4

Contemporary Approaches to the Histogram Modification Based Data Hiding Techniques

Yildiray Yalman[1], Feyzi Akar[2] and Ismail Erturk[1]
[1]*Turgut Ozal University,*
[2]*Turkish Naval Academy,*
Turkey

1. Introduction

The main objective of this chapter is to present the contemporary approaches to the steganography/data hiding applications, which are based on image histogram modifications. An image histogram is a type of histogram acting as a graphical representation of the tonal distribution in a digital image. The stego images that are produced by using such data hiding techniques are inherently robust against main geometrical attacks such as rotation, scattered tiles and warping, as well as other main attacks. An up-to-date method and its example application to the latest histogram modification based steganography methods and its results are presented in detail and compared to those of the classical ones in the following sections.

Contemporary data hiding applications are usually based on computer software where a vast variety of mathematical algorithms are applied. They have recently made a challenging progress together with the new developments in computer technologies. Quite a lot of data hiding methods and their applications have been proposed since the beginning of 1950s (Cox & Miller, 2002; Ni et al., 2004). However, their initial applications in many areas were unable to ensure a high or required level of information security in time. Thus, both new data hiding techniques and their development have always received ever−increasing interest in parallel to the emerging computer technologies and algorithms (Yalman & Erturk, 2009).

Using data hiding techniques in secret communication purposes has been well proved to be promising. However, a few third parties are usually intended to extract and destroy hidden data (secret bits or stego bits) in cover media in such applications. Most known and easiest of such attacks are lossy compression, LSB changing, cropping, etc. In addition, geometrical attacks have recently appeared to usually change only the pixels' positions of an image, e.g. rotation, scattered tiles and warping. But, these geometrical attacks do not change the image histogram that plots the number of pixels for each tonal value. By looking at the histogram for a specific image, an observer will be able to judge the entire tonal distribution at a glance and he/she can identify unusual situations (comb effect, possibility of hidden data transport etc.) on it. Motivated from these points of view, one of the main objectives of this chapter is to present the contemporary approaches in steganography applications, based on image

histogram modifications. The resulting covered/stego images are reasonably robust against main geometrical attacks, which do not change the image histogram, with high quality measurements in terms of human vision system as well as statistically.

Rest of the chapter is organized as follows. Fundamentals of the digital image and image steganography are explained in the following section. Section three details both contemporary approaches to the histogram modification−based data hiding and the HSV method, its implementation, example applications in three well known images together with comparisons to those of the other classical counterparts and its steganalysis. And, final remarks are presented in the last section.

2. The digital image fundamentals and image steganography

2.1 Digital image

A pixel or picture element is the smallest item of information in a digital image (object) that is represented by a series of ***X*** rows and ***Y*** columns. Pixels are normally arranged in a two−dimensional grid and are often signified using tiny dots, squares, rectangles etc. Each pixel is the smallest sample of an original image (object), where more samples naturally provide more accurate and better demonstrations of the original. The intensity of each pixel is typically variable; for example in color systems, each pixel has classically three or four components, e.g., **RGB** (**R**ed, **G**reen and **B**lue) or **CMYK** (**C**yan, Magenta, **Y**ellow and blac**K**) respectively (Sahin et al., 2006; Cetin & Ozcerit, 2009).

Digital images are commonly saved in a grayscale mode in computer systems. The number of bits in order to represent each pixel establishes how many colors or shades of gray are allowed to be displayed. For example, in an 8−bit color mode, the color monitor uses 8 bits for each pixel, allowing displaying 2^8 (256) different colors of gray.

In most cases, many types of differences or deteriorations in numerical values of a digital image cannot be easily perceived by the **H**uman **V**isual **S**ystem (**HVS**) (Fig. 1) which initiates the idea of steganography applications performed through this natural state. In such applications a cover media such as image, video, audio or any other types of multimedia is necessary.

(a) RGB: (197, 100, 45) (b) RGB: (196, 102, 47)

Fig. 1. Magnified original pixel color (a) and stego pixel color deteriorated with hidden data (b).

2.2 Image steganography

Steganography is the art and science of hiding messages or critical information to be relayed. The term steganography is derived from the Greek words steganos (στεγανός) meaning "covered or protected" and graphei (γράφη) meaning "writing". Steganography, therefore, is the all means for covered writing (Fig. 2).

Today, the term steganography states the disguise of secret/critical digital information within computer files. For example, a sender might start with an ordinary−looking digital image file, and then adjust the color of every 10th pixel to correspond to a letter in the alphabet (a change so subtle that anyone, who is not actively or intentionally looking for it, is unlikely to perceive it). It differentiates from the cryptography in that the latter conceals the meaning and content of a secret message, though is unable to conceal the fact that there is a message (Yalman, 2010; Papapanagiotou et al., 2005). Both steganography and cryptography can be combined for optimum and highly reliable communication security (Akar, 2005).

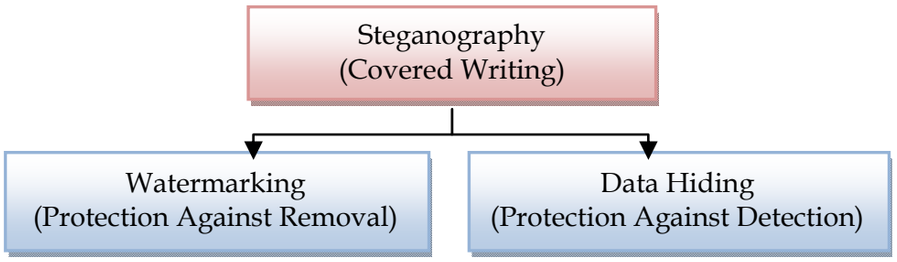
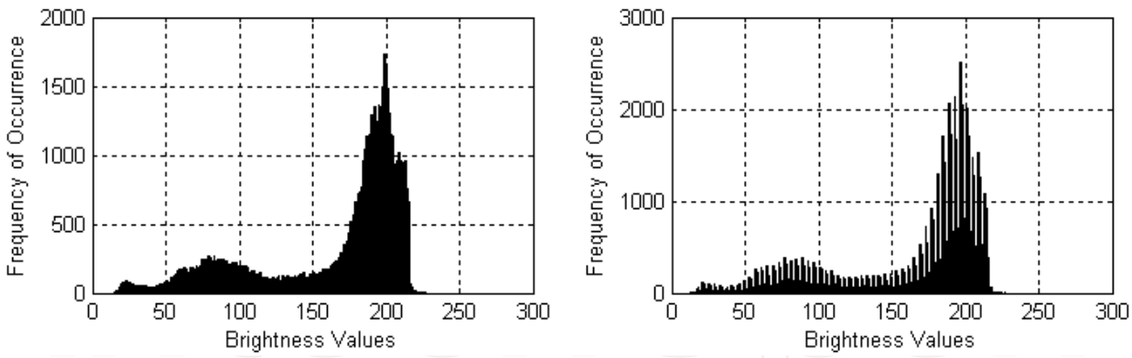


Fig. 2. Directions within steganography methods.

There are a lot of studies about digital image steganography presented in the literature. Almost all of these proposed methods have diverse effects to the image (cover media) due to adding noise or deterioration on it. Although, the HVS is unable to detect these distortions, this situation is totally different, considering the distribution of brightness values on the image histogram. For example, while the HVS cannot sense the differences between images presented in Fig. 3−a and −b (original and stego images) itself, it can easily recognize the difference between their histograms given in Fig. 4−a and −b.

(a) (b)

Fig. 3. Original image (a) and stego image encoded by using LSB−2bits (b).



(a) (b)

Fig. 4. Original image (a) and stego image (b) histograms.

This change in the image histograms is called as "comb effect" in literature (Yalman & Erturk, 2009). It basically points out the unbalanced/deteriorated brightness value distribution and may easily lead to the detection of the covered message.

