	**5812**
Peppers	PVD	12637	6368
	AdaptivePVD	**12637**	**6368**
	ImprovedPVD	12414	6164
	Adaptive+ImprovedPVD	**12411**	**6156**

4.2 The Quality of Stego Images

One of the most important factors of hiding schemes is the quality of stego images. The experimental results of the proposed hiding strategy are shown in Table3 and Table4, where those values with the bold font are the improved results. From these two tables, we can find that our hiding strategy improves greatly the quality of existing hiding schemes. Although the PSNR does not reflect accurately the HVS, higher PNSR values still show the merit of proposed strategy. In the following sub-section, we will discuss the effect on HVS.

Table 3. The PSNR values with different hiding rate when using 6 sub-intervals

Image	Hiding rate	PVD	AdaptivePVD	ImprovedPVD	Adaptive+ ImprovedPVD
Lena	20%	49.04	**50.48**	51.85	**52.65**
	50%	43.87	**46.52**	46.67	**48.57**
	80%	40.97	**44.52**	44.02	**46.55**
	100%	40.01	**39.05**	42.93	**42.84**
Baboon	20%	43.11	**50.27**	46.42	**51.80**
	50%	40.39	**46.28**	43.52	**47.84**
	80%	38.69	**42.21**	41.90	**44.76**
	100%	37.66	**37.15**	40.83	**40.78**
Boat	20%	49.75	**50.12**	53.15	**52.95**

Table 3. (*continued*)

	50%	44.61	**46.50**	47.85	**49.02**
	80%	41.66	**44.33**	45.01	**46.99**
	100%	40.67	**40.15**	43.96	**43.80**
Peppers	20%	48.12	**50.59**	50.68	**52.82**
	50%	43.46	**46.68**	46.60	**48.81**
	80%	41.65	**44.69**	44.67	**46.74**
	100%	40.93	**40.44**	43.87	**43.73**

Table 4. The PSNR values with different hiding rate when using 13 sub-intervals

Image	Hiding rate	PVD	AdaptivePVD	ImprovedPVD	Adaptive+ ImprovedPVD
Lena	20%	55.36	**58.86**	56.71	**59.04**
	50%	50.03	**54.14**	51.03	**54.62**
	80%	47.36	**50.88**	48.65	**51.32**
	100%	46.36	**46.18**	47.93	**47.82**
Baboon	20%	49.50	**56.83**	51.10	**57.45**
	50%	46.59	**52.10**	48.14	**52.77**
	80%	44.91	**49.07**	46.45	**48.81**
	100%	43.88	**43.81**	45.65	**45.50**
Boat	20%	56.76	**59.51**	57.84	**59.62**
	50%	51.56	**55.09**	52.69	**55.50**
	80%	48.40	**53.18**	49.57	**52.45**
	100%	47.31	**47.25**	48.73	**48.88**
Peppers	20%	54.21	**59.20**	55.69	**59.38**
	50%	50.09	**54.65**	51.14	**55.13**
	80%	48.32	**51.95**	49.38	**52.26**
	100%	47.52	**47.22**	48.81	**48.93**

4.3 The Perceptual Quality of Stego Images

From the above sub-section 4.2, we have found that the proposed strategy has improved greatly the quality of stego images from the PSNR value. In this section, the perceptual quality will be discussed. Experiments have indicated that the perceptual quality is also improved, in fact, the proposed strategy has similar characteristic with HVS-based hiding schemes. Four images in Fig.4 are the capacity distribution images of four images in Fig.2. In Fig.4, the larger gray value means the higher hiding capacity. The left column images are the capacity distribution images reordered by proposed strategy. The right column images indicate that the hiding scheme exhibits some HVS characteristics. For the left column images, the larger the dark area in an image is, the less the hiding capacity of the image is. It further indicates that smooth images have lower hiding capacity (like Lena, Peppers and Boat images), and texture

