Batman Üniversitesi Yaşam Bilimleri Dergisi
Batman University Journal of Life Sciences
E-ISSN: 2459-0614

Batman Üniversitesi Yaşam Bilimleri Dergisi 12 (2), 2022, 105-118

Implementation of a Low-Cost Tamper Detection Method for RGB Images

Hüseyin Bilal Macit

Burdur Mehmet Akif Ersoy University, Bucak ZTYO, Department of Information Systems and Technologies, Burdur, Türkiye

Doi: 10.55024/buyasambid.1093220

ARTICLE INFO

Article history:
Received: 25.03.2022
Received in revised form
Accepted: 05.09.2022
Available online: 30.12.2022

Key words:
Image Tampering, Parity Code, LSB.

* Hüseyin Bilal Macit.
hbmacit@mehmetakif.edu.tr
Orcid: 0000-0002-5325-5416

ABSTRACT

Thanks to developing technology, a large number of images are transferred on the internet. The vast majority of these images are uploaded to social media platforms and have a low level of privacy. Due to the ease of access to digital images and the fact that the image can be easily tampered with, manipulated images can be used in various forgery and fraud methods. Various algorithms have been proposed in the literature to determine whether the digital image is original or tampered. In this study, the even parity bit method, which is a very simple and low-complex error detection mechanism, is applied to the layer data of 3-layer color images with a steganographic approach and it is aimed to detect tampering in digital images. The proposed method has been applied to a group of test images that are frequently used in image processing applications in the literature. The method produced more successful results than other methods based on similarity scores in the preprocessing stage of active image tampering detection. In the image tampering detection stage, low success is achieved in content-based attacks, but high success is achieved in geometric-based attacks.

RGB Görüntüler İçin Düşük Maliyetli Bir Görüntü Tahrif Tespiti Yöntemi Uygulaması

Hüseyin Bilal Macit

Burdur Mehmet Akif Ersoy Üniversitesi, Bucak ZTYO, Bilişim Sistemleri ve Teknolojileri Bölümü, Burdur, Türkiye

Doi: 10.55024/buyasambid.1093220

Makale Bilgisi

Makale geçmişi:
İlk gönderim tarihi: 25.03.2022
Düzeltme tarihi
Kabul tarihi: 05.09.2022
Yayın tarihi: 30.12.2022

Anahtar Kelimeler:
Görüntü Tahrifi, Eşlik Kodu, LSB

* Hüseyin Bilal Macit
hbmacit@mehmetakif.edu.tr
Orcid: 0000-0002-5325-5416

Özet

Gelişen teknoloji sayesinde internet üzerinde çok sayıda görüntü transfer edilebilmektedir. Bu görüntülerin büyük çoğunluğu sosyal medya platformlarına yükleniyor ve düşük düzeyde mahremiyete sahip. Dijital görüntülere erişimin kolay olması ve görüntünün kolayca tahrif edilebilmesi nedeniyle, manipüle edilmiş görüntüler çeşitli sahtecilik ve dolandırıcılık yöntemlerinde kullanılabilmektedir. Sayısal görüntünün orijinal mi yoksa tahrif edilmiş mi olduğunu belirlemek için literatürde çeşitli algoritmalar önerilmiştir. Bu çalışmada, basit ve düşük işlem karmaşıklığına sahip bir hata tespit mekanizması olan çift eşlik biti yöntemi, 3 katmanlı renkli görüntülerin katman verilerine steganografik bir yaklaşımla uygulanmakta ve sayısal görüntülerde manipülasyonun tespit edilmesi amaçlanmaktadır. Önerilen yöntem literatürde görüntü işleme uygulamalarında sıkça kullanılan bir grup test görüntüsünde uygulanmıştır. Yöntem aktif görüntü tahrif tespiti ön işlem aşamasında benzerlik skorları baz alındığında diğer yöntemlerden daha başarılı sonuçlar üretmiştir. Görüntü tahrifi tespiti aşamasında içerik tabanlı saldırılarda düşük başarı elde edilmiş ancak geometrik tabanlı saldırılarda yüksek başarı elde edilmiştir.