In Fig. 4−a and −b, not only are the image histogram appearances different but also the frequency of occurrence of the brightness values are extremely fluctuated. This natural fact can easily be comprehended by doing a simple check on the stego image histogram without even knowing the original image histogram. As a result one can basically assume that the image had been processed for a reason such as conveying a secret message/information. Regarding all of these important points, histogram−based data hiding applications are highlighted in this chapter because they all aim at producing a stego image histogram without revealing the combing effect.

2.3 Quality measures used in image steganography evaluation

Here **P**eak **S**ignal to **N**oise **R**atio (**PSNR**) and **M**ean **S**quare **E**rror (**MSE**) parameters are considered for statistical analysis of the steganography methods. The MSE should be computed first as given in equation (1) and equation (2) (Sencar et al., 2004) then the PSNR can be derived as in equation (3) (Netravali & Haskell, 1995; Rabbani & Jones, 1991), where "O" and "S" are the original and stego image pixel values (binary) respectively to be compared and the image size is "X × Y". PSNR result of the stego images produced by all of the histogram−based data hiding techniques is guaranteed to be above the other classical techniques' performance in terms of statistical and perceptual invisibility. Note that, equations (1) and (2) are specified for only monochrome images; for color images, the denominator of the equation (3) is multiplied by a factor 3. To compute the PSNR, the block first calculates the mean−squared error using the following equation:

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left\| O(i,j) - S(i,j) \right\|^2 \tag{1}$$

$$MSE = \frac{\sum_{m,n} \left[O(i,j) - S(i,j) \right]^2}{m \times n} \tag{2}$$

$$PSNR = 10\log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{3}$$

Steganography methods and their applications are validated through well−known quality measures. PSNR value is the fundamental metric but it does not match with the HVS exactly. For this reason, different quality measures have been presented and discussed in the literature for the last decade. In addition to PSNR, there are several perceptual measures such as **U**niversal Image **Q**uality **I**ndex (**UQI**) (Wang & Bovik, 2002), **V**isual **I**nformation **F**idelity (**VIF**) (Sheikh & Bovik, 2006) and **M**ean **S**tructural **S**imilarity (**M−SSIM**) (Wang et al., 2004) in order to evaluate and analyze the data hiding methods. The UQI, VIF and the M−SSIM are measured as a quality result (**Q**) that ranges between [−1 and 1], between [0 and 1] and between [0 and 1] respectively, meaning that the best Q value can be 1 for all of them. All of quality measures mentioned above are based upon statistical techniques whose results are compatible with the HVS.

3. Contemporary approaches to the histogram modification−based data hiding

In this section, histogram modification−based data hiding methods presented in the literature are explained. In addition to these, a contemporary method called as HSV, its application and its steganalysis are described in detail.

3.1 Histogram modification−based data hiding

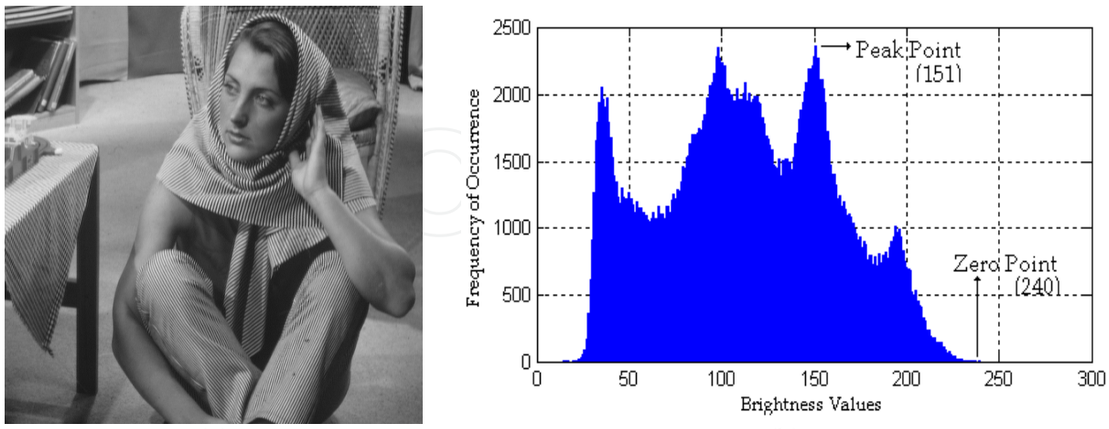
By looking at the histogram of a specific object, an observer would be able to judge the entire tonal distribution of the image at a glance. Histogram modification-based data hiding methods utilize this asset of the digital images to convey any type of secret information.

Ni et al. (2006) initially introduced a histogram based data hiding technique where the crucial information is embedded into the image histogram. Pairs of peak points and zero points are used to achieve low embedding distortion with respect to providing low data hiding capacity. Another histogram modification technique for data hiding has been extensively worked out recently in Fallahpour and Sedaaghi's paper (Fallahpour & Sedaaghi, 2007). It is fundamentally based on block−based. Lee et al. (2006) also proposed a reversible data hiding scheme based on histogram modification of difference images. To increase data hiding ability, Chang et al. (2008) presented an efficient extension of the histogram modification technique by considering the pixel difference instead of simple pixel value and Teng et al. (2010) had some other similar proposals. They also exploited a histogram shifting technique to prevent problems raised about overflow and underflow. Some of these methods are described in the following sub−sections.

3.1.1 Ni et al.'s method

Ni et al. (2006) developed a stimulating algorithm for hiding data in gray level images. The algorithm is based on image histogram modification. If a grayscale image is used as a cover image, firstly, its histogram is generated (Fig. 5 (a)). Moreover, a Peak Point (PP) and a Zero Point (ZP) are determined in the histogram (Fig. 5 (b)). The PP occurs on gray Brightness

Value (BV) 151 and the ZP occurs at 240 as illustrated in Fig. 5 (b). Making the capacity as large as possible is the aim of the discovery of the PP. The number of bits hidden/inserted in the cover image is the same as the number of pixels related to the PP.



(a) (b)

Fig. 5. The Barbara image (a) and zero point and peak point of its histogram (b).

It is assumed that the PP is less than the ZP. Additionally, the whole cover image is scanned in a specific order (e.g., from top to bottom, right to left). At the first step, the gray BV of the pixels in the range (PP, ZP) is increased by 1. As for the example of the Barbara image, the histogram in the range between 151 (exclusive) and 240 (inclusive) is shifted 1 unit to the right, as demonstrated in Fig. 6.

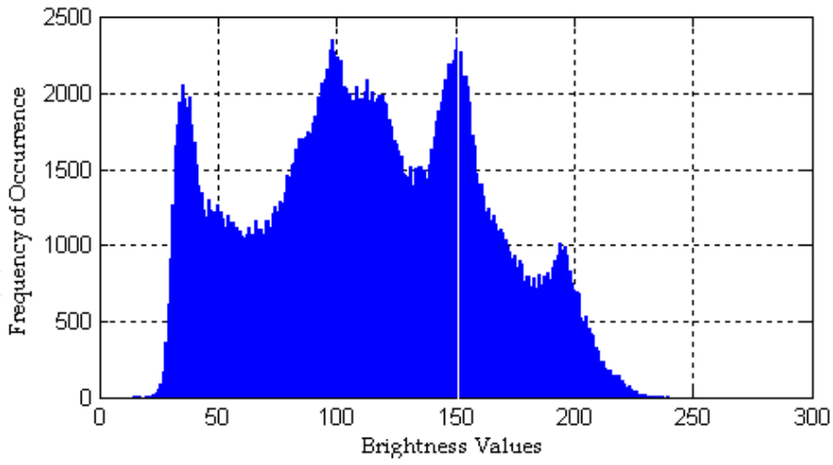


Fig. 6. A crack appears on the original peak BV.

Secondly, the obtained histogram−shifted image is scanned once again in the same order. Once a pixel with gray BV is encountered, the binary sequence of the secret information is explored to be hidden. If the corresponding bit in the sequence is $(1)_2$, the pixel value is increased by 1. Otherwise the pixel brightness value (BV) remains unchanged. As far as the Barbara image in Fig. 5(a) is concerned, it is scanned to seek for the pixels with gray value 151 one at a time. If the corresponding bit in the secret data is binary "1"then the pixel is

extended to 152, else it is kept as 151. Finally, the original peak in the histogram is slightly changed as given in Fig. 7.