image has higher hiding capacity (like Baboon image). According to the hiding steps, first the secret message will be hidden into the darkest region with least hiding capacity. Hence, at the beginning of hiding, our method will use more hiding units to hide message than original hiding method, however, the distortions of these hiding units are far less than the just noticeable distortion (JND) value. With the increasing the size of secret message, those hiding units with higher hiding capacity will be used to hide message, and the distortions will become larger. Finally, the distortion of proposed strategy becomes comparable with that of original method. In fact, we found that most of distortions are less than the JND values. Namely, the effect on HVS is also good. The Table 5 shows the SSIM index values. From this Table, the difference between hiding with proposed strategy and hiding without proposed strategy is neglectable. It also means that the proposed hiding strategy does not induce any negative effect on the HVS property of stego images. Furthermore, in Fig.5, (a)~(c) are the difference images between stego images and host images with different hiding rate 50%, 80% and 100% by PVD, (d)~(f) are the difference images between stego images and host images with different hiding rate 50%, 80% and 100%generated by Improved PVD[9], (g)~(i) are generated by Adaptive+ ImprovedPVD. It is noted that the difference images are obtained by $30 \times$ abs (hostimage - stegoimage) for better visual presentation. From this Figure, we can find that compared with PVD and Improved PVD, the proposed strategy can improve the visual quality of stego images remarkably.

Table 5. The SSIM index values with different hiding rate when using 6 sub-intervals

Image	Hiding rate	PVD	AdaptivePVD	ImprovedPVD	Adaptive+ ImprovedPVD
Lena	20%	0.9944	**0.9943**	0.9969	**0.9975**
	50%	0.9878	**0.9880**	0.9932	**0.9938**
	80%	0.9820	**0.9822**	0.9900	**0.9902**
	100%	0.9785	**0.9778**	0.9878	**0.9878**
Baboon	20%	0.9980	**0.9983**	0.9991	**0.9986**
	50%	0.9935	**0.9939**	0.9965	**0.9967**
	80%	0.9884	**0.9905**	0.9937	**0.9945**
	100%	0.9850	**0.9844**	0.9919	**0.9919**
Boat	20%	0.9938	**0.9938**	0.9972	**0.9978**
	50%	0.9894	**0.9892**	0.9948	**0.9945**
	80%	0.9840	**0.9805**	0.9918	**0.9911**
	100%	0.9782	**0.9782**	0.9894	**0.9895**
Peppers	20%	0.9958	**0.9954**	0.9977	**0.9977**
	50%	0.9899	**0.9895**	0.9945	**0.9943**
	80%	0.9837	**0.9835**	0.9907	**0.9907**
	100%	0.9802	**0.9799**	0.9888	**0.9886**