1. INTRODUCTION

Today, digital images have completely replaced traditional photographs in every aspect of life (Mishra and Adhikary, 2013). Thanks to the latest developments in internet and storage technologies, digital images are easily distributed over the internet (Vaishnavi and Subashini, 2015). These digital images include unimportant images such as traffic signs, circuit diagrams, and product photos, less important images such as photographs, medical images, and very important images such as signed documents, identity documents, cashier checks, and promissory notes. Changing or manipulating a digital image is much easier compared to a traditional image. In traditional photography, difficult processes were required to modify a photograph such as retouching with ink (Mishra and Adhikary, 2013). Digital images can be changed with free or inexpensive software such as Photoshop, Corel Paint Shop, Photoscape, PhotoPlus, GIMP, and Pixelmator. This makes it easier for malicious manipulations on digital images. Malicious manipulation of an image is called an "attack". Image attacks are divided into three classes; geometric-based (rotation, zooming, cropping, shearing, etc.), enhancement-based (histogram equalization, color modification, contrast adjustment, filtering, etc.) and content-based (cut, copy, move, paste, etc.) (Shashidhar and Ramesh, 2017). Malicious attackers usually try to change the meaningful information of an image to change its meaning (Wang and Chen,

2007). It is now quite difficult to distinguish whether an image is a real camera output or a manipulated image.

1.1. Literature Overview

Wang and Chen (2007) present a novel color image watermarking scheme for both tamper detection and tampered image recovery. They embed watermarks consisting of the authentication data and the recovery data into image blocks. They successfully recovered test images with acceptable visual quality. Vaishnavi and Subashini (2015) proposed a novel method of fragile watermarking to detect image tampers. They implemented by an edge image and chaotic Arnold map. The edge image is obtained from the watermark image using the Canny edge detection operator. The method they proposed method efficiently localizes the tampered regions. Taha et al (2018) presented a blind image tamper detection and self-recovery method using the Lifting Scheme which is characterized by simplicity and integer-based calculations and LSB modification. Their method performs well in terms of detection and recovery for different types of tampering as removing and cloning. Rawat and Raman (2011) proposed a passive tamper detection method that employs chaotic maps to withstand counterfeiting attacks. They demonstrated that the proposed scheme achieves superior tamper detection and localization accuracy under different attacks such as copy-and-paste attacks and collage attacks. Dirik and Nemon (2009) introduced tamper detection techniques based on artifacts created by Color Filter Array which are based on computing a single feature and a simple threshold-based classifier. They tested the approach over authentic, tampered, and computer-generated images. Their method results in reasonably low error rates. Golea (2019) proposed a region of interest-based fragile watermarking scheme for medical image tamper detection. The CRC code is based on a standard polynomial generator CRC-32 with more particular mathematical properties and is performed on each packet to generate a watermark to be inserted in the spatial domain. To check tampering, they extracted the watermark. The results of the experiments show the validity of the proposed approach in terms of imperceptibility and efficiency to detect reliable and strong attacks.

1.2. Image Tampering

The unauthorized modification of the meaningful part of an image is called "tampering". Analysing and detecting whether digital images have been tampered or not is an important area of research. Image tampering attacks can be classified as follows:

Cloning-Based Attack: To hide a region on the image, a part of the image is copied and pasted into the region to be hidden. It is easily detectable by the naked eye when performed by a non-expert attacker. Figure 1 shows an example of a cloning-based attack.

Figure 1. Cloning-based attack (Shashidhar and Ramesh, 2017)

Image Splicing Attack: It is the process of obtaining a new image by combining some regions of two or more images. Figure 2 shows an example of an image splicing attack.