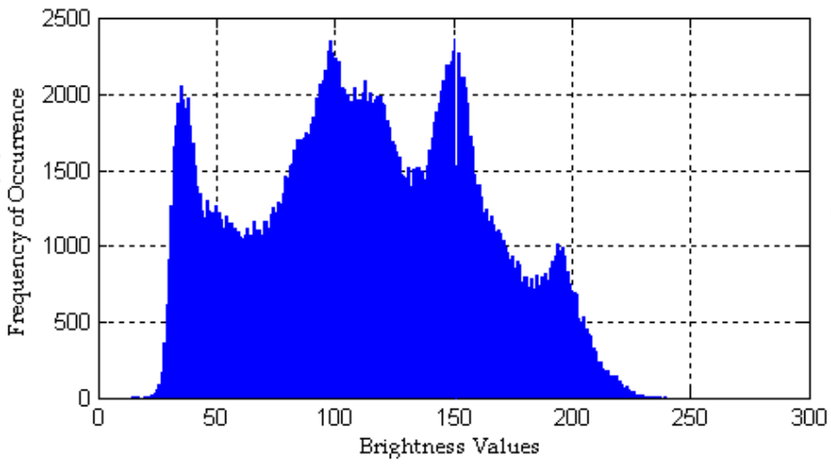


Fig. 7. The histogram after embedding secret data.

When we need to extract the embedded data from the stego−image by using the above algorithm, finding the PP and the ZP are necessary to realize first. Generally, the BVs of PP and ZP are considered as side information and transmitted to the receiver necessarily. In addition, all communication parties are able to modify the BV of the pixel in the specific position, from the upper leftmost pixel to sign PP and ZP in the stego−image.

As a result, the digital images coded by this method cannot be detected by HVS. But steganalysist can still easily notice changes on image histogram because deterioration on its view is marginal as seen in Fig. 7. Above all, Ni et al.'s (2006) method offers very low data embedding capacity making it very unsuitable in current steganography applications.

3.1.2 Teng et al.'s method

Teng et al. (2010) slightly improved Ni et al.'s method detailed above. In accordance with Ni et al.'s method, the data hiding capacity of the cover image directly depends on the frequency of occurrence of the pixels. If it is expected to enhance the data hiding capacity, they offer that they should make more pixels with the BV of the peak. A method to enlarge the number of pixels with the BV of the PP is revealed by taking advantage of the histogram re−quantization.

The histogram quantization is a method in order to reduce the number of bins of the histogram by mapping the pixels into fewer levels. Teng et al. (2010) mapped the pixels in the digital image into 64 levels. Fig. 8 depicts the histogram of the Barbara image with 64 levels based on histogram re−quantization.

Comparing Fig. 8 to Fig. 5 (b), the frequency of occurrence of pixels in peak increases more than 2 times. The hidden data is to be concealed in these pixels. The encoding steps of the Teng et al.'s (2010) data embedding algorithm are given as follows:

1. Assume that D is the difference between two nearest signified gray levels as the total number of bins and generate the histogram of the cover image in 256/D levels.
2. Find the peak bin [PP, PP+D) and the zero bin [ZP, ZP+D) in the histogram.
3. If PP<ZP then a pixel with gray BV in the range of [PP+D, ZP) is increased by D. Afterwards, the cover image is scanned in the given order and the pixels with gray BV in the range [PP, PP+D) are increased by D if its corresponding bit in the secret data is binary "1". If PP>ZP, then a pixel with gray BV in the range of [ZP+D, PP) is decreased by D. Afterwards, the cover image is scanned in the given order and the pixels with gray BV in the range (PP−D, PP] are reduced by D if its corresponding bit in the data is binary "1".
4. Get the stego−image and relay it to the receiver.

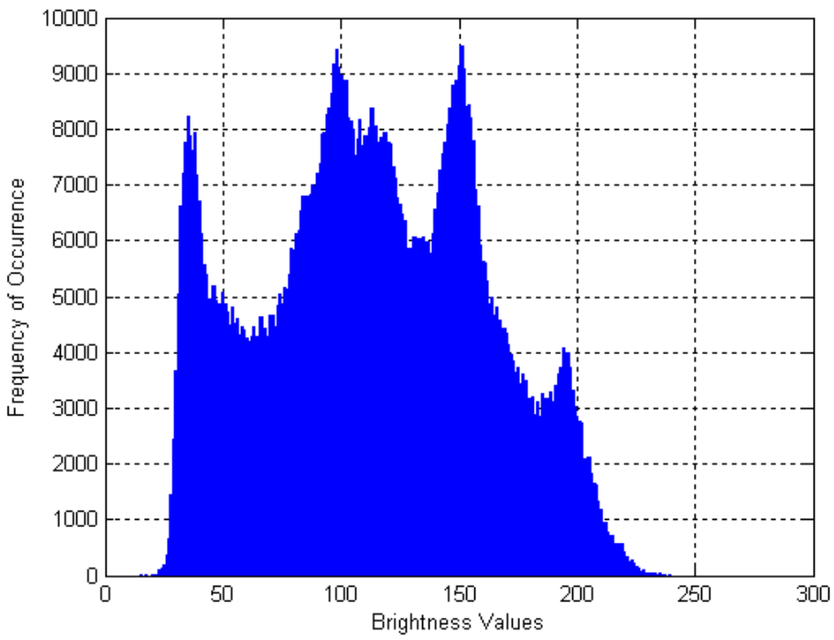


Fig. 8. The histogram of the Barbara image with 64 levels.

According to the Teng et al.'s (2010) method, the value of 4 or 8 is recommended for D because the image quantized into 64 or 32 levels is generally lossless for HVS.

Data extracting part of algorithm are given as follows:

1. The stego−image is scanned in the same given order.
2. The pixel with gray BV [PP, PP+D) indicates the corresponding data bit is 0.
3. If PP<ZP, then a pixel with gray BV [PP+D, PP+2D) indicates the corresponding data bit is 1.
   If PP>ZP, then a pixel with gray BV [PP−D, PP) indicates the corresponding data bit is 1.
4. Get the hidden embedded data.

This method offers more capacity than Ni et al.'s method but it still offers small capacity and unacceptably low PSNR values according to the other histogram based data hiding approaches.

3.1.3 Krishna et al.'s method

Krishna et al.'s (2010) introduced another histogram based data hiding algorithm too. However it is a reversible data hiding technique based on histogram modification using pairs of PP and ZPs has the process of adding '1', if peak BV of pixel has been encountered. Otherwise, '0' is added, i.e. if zero is detected. From this they can estimate the number of pixels in the image with peak BVs. However, for an unusual image with equal histogram, with this technique minimum points can be embedded. Also the peak and minimum points should be requirement of the receiver for full recovering.

As an alternative to using pixel BV, the differences between neighbor pixels are considered; they get the differences have almost a zero−mean and a Laplacian−like distribution. Although this leads to a partial improvement in data embedding ability, still the PP pairs are tallied to the receiver. To demonstrate the method, consider an 8−bit grayscale cover image with a pixel BV, x_i denoting the grayscale BV of i^{th} pixel (between 0 and 255). The image is scanned in inverse s−order and the differences between neighbor pixels are calculated. By determining the PP from the pixel differences, another scan is realized on whole image in inverse s–order as previously.

If the difference is higher than the peak BV, the secret data bit cannot be embedded so x_i is shifted by 1 unit. However, for the pixel difference is less than the peak BV, a secret data bit is embedded. At the receiver side, the whole image is scanned in the given same order and the secret data bit is extracted from the stego image. Here Krishna et al.'s (2010) suggests using only one PP for data hiding. For large data embedding capacities above process is reiterated and the PPs to be noted for every hiding pass. In order to correspond the multiple PPs, a binary tree structure is designed as presented in Fig. 9.