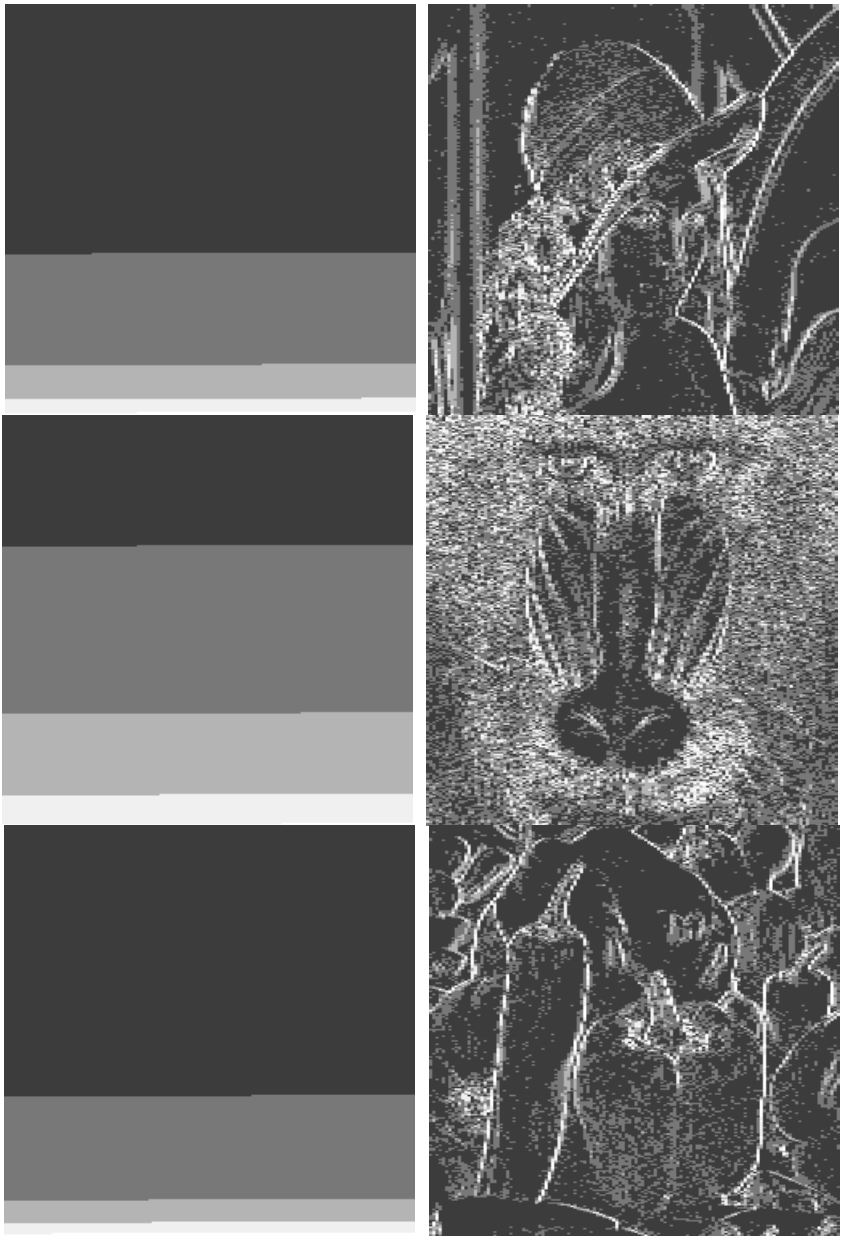


Fig. 4. The hiding capacity distribution images which are obtained by replacing the original pixel values with the hiding capacity of each hiding unit. The left column images are the reordered hiding capacity distribution images, and the right column images are obtained by inverse permuating the left column images. It is noted that these images are obtained by $60 \times$ *CapacityofHidingUnit* for better visual presentation.

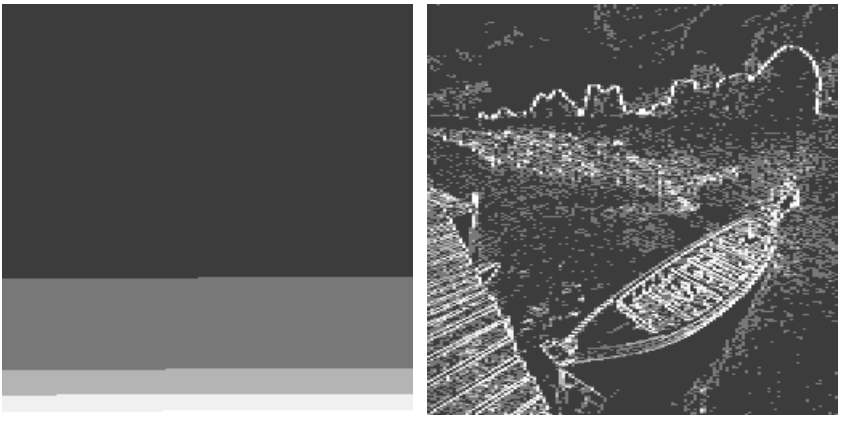
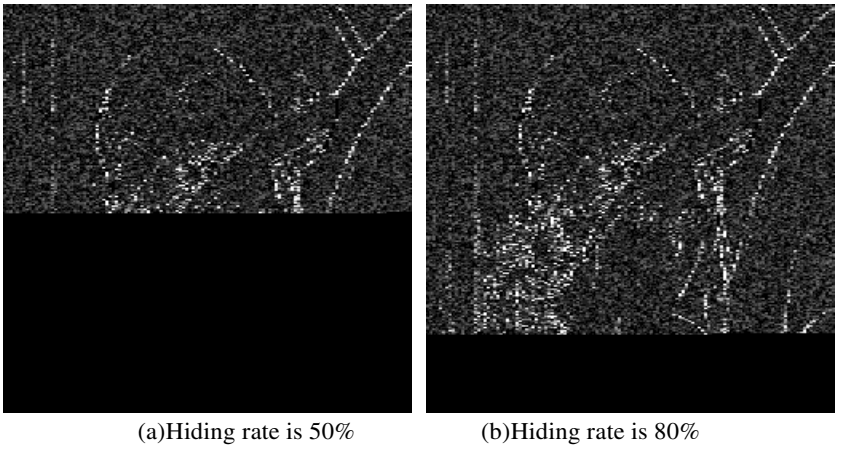
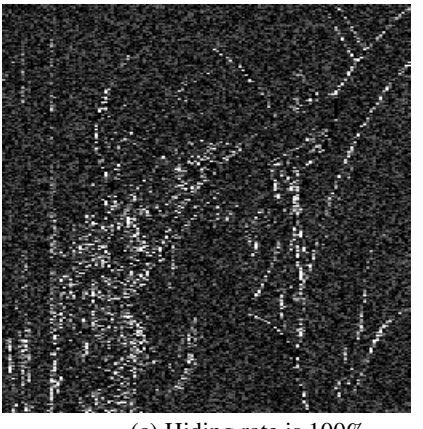