Figure 2. Image splicing attack (Qureshi and Deriche, 2015)

Copy-move attack: It is one of the most common tampering attacks. It is performed by copying a region of the image and pasting it into another region. It is very difficult to detect because the source and target images are the same. An original image and attacked versions are shown in figure 3.

Figure 3. Copy-move attack (Yeap et al., 2018)

Image Retouching: This method is generally used to improve image properties such as brightness, contrast, or images in the image, and is also rarely used for image tampering. Figure 4 shows an example of image retouching.

Figure 4. Image retouching attack (Alamro and Nooraini, 2017)

Image resampling: It is the type of attack where the size or resolution of the input image is changed. For example, reducing an image with a pixel density of 300DPI to a pixel density of 60DPI. Figure 5 shows an example of hybrid attack of image resampling and cropping.

Figure 5. Image resampling attack

Various methods can be used to detect a tampering attack applied to an image. These methods are examined in two classes as active and passive methods (Deshpande and Kanikar, 2012). Figure 6 shows the classification of image tamper detection techniques.

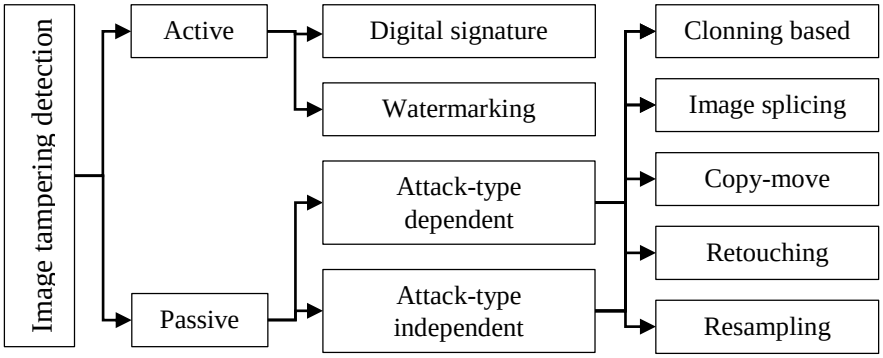


Figure 6. Classification of tamper detection methods

Active techniques require pre-processing such as adding a digital signature or watermark (Gulivindala and Rao, 2013). In active techniques, image tampering is detected by looking at the current state of the digital signature or watermark which is previously hidden in the image. Active techniques are considered the most efficient tamper detection techniques. Passive techniques are called blind techniques because they do not need the original image beforehand for detection operations. These techniques make decisions using some semantic, statistical data and threshold values extracted from the image. Active methods usually provide accurate results. Passive methods consist of complex algorithms that are difficult to implement. Passive methods are frequently used in forensic cases (Granty et al., 2010). In passive techniques, if the attack type applied to the image is predicted, the probability of tamper detection is higher (Chennamma and Madhushree, 2022).

The performance of the tampering algorithm can be evaluated by following criteria (Vaishnavi and Subashini, 2015;Taha et al., 2018).

1. **Tamper detection:** The algorithm should report whether the image is tampered or not.

2. **Imperceptibility:** In active methods, the watermark or digital signature placed on the image should not be noticed by the human vision system (HVS).

3. **Tampered region detection:** The algorithm should detect and report the tampered region in the image.

4. **Self-recovery:** The method should recover the tampered area.

5. **Blind detection:** The original image is not required for tamper detection.

6. **Efficiency:** The processing complexity of the algorithm should be minimal.

7. **Security:** Even if the watermark or digital signature hidden on the image can be detected in active methods, it should not be easily destroyed.

8. Sensitivity: In active methods, the pre-embedded watermark or digital signature should be strong to simple image processing processes but it should be vulnerable to malicious tampering.

1.3. Error Detection and Correction

Techniques that ensure data security in unreliable storage or transmission environments are called error correction techniques. Unsafe transmission and storage environments are sensitive to noise. Partial changes in data may occur during the reading or transmission of data. The primary purpose of error correction techniques is to detect error. However, some techniques can also make partial corrections to the error (Zulfira et al., 2