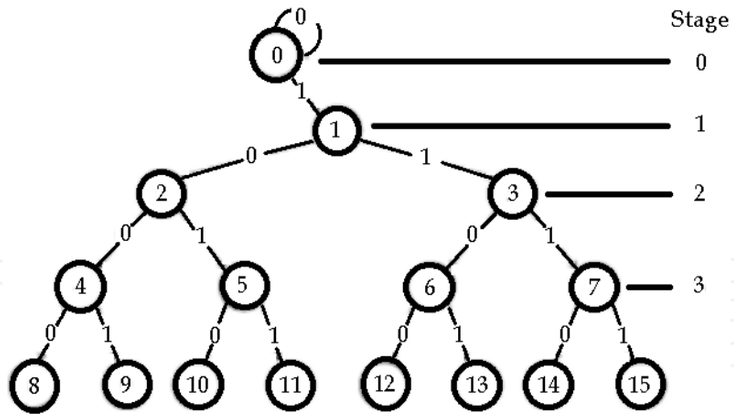


Fig. 9. The binary tree structure.

PPs are assumed to be 2^S for embedding secret data, where S is a stage of binary tree. Here each node is a PP. If the pixel difference is less than the peak BV, secret data bit can be embedded. If '0' bit to be added, the left sub-node is visited otherwise the right sub-node is visited. Employing this method the payloads are increased; tree stages are also increased in turn image distortion is increased too. All recipients need to share with sender its tree stage S, because this tree establishes multiple PPs. If the difference is higher than the peak BV secret data bit cannot be embedded so x_i is shifted by 1 unit. However, for pixel difference less than

the peak BV, a secret data bit is embedded. At the receiver side, the whole image is scanned in the same order and the secret data bit is extracted from the stego image (Krishna et al.'s, 2010).

Pixels are modified in the above process, which might lead to overflow or underflow. Krishna et al. (2010) suggest narrowing the histogram from both sides to avoid overflow and underflow. So the method shifts both sides by 2^S units meaning that the histogram is narrowed in the range 2^S, $255-2^S$. The histogram shifting information is recorded as overhead book maintaining information that has to be hidden into the cover image itself with payload.

Data embedding and extracting algorithms of the Krishna et al.'s method are as follows:

a. Embedding process:

An N−pixel 8−bit grayscale image with pixel value x_i (between 0 and 255) is considered for embedding process (Krishna et al.'s, 2010).

1. Find the stage S of the binary tree. S determines the data embedding capacity.
2. Histogram is shifted from both sides narrowing in the range 2^S, $255-2^S$, still maintaining the shifting information in the payload.
3. Scan the entire image in an inverse s−order and find differences between neighbor pixel BVs.
4. Again scan the entire image in inverse s−order. Determine if x_i should be shifted by 2^S.
5. If the difference is less than 2^S, embed a secret data bit with x_i.

b. Extraction process:

At receiver side hidden data is extracted from the stego image and the original image is recovered with the help of S stage of a binary tree (Krishna et al.'s, 2010).

1. Scan the stego image in inverse s−order.
2. Calculate the differences to extract the embedded secret data bits.
3. Recover original cover image by shifting in the reverse order as done during the data embedding process.
4. Repeat step 2 until the embedded secret data is fully extracted.
5. Extract the overhead information from the extracted secret data.

3.2 A contemporary histogram modification−based data hiding method: HSV

The HSV method differs from the existing histogram−based data hiding methods regarding the following two important aspects. First of all, stego images and their histograms cannot be detected by the HVS. Secondly, data hiding capacity is comparatively higher than the other methods. Its algorithm can be easily applied because it is too simple. Following sub−sections detail the method.

3.2.1 HSV method

This part is mainly based on Yalman & Erturk's method (Yalman & Erturk, 2009) that is shortly called in (Yalman, 2010) as **HSV (H**istogram based **S**teganography on **V**ideo**)**. In this scheme, the HSV approach mainly utilizes the LSB secret data embedding technique and histogram processing.

The HSV and its algorithm principally modify a cover image's histogram for data hiding where neither the resulting new stego image nor its histogram is noticeably different from the original. Therefore, they are both perceived exactly same as the original ones by the HVS (Human Visual System). The HSV method considers the **F**requency of **O**ccurrence (**FO**) of the pixel **B**rightness **V**alues (**BVs**) of the cover image, and then the data hiding process is realized based upon it. First of all, the cover image histogram is produced where the lowest and the highest BVs are determined and named as the Lowest **B**rightness **V**alue (**LBV**) and the **U**pper **B**rightness **V**alue (**UBV**), respectively. These two margins are used to indicate where the process of data hiding could be accomplished (Fig. 10). The idea of the HSV and its implementation processes are explained with an example as follows;

Let's assume the first three stego bits of the secret data are $(010)_2$ and the cover image's histogram produced is given as in the Fig. 10 where the FOs of the LBV (i.e., 22), 23 and 24 BVs are "6", "18" and "31" (Table 1).

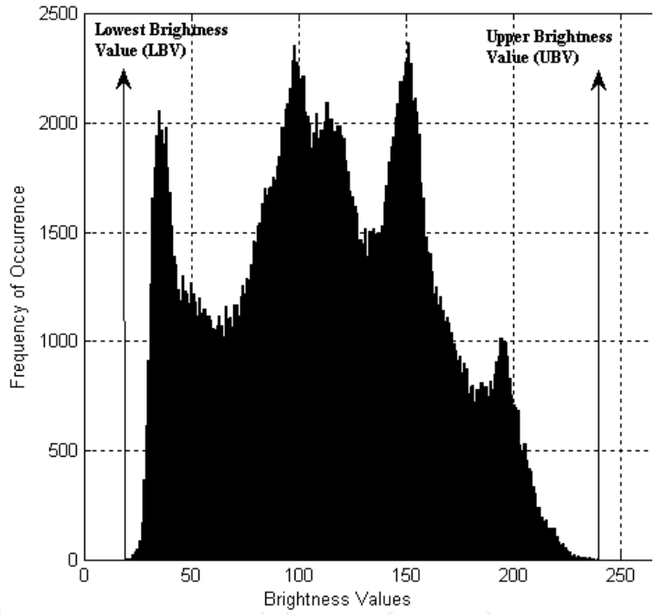


Fig. 10. Determining the LBV and UBV of an image histogram.

Brightness Values (BVs)	22	23	24
Frequency of Occurrence (FOs) of the BVs	6	18	31

Table 1. The first three BVs and their FOs.

At the first step, the algorithm takes that the first stego bit is "0" and the FO of the LBV is 6. Then it computes the Mod2 of this FO as "0" (6 Mod2 = 0). After that, whether the stego bit and the Mod2 process result are equal is checked. If so, as in this example (i.e., 0 = 0), the BV remains as it is (i.e., 22) meaning that it contains now the stego bit "0".

It then proceeds with the following second stego bit "1" and the FO (i.e., 18) of the next BV (i.e., 23) as the second step. Similar to the previous step, the algorithm computes the Mod2

of this FO as "0" (18 Mod2 = 0). After that, whether the stego bit and the Mod2 process result are equal is checked. Since they are not equal (i.e., $1 \neq 0$), now one pixel of the image, whose BV is 23, is changed to the next following BV (i.e., 24). Thus, the resulting FOs of the BV(23) and BV(24) are changed to "17" and "32" respectively, which means that the new FO of the BV(23) contains now the second stego bit "1" (Table 2).

These two processes are applied for the following stego bits and the FOs of the BVs repeatedly until all of the stego bits are embedded in or the UBV is reached indicating the full image data hiding capacity is already utilized.

Brightness Values (BVs)	22	23	24
Frequency of Occurrence (FOs) of the BVs	6	17	32

Table 2. The first three BVs and their FOs after completing $(010)_2$ hiding processes.

The HSV fundamentally differs from the well−known LSB data hiding method, where the least significant bits of the digitalized values of the pixels are modified, in that the frequency of occurrence of the cover image brightness values are modified to embed secret data in. It intrinsically overcomes the salt & pepper effect (Ni et al., 2008) as the boundaries of the resulting brightness values of the stego image are always between "0" and "255". In addition, considering its usage in 24−bit RGB color images (i.e., producing three different histograms for the R, G and B channels of each pixel) separately, a much higher data hiding capacity is well achievable. Another outstanding aspect of the HSV is about dynamically increasing the data embedding capacity (if required), well trading off the PSNR results insignificantly. This is easily achievable dividing the cover image into multiple parts and then applying the method to each part concurrently (Yalman & Erturk, 2009).