Fig. 4. (*continued*)

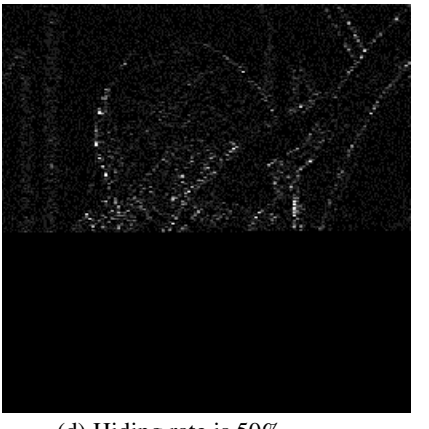


(a)Hiding rate is 50% (b)Hiding rate is 80%

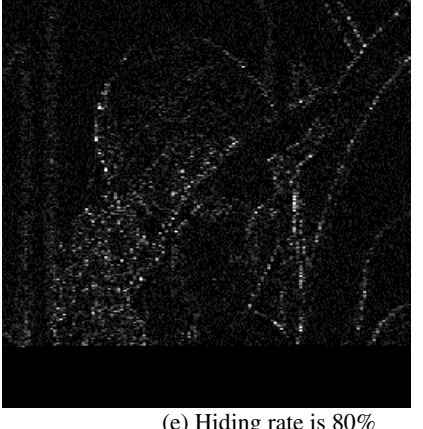
Fig. 5. Difference images between stego images and host images with different hiding rate and methods: (a)~(c) The hiding algorithm is Wu's PVD[9]with different hiding rate, (d)~(f) The hiding algorithm is Zhang's improved PVD[10]with different hiding rate, (g)~(i) The hiding algorithm is Zhang's improved PVD integrated with the proposed hiding strategy. It is noted that the difference images are obtained by 30×*abs (hostimage - stegoimage)* for better visual presentation.



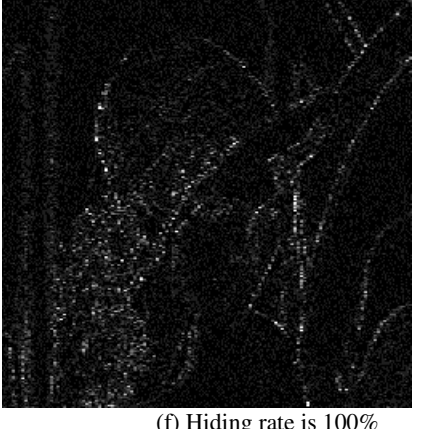
(c) Hiding rate is 100%



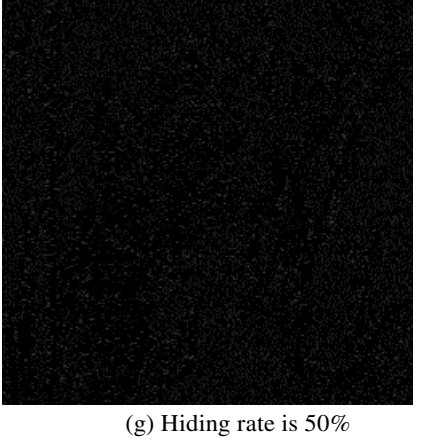
(d) Hiding rate is 50%



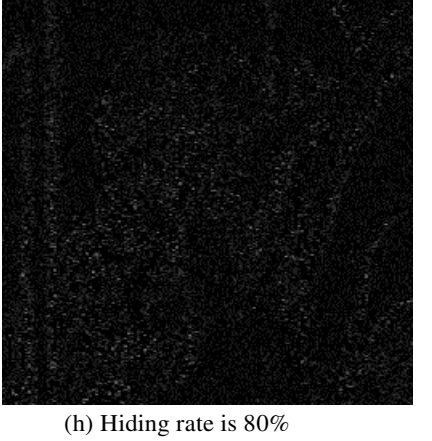
(e) Hiding rate is 80%



(f) Hiding rate is 100%



(g) Hiding rate is 50%



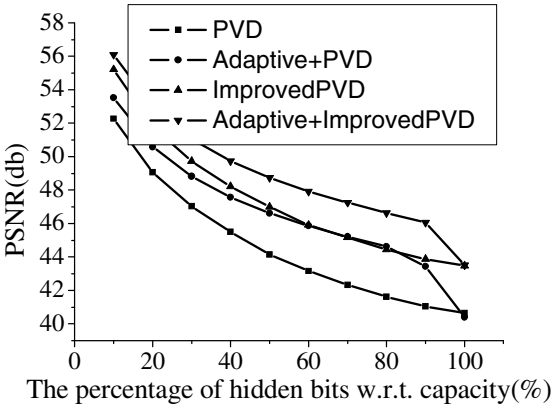
(h) Hiding rate is 80%

(i) Hiding rate is 100%

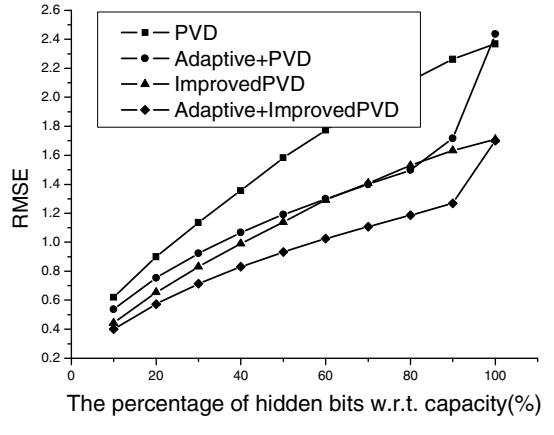
Fig. 5. (*continued*)

The more detailed results are shown in Fig. 6 and Fig7. where message is hidden into the Lena image with different hiding rate. In Fig.6, the number of the sub-intervals is six, and the number of the sub-intervals is thirteen in Fig.7.

From these Figures, it can be observed that the proposed strategy outperforms PVD and ImprovedPVD, without changing the capacity significantly. In Fig.6, if the hiding rate is less than 90%, the gain obtained from the proposed hiding strategy is about from 1.246db to 2.997db in PVD, and from 0.879db to 2.197db in ImprovedPVD. Here, the total average gains are 1.9859db and 1.5157db in PVD and ImprovedPVD respectively. In Fig.7, if the hiding rate is less than 90%, then the gain obtained from the proposed hiding strategy is about from 2.484 to 4.065db in PVD, and from 1.342db to 3.084db in ImprovedPVD. Here, the total average gains are 3.2608db and 2.2769db in PVD and ImprovedPVD respectively. From (c) and (d) of Fig.6 and Fig.7, we observe that the number of overblocks is comparable in all methods. However, when the size of the hidden bits is much less than the capacity, the number of the overblocks of proposed scheme decreases remarkably. We found that the overflow phenomenon occurs only when the number of hidden bits approximating the capacity in our proposed algorithm. But, the existing schemes have overblocks from the beginning of hiding. This indicates that most of overblocks occur on those areas with higher capacity hiding units. Moreover, to some extent, the enhancement of quality of stego-image indicates that the breaking probability by steganalysis also decreases.