Obtaining the stego bits from the stego image using the HSV is much easier than the above secret data embedding process. Initially the stego image histogram is obtained for the secret data extraction process and the LBV & UBV of the histogram are determined. After that the Mod2 process is realized starting from the FO of LBV (i.e., "6" in the example), which is repeated until the UBV is reached in the histogram (Yalman & Erturk, 2009). Considering the data hiding example, following processes explain revealing the stego bits (i.e., "$(010)_2$") extracted from the stego image histogram values (Table II). Thus the stego bits are calculated as follows;

$$\left.\begin{array}{l} 6 \text{ Mod2} = 0 \\ 17 \text{ Mod2} = 1 \\ 32 \text{ Mod2} = 0 \end{array}\right\} \longrightarrow (010)_2$$

3.2.2 Example HSV applications and comparative analysis of the results

Experimental visual results of the HSV approach on the well−known images Lena, Baboon and Peppers are displayed in Fig. 11 (Yalman & Erturk, 2009). The visual differences between the original cover images (i.e., Fig. 11−a, −b and −c) and the corresponding stego

images with random hidden data (i.e., Fig. 11−d, −e and −f) can be hardly detected by the human eyes (i.e., the HVS). This is the most important of any steganography application, well achieved by using the HSV.

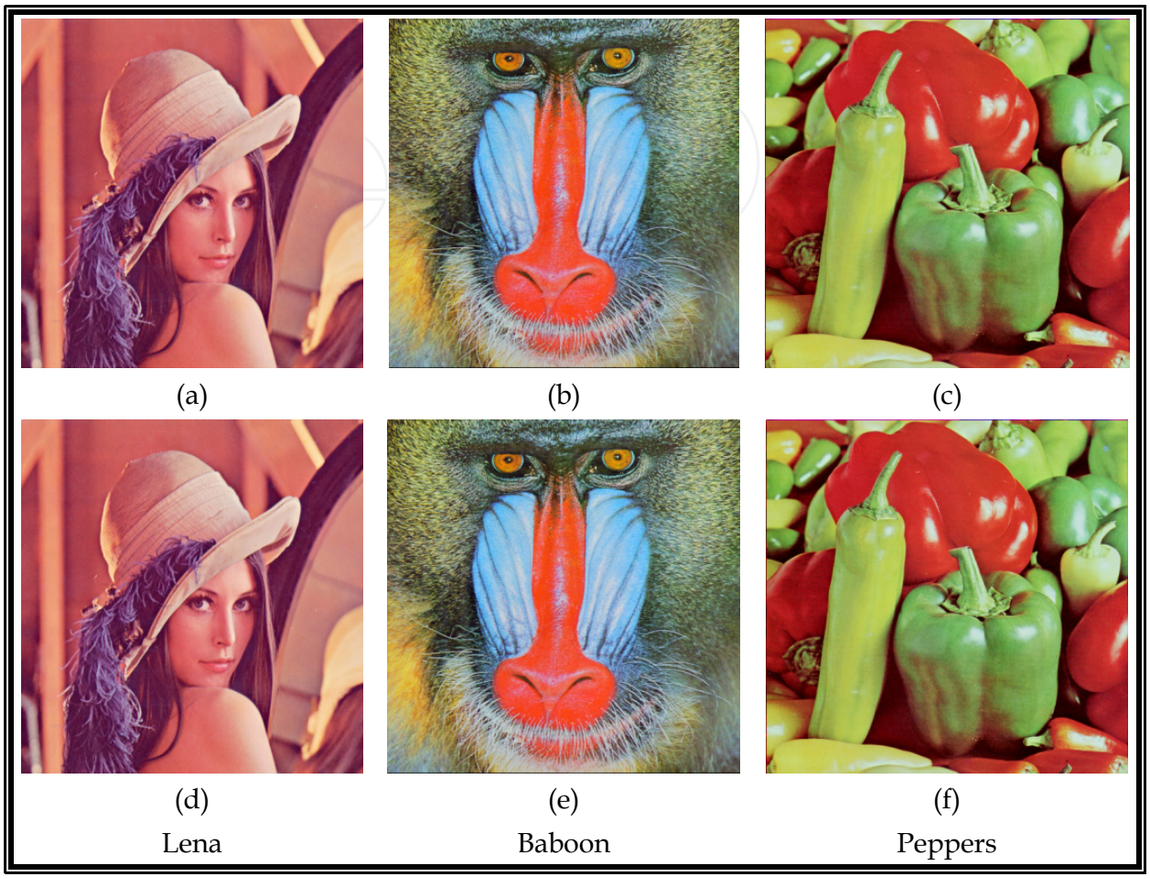


Fig. 11. Original cover images (a−b−c) and the HSV stego images (d−e−f).

The differences among the original, HSV stego and a classical stego images can be realized by the HVS when images are enlarged/magnified highly (Fig. 12). In Fig. 12−c the picture contains more hidden data but this result is not enough due to the fact that the distortion on Lena image is noticeable and this is unacceptable for the users.

Fig. 13 shows the most crucial and valuable aspect of the HSV, in which the Lena is utilized as the cover image as an example. Fig. 13−a, −b and −c are the original cover image R, G and B histograms respectively. Fig 13−d, −e and −f are the stego image R, G and B histograms obtained from the resulting stego image applying the HSV while Fig. 13−g, −h and –i belong to the stego image R, G and B histograms obtained from the resulting stego image applying another traditional method (Yalman & Erturk, 2009). It is clearly understood that there is almost no change on the R, G and B histograms as a result of utilizing the HSV with respect to their original histograms. Moreover, the HVS cannot sense the changes resulted from the data hiding process. On the other hand, many of the classical data hiding algorithms result in highly fluctuating and combing changes in the stego histograms, for instance in the RWB technique (e.g., Fig. 13−g, −h and −i) (Akar & Varol, 2004).

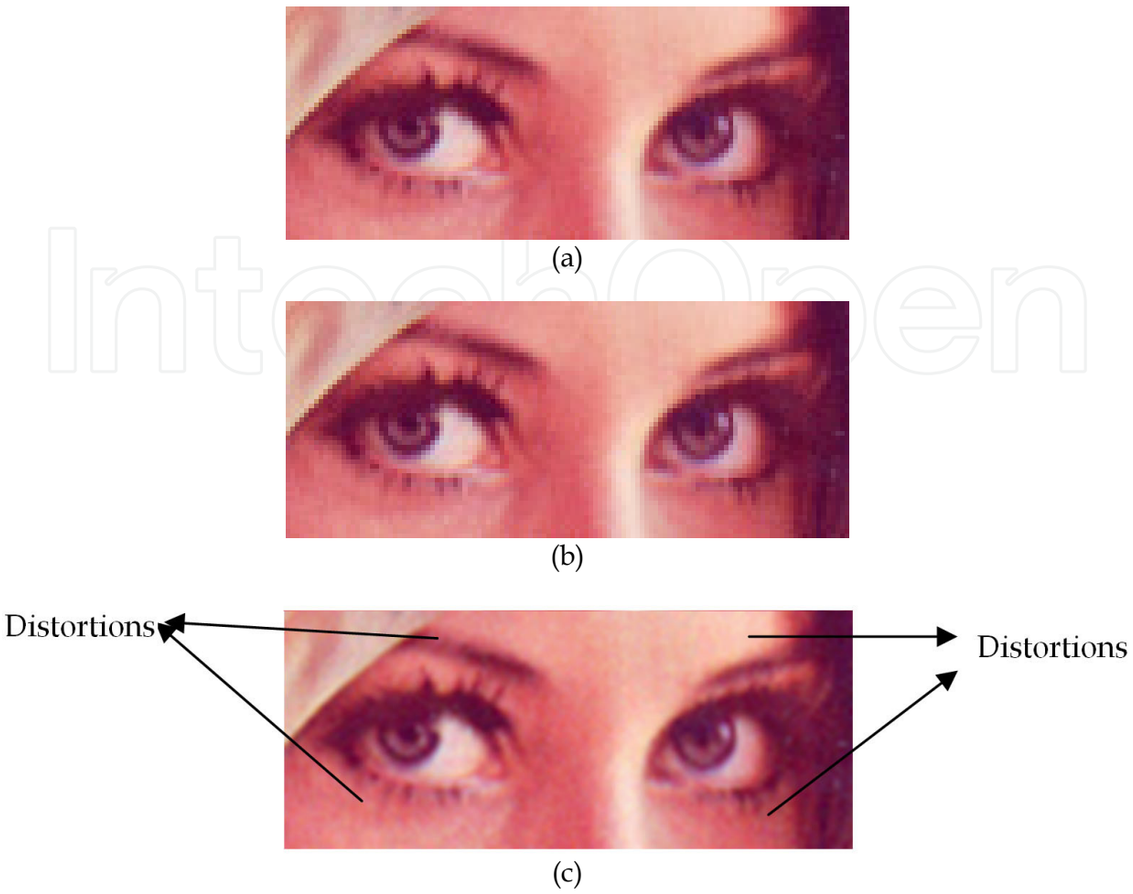


Fig. 12. Original and marked "Lena" images; original (a), coded by HSV (b), coded by classical RGB weight based (RWB) encoding technique (c).

Another important issue in comparing the HSV data hiding approach with its counterparts (e.g., presented in Chrysochos et al.'s (2007) paper that is also based on histogram modifications) that produce almost all the same stego image histograms as HSV does is the PSNR performance with respect to the maximum secret data embedding capacity.

The HSV always results in better PSNR values as well as doubles the data embedding capacity compared to those of Chrysochos' algorithm (Table 3, Table 4). It should be noted

	Chrysochos et al.'s (2007)		**HSV**	
	PSNR (dB)	*Embedded Data (Bits)*	*PSNR (dB)*	*Embedded Data (Bits)*
Lena	54.12	360	62.75	665
Baboon	53.10	300	59.25	747
Peppers	55.18	300	56.01	700

Table 3. Experimental results for different images and embedded bits (Yalman & Erturk, 2009).

that the maximum secret data embedding capacity of the HSV is directly related to the difference between the UBV and the LBV. Therefore, it is a very clear fact that HSV is mostly suited to the cover images with brightness value histograms uniformly distributed from "0" to "255" with respect to data embedding capacity.

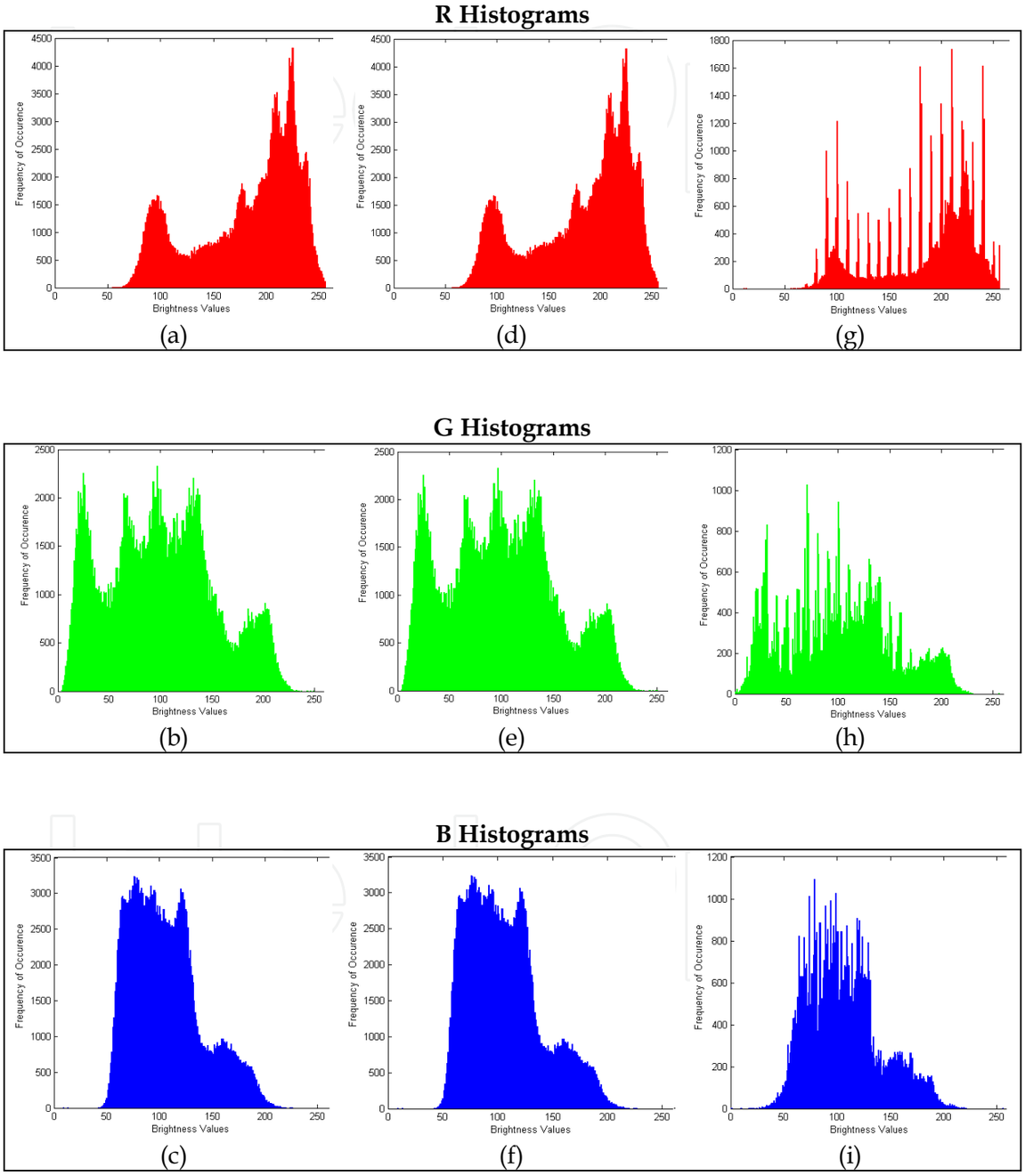


Fig. 13. The original Lena R−G−B histograms (a−b−c) and their stego image counterparts applying both HSV (d−e−f) and the RWB (g−h−i).

	Embedded Data (bits)	**Chrysochos et al. ′s (2007)**	**HSV**
		PSNR (dB)	*PSNR (dB)*
Lena	360	54.12	72.59
Baboon	300	53.10	84.81
Peppers	300	55.18	80.84

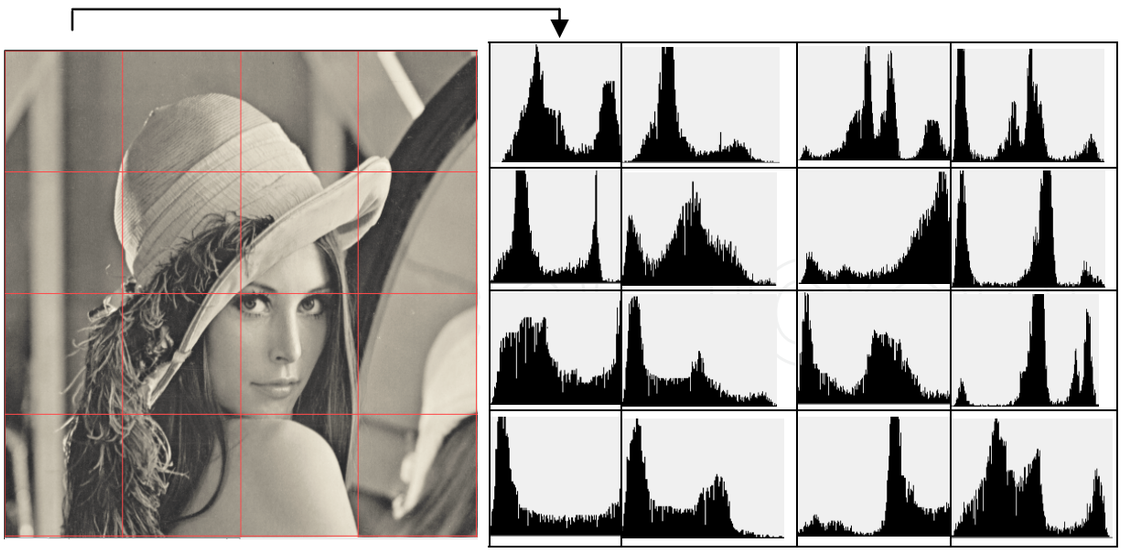
Table 4. Comparisons of PSNR results for the same data embedding capacity (Yalman, 2010).

As mentioned above, only a PSNR analysis is not adequate for a complete quality assessment of any steganography method. In addition to PSNR; VIF, UQI and M−SSIM visual quality measures are also used for the performance comparisons in this chapter (Table 5). Considering these parameters, the HSV method gives results that are closer the finest quality (about 1).

	RWB	**LSB (2bits)**	**LSB**	**HSV**	**RWB**	**LSB (2bits)**	**LSB**	**HSV**
	Lena				**Baboon**			
VIF	0.9798	0.9802	0.9981	**0.9993**	0.9823	0.9888	0.9930	**0.9997**
UQI	0.9237	0.9440	0.9988	**0.9995**	0.9724	0.9814	0.9945	**0.9998**
M−SSIM	0.9531	0.9654	0.9991	**0.9997**	0.9801	0.9890	0.9980	**0.9999**
	Peppers				**Airplane**			
VIF	0.9573	0.9786	0.9980	**0.9998**	0.9571	0.9325	0.9976	**0.9997**
UQI	0.9012	0.9435	0.9976	**0.9999**	0.8375	0.9390	0.9950	**0.9997**
M−SSIM	0.9366	0.9703	0.9986	**0.9999**	0.9422	0.9612	0.9980	**0.9998**

Table 5. Experimental results for different statistical metrics for different 512×512 gray images coded by using four steganography methods.

The data hiding capacity of the HSV can be improved by applying a histogram stretching technique (or histogram equalization) that consequently increases the difference between the LBV and the UBV of an image histogram or HSV can be applied in small pieces of image to increase the capacity (Fig. 14). If the method is applied onto the well decided pieces of the cover image, it will increase the payload size exponentially as the results in Fig. 15 show it clearly.



(a) (b)

Fig. 14. Implementation of the HSV to small parts (b) of the image (a).

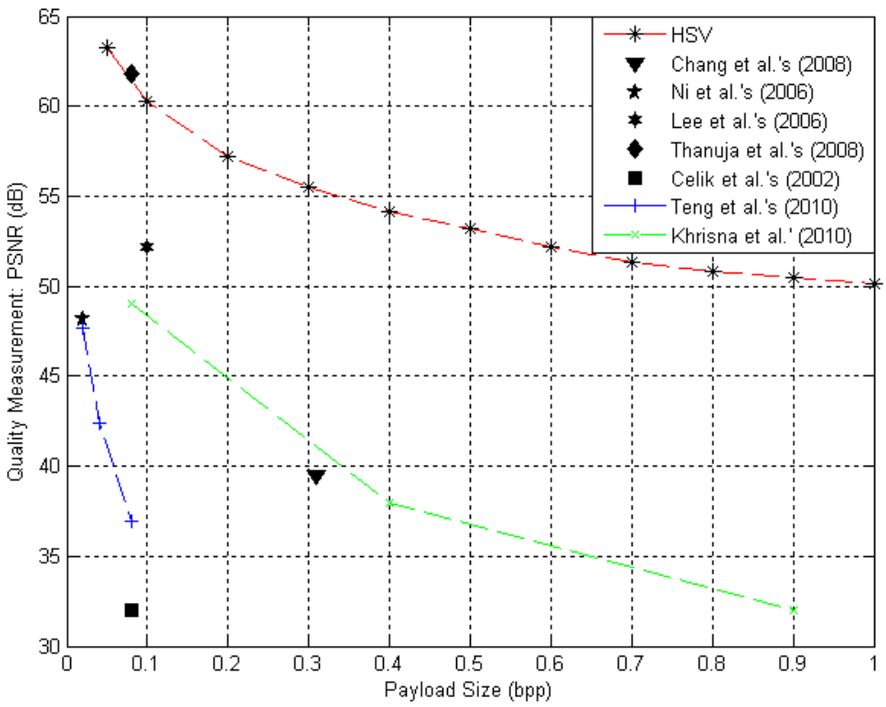


Fig. 15. PSNR versus payload size for test image Lena (512×512).

In addition to the above distinguishing results, the HSV is also uniquely strong against geometric attacks like the other classical histogram based steganography methods. The geometric attacks usually change the pixel positions of a digital image, but unable to effect the image histogram. Based on this fact, the HSV is clearly defiant to such attacks as rotation, scattering tiles, warping etc. (Fig. 16).

(a) Scattered Tiles (b) Rotation (45°) (c) Warping

Fig. 16. Main geometrical attacks against a stego image (Peppers).

3.2.3 Steganalysis of the HSV

In this sub−section, the HSV method is studied using three well−known steganalysis methods that are namely the Raw−Quick−Pairs (RQP), Stegdetect Tool and Visual Steganalysis. The primary objective of any steganographic method is to cover and transmit secret information within a cover image (Lin et al., 2008). If this fact is probable for an attacker to find out or to realize that a cover image has hidden information then thisobjective of the data hiding method is failed. Such type of attacks is called as detection attacks (Fridrich et al., 2000).

The RQP steganalysis method is fundamentally based on statistics of the numbers of unique colors and close−color couples in a 24−bit RGB image (Fridrich et al., 2000). A couple of colors are defined as close if

$$|R_1 - R_2| \le 1, |G_1 - G_2| \le 1 \text{ and } |B_1 - B_2| \le 1. \quad (4)$$

When data are embedded into a digital cover image using the LSB−based data hiding technique, the number of unique colors U generally increases so that the following measure will increase (Fridrich et al., 2000):

$$Q = \frac{P}{\binom{U}{2}} \quad (5)$$

where P is the number of close−color pairs and the number of all possible color pairs/couples in the color palette. If the cover image already contains a hidden message and after embedding a known new data on top of it, the difference between the Q_1 and the Q_2 values is marginally small (e.g., less than 0.009). Otherwise, a high variation between the Q_1 and the Q_2 values above 0.01 means that the steganography method producing the stego image is confirmed (Sahin et al., 2007). Thus, it is easily possible to sense the occurrence of a secret data by adding a test data into a stego image and observing the amount of variation in Q .

Three different stego images (Fig. 11−d, −e, −f) that are produced by using the HSV are evaluated using the RQP method and the results are presented in Table 6. Easily understood from all of the examples, the variation in Q values of the stego images and the stego images with test data ($|Q_1 - Q_2|$) are such high that one could not realize the existence of the hidden data in the cover images, justifying and validating the HSV method.

Stego image	Q_1 **(for the stego image)**	Q_2 **(for the stego image with test data)**	$\|Q_1 - Q_2\|$
Lena	0.24621	0.22568	0.02053
Baboon	0.40172	0.38625	0.01547
Peppers	0.31940	0.30851	0.01089

Table 6. RQP steganalysis of the HSV.

Robustness of the HSV is also checked using a software tool called as Stegdetect (OutGuess Steganography Detection Tool). The output from the Stegdetect lists either a steganographic application found in the cover image or "negative" if no steganographic substance could be detected. The results for the three stego images given in Fig. 11−d, −e, −f) are all obtained as negative as given in Table 7. Therefore this second method also well confirms the use of HSV and its consistency.

Stego image	**Stegdetect Result**
Lena	Negative
Baboon	Negative
Peppers	Negative

Table 7. Stegdetect steganalysis of the HSV.

Since human beings have really sophisticated pattern recognition capabilities that are mainly optimized for images, one possibility is that they will be superior to the other classical techniques (e.g., RQP and Stegdetect), and human visual observation may succeed where steganalysis methods based on statistical analysis are not successful (Watters et al., 2005). In order to justify the superiority of the HSV, an output stego image and the original cover image are depicted in Fig. 17 for visual observation. Having done many ordinary user visual tests, none has come up with the conclusion that the figures are different in anyhow.

(a) (b)

Fig. 17. An example of HSV output stego image (b) and original cover image (a) for human visual observation test.

4. Conclusions

In this chapter, the histogram−based steganography, classical and contemporary approaches and their advantages are presented. These advantages can be summarized as follows; relatively high data hiding capacity, imperceptibility on stego image and its histogram, high statistical quality for stego images not only in terms of PSNR but also VIF, UQI and M−SSIM, applicable to very small digital images and robust against geometrical attacks.

A contemporary histogram−based data hiding approach (HSV), its applications and evaluations are also given in detail. Application and experimental results of the HSV for well−known test images Lena, Baboon and Peppers clearly show that the visual differences between the original and the corresponding stego images with random hidden data cannot be detected by the human visual system. As a concluding final remark, the original image histogram is almost same as the resulting stego image histogram produced by using the HSV; thus, neither visual nor statistical comparison of them (assuming that the attacker has the original image and its histogram although this is an extremely difficult case) enables perception of any data hiding application being realized. All of these results confirm the success of histogram−based approaches to data hiding over classical methods.

5. References

Akar, F. & Varol, H.S. (2004). A New RGB Weighted Encoding Technique for Efficient Information Hiding in Images. *Journal of Naval Science and Engineering*, vol. 2, July 2004, pp. 21−36.

Akar, F. (2005). Implementation of Information Security Based on Steganography and Cryptology. *Marmara University, PhD. Thesis*, 2005.

Celik, M. U.; Sharma, G.; Tekalp, A. M. & Saber, E. (2002). Reversible Data Hiding, *IEEE International Conference of Image Processing*, vol. 2, pp. 157−160, 2002.

Cetin, O. & Ozcerit, A. T. (2009). A New Steganography Algorithm Based on Color Histograms for Data Embedding into Raw Video Streams. *Computers & Security*, 2009, pp. 670–682.

Chang, C. C.; Tai, W. L. & Chen, K. N. (2008). Lossless Data Hiding Based on Histogram Modification for Image Authentication, *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, pp. 506−511, 2008.

Chrysochos, E.; Fotopoulos, V.; Skodras, A. & Xenos, M. (2007). Reversible Image Watermarking Based on Histogram Modification, *11th Conference on Informatics with International Participation*, vol. B, pp. 93–104, Greece, 2007.

Cox, I. J. & Miller, M. L. (2002). The First 50 Years of Electronic Watermarking. *Journal of Applied Signal Processing*, vol. 16, no. 4, 2002, pp. 126–132.

Fallahpour, M. & Sedaaghi, M. H. (2007). High Capacity Lossless Data Hiding Based on Histogram Modification. *IEICE Electronics Express*, vol. 4, no. 7, 2007, pp. 205−210.

Fridrich, J.; Du, R. & Long, M. (2000). Steganalysis of LSB Encoding in Color Images, *IEEE Int. Conf. on Multimedia and Expo (ICME)*, vol. 3, pp. 1279−1282, 2000.

Krishna, S. L. V.; Rahim, B. A.; Shaik, F. & Rajan, K. S. (2010). Lossless Embedding Using Pixel Differences and Histogram Shifting Technique, *IEEE Recent Advances in Space Technology Services and Climate Change (RSTSCC)*, pp. 213−216, 2010.

Lee, S. K.; Suh, Y. H. & Ho, Y. S. (2006). Reversible Image Authentication Based on Watermarking, *IEEE International Conference on Multimedia and Expo*, Canada, pp. 1321−1324, 2006.

Lin, C. Y.; Chang, C. C. & Wang, Y. Z. (2008). Reversible Steganographic Method with High Payload for JPEG Images, *IEICE Transactions on Information and Systems*, vol. E91.D, no. 3, 2008, pp. 836−845.

Netravali A. N. & Haskell, B. G. (1995). *Digital Pictures: Representation, Compression and Standards*, Plenum Press, New York, 1995.

Ni, Z.; Shi, Y. Q.; Ansari, N.; Su, W.; Sun, Q. & Lin, X. (2004). Robust Lossless Image Data Hiding, *IEEE Int. Conference on Multimedia and Expo (ICME)*, pp. 2199–2202, 2004.

Ni, Z.; Shi, Y. Q.; Ansari, N. & Su, W. (2006). Reversible Data Hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, Mar. 2006, pp. 354−362.

Ni, Z.; Shi, Y. Q.; Ansari, N.; Su, W., Sun, Q. & Lin, X. (2008). Robust Lossless Image Data Hiding Designed for Semi−Fragile Image Authentication. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 4, 2008, pp. 497–509.

OutGuess Steganography Detection Tool: (http://www.outguess.org/detection.php, accessed August 9, 2011.

Papapanagiotou, K.; Kelliniz, E.; Marias, G. F. & Georgiadis, P. (2005). Alternatives for Multimedia Messaging System Steganography, *Lecture Notes in Computer Science, Computational Intelligence and Security*, vol. 3802, pp. 589–596, 2005.

Rabbani, M. & Jones, P. W. (1991). *Digital Image Compression Techniques*, SPIE Optical Engineering Press, Washington, 1991.

Sahin, A.; Bulus, E. & Sakalli, M. T. (2006). LSB Data Hiding on 24 Bits RGB Images. *Trakya University Journal of Science*, 2006, pp. 17–22.

Sahin, A.; Bulus, E.; Sakalli, M. T. & Bulus, H. N. (2007). The Grasp of the Hidden Information on Images With the RQP Steganalysis Method", *IXth Akademik Bilisim Conferences*, pp. 83−87, 2007.

Sencar, H. T.; Ramkumar, M. & Akansu, A. N. (2004). *Data Hiding Fundamentals and Applications*, Elsevier Academic Press, New York, 2004.

Sheikh, H. D. & Bovik, A. C. (2006). Image Information and Visual Quality. *IEEE Transactions on Image Processing*, vol. 15, 2006, pp. 430–444.

Teng, C. Y.; Shiau, Y. H. & Chen, C. C. (2010). A Data Hiding Algorithm Based on Histogram Re−quantization, *IEEE 5th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*, pp. 1088−1091, 2010.

Thanuja, T. C.; Nagaraj, R. & Kumari, M. U. (2008). Reversible Data Hiding Using Increased Peak Histogram, *IEEE Proceedings of International Workshop on Data Mining and Artificial Intelligence (DMAI' 08)*, pp.44−47, 2008.

Wang, Z. & Bovik, A.C. (2002). A Universal Image Quality Index. *IEEE Signal Processing Letters*, vol. 9, 2002, pp. 81–84.

Wang, Z.; Bovik, A. C.; Sheikh, H.D. & Simoncelli, E.P. (2004). Image Quality Assessment: From Error Visibility To Structural Similarity. *IEEE Transactions on Image Processing*, vol. 13, 2004, pp. 600–612.

Watters, P. A.; Martin, F. & Stripf, H. S. (2005). Visual Steganalysis of LSB−Encoded Natural Images, *Proc. of the IEEE 3rd International Conference on Information Technology and Applications (ICITA'05)*, vol. 1, pp. 746−751, 2005.

Yalman, Y. & Erturk, I. (2009). A New Histogram Modification Based Robust Image Data Hiding Technique, *IEEE 24th Int. Symposium on Computer and Information Sciences (ISCIS'09)*, pp.39–43, 2009.

Yalman, Y. (2010). Design and Implementation of A Steganography Method Based on Histogram Modification for Digital Images. *PhD. Thesis*, Kocaeli University, 2010.

IntechOpen