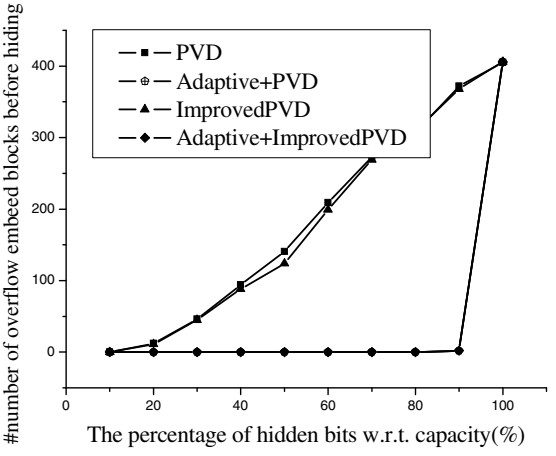


(a)

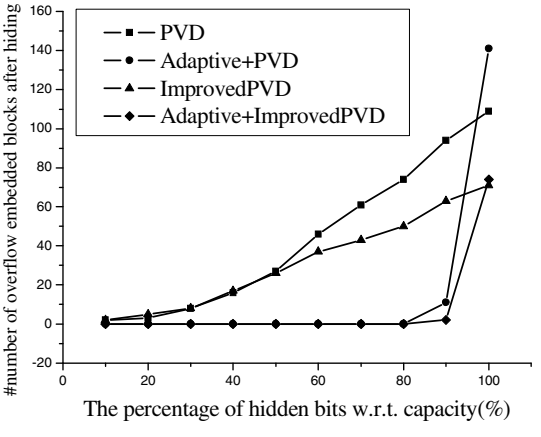


(b)

Fig. 6. The performance of proposed algorithm when the number of sub-intervals is six: (a). The relation between PSNR and message length, (b) The relation between RMSE and message length, (c) The relation between the number of overflow blocks and message length before hiding message and (d) The relation between the number of overflow blocks and message length after hiding message.

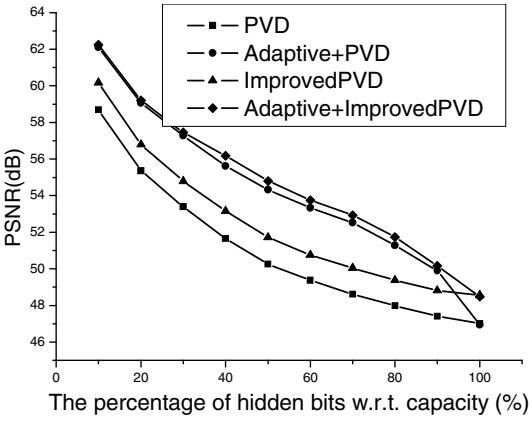


(c)

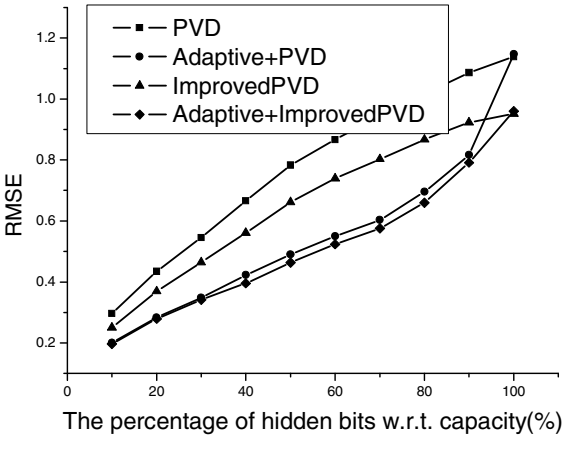


(d)

Fig. 6. (*continued*)

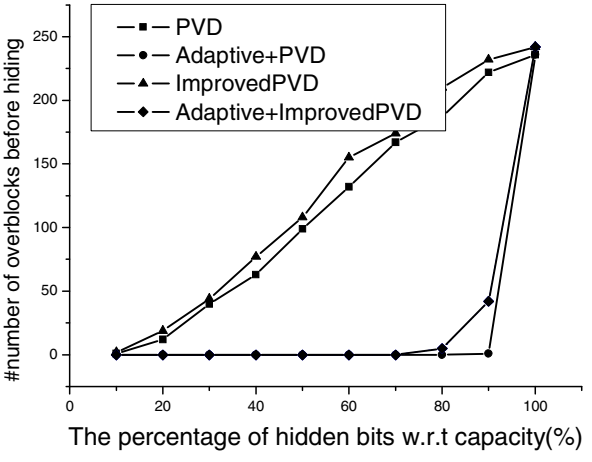


(a)

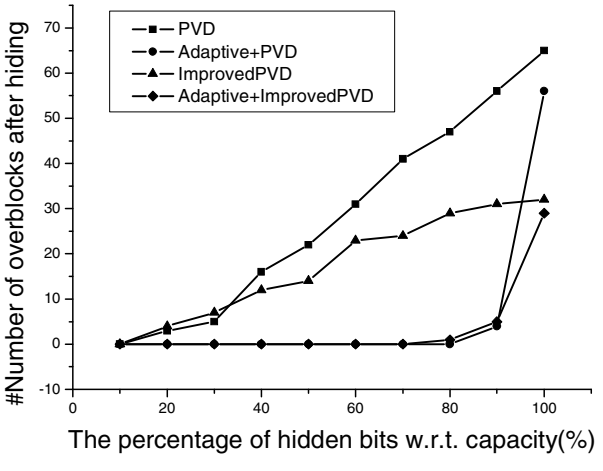


(b)

Fig. 7. The performance of proposed algorithm when the number of sub-intervals is thirteen: (a). The relation between PSNR and message length, (b) The relation between RMSE and message length, (c) The relation between the number of overflow blocks and message length before hiding message and (d) The relation between the number of overflow blocks and message length after hiding message.



(c)



(d)

Fig. 7. (*continued*)

5 Discussions about Steganalysis

Although experimental results have indicated that the proposed strategy is undetectable by the observer's senses, PSNR and perceptual quality, it does not mean that statistical analysis can not reveal the presence of a hidden message. In fact, modern steganalysis is always to break steganography techniques by using some statistical properties of stego-images. In this section, we only give a simple discussion about the proposed method's performance on resisting steganalysis.

First, we can find the abnormal phenomenon in Fig.5(a) (b), (d) and (e) where the up part and the down part of the same difference image has difference statistics, however, in (g) and (h), we do not observe the phenomenon. In fact, this abnormal phenomenon is always utilized to design the steganalysis technique. In our proposed strategy, this uneven difference image statistics is removed completely. It means that the possibility of revealing the hiding fact is decreased greatly.

Second, from (a) and (b) of the Fig.6 and Fig.7, we know the curves are increasing or decreasing smoothly, which is different from the curves in Fig.1. It means that the proposed method is a progressive quality hiding method. Hence, in steganalysis, one can not detect directly the irregular jump by increasing the size of secret message.

Third, from (c) and (d) of the Fig.6 and Fig.7, the number of over-blocks is relative larger even the size of hidden message is far less than the hiding capacity. This property also can be used in steganalysis. However, by using our proposed strategy, the number of over-blocks is few when the size of hidden message is far less than the hiding capacity of host image.

In fact, our proposed strategy not only improves the quality of stego images but also alleviates or removes some statistical properties of stego-images, and consequently decreases the possibility of steganalysis. Moreover, hiding equivalence classes are independent of each other, so we can select or design more powerful hiding or security measures into all hiding units in one hiding equivalence class even can design some complementary hiding strategies for different hiding equivalence classes to resist modern steganalysis, such as measures in F5 algorithm in [16], the two passes manipulation in OutGuess in[17] and the complementary embedding strategy in [19].

6 Conclusions

This paper analyzes the influence of specific hiding strategy on performance of steganography. It is shown that most existing data hiding algorithms have the potential spaces to improve the performance. However, different hiding strategies have different optimal measures. It depends on the specific data hiding techniques. Based on the mentioned-above observations, we propose to partition the data hiding units into difference equivalence classes in terms of the equivalence relation constructed by the hiding capacity of each hiding unit. To avoid the large distortion caused by hiding small secret message, we propose to hide the secret message from the equivalence class with the least hiding capacity to the equivalence class with the largest hiding capacity. Actually the idea not only enables us to improve the quality of stego image, but it may also allow us to use different and more efficient hiding

strategies for different equivalence classes to improve the security. This strategy can be extended to all hiding algorithms whose hiding units are independent of each other. Taking the adaptive hiding schemes [9, 10] as an example, we apply the proposed hiding strategy into them. Extensive experimental results verify the effectiveness of the proposed data hiding strategy.

Acknowledgments. This work is supported supported by the Natural Science Foundation of China (60803147), the New Teacher Program Foundation (200802131023), the Fundamental Research Funds for the Central Universities (HIT.NSRIF.2009068), the Development Program for Outstanding Young Teachers in Harbin Institute of Technology (HITQNJS.2008.048)and Major State Basic Research Development Program of China (973 Program) (2009CB320906).

References

[1] Cox, I.J., Miller, M.L., Bloom, J.A., Fridrich, J., Kalker, T.: Digital Watermarking and Steganography, 2nd edn. Morgan Kaufmann, San Francisco (2008)
[2] Wang, S.J.: Steganography Of Capacity Required Using Modulo Operator For Embedding Secret Image. Applied Mathematics and Computation 164(1), 99–116 (2005)
[3] Fridrich, J., Soukal, D.: Matrix Embedding for Large Payloads. IEEE Trans. On IFS 1(1), 390–395 (2006)
[4] Gao, X.B., An, L.L., Li, X.L., Tao, D.C.: Reversiblility Improved Lossless Data Hiding. Signal Processing 89, 2053–2065 (2009)
[5] Mielikainen, J.: LSB Matching Revisited. IEEE Signal Processing 13(5), 285–287 (2006)
[6] Chan, C.K., Cheng, L.M.: Hiding Data in Image by Simple LSB Substitution. Pattern Recognition 37(3), 469–474 (2004)
[7] Chang, C.C., Chan, C.S., Fan, Y.H.: Image Hiding Scheme with Modulus Function and Dynamic Programming Strategy on Partitioned Pixels. Pattern Recognition 39(6), 1155–1167 (2006)
[8] Yu, Y.H., Chang, C.C., Hu, Y.C.: Hiding Secret Data In Images Via Predictive Coding. Pattern Recognition 38(5), 691–705 (2005)
[9] Wu, D.C., Tsai, W.H.: A Steganographic Method For Images By Pixel-Value Differencing. Pattern Recognition 24(9), 1613–1626 (2003)
[10] Zhang, X.P., Wang, S.Z.: Vulnerability Of Pixel-Value Differencing Steganography To Histogram Analysis And Modification For Enhanced Security. Pattern Recognition 25(3), 331–339 (2004)
[11] Chang, C.C., Tseng, H.W.: A Steganographic Method For Digital Images Using Side Match. Pattern Recognition 25(12), 1431–1437 (2004)
[12] Lee, C.C., Wu, H.C., Tsai, C.S., Chu, Y.P.: Adaptive Lossless Steganographic Scheme with Centralized Difference Expansion. Pattern Recognition 41(6), 2097–2106 (2008)
[13] Lin, C.C., Hsueh, N.L.: A Lossless Data Hiding Scheme Based on Three-pixel Block Differences. Pattern Recognition 41(4), 1415–1425 (2008)
[14] Jung, K.H., Yoo, K.Y.: Data Hiding Method Using Image Interpolation. Computer Standards & Interfaces 31(2), 465–470 (2009)
[15] Liu, S.H., Yao, H.X., Gao, W., Yang, D.G.: Minimizing the Distortion Spatial Data Hiding Based on Equivalence Class. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) ICIC 2007, Part I. LNCS, vol. 4681, pp. 667–678. Springer, Heidelberg (2007)

[16] Westfeld, A.: High Capacity Despite Better Steganalysis (F5-a Steganographic Algorithm). In: Moskowitz, I.S. (ed.) IH 2001. LNCS, vol. 2137, pp. 289–302. Springer, Heidelberg (2001)
[17] Provos, N.: Defending Against Statistical Steganalysis. In: 10th USENIX Security Symposium, Washington, DC, pp. 323–336 (2001)
[18] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing 13(4), 600–612 (2004)
[19] Liu, C.L., Liao, S.R.: High-performance JPEG Steganography using complementary embedding strategy. Pattern Recognition 41(9), 2945–2955 (2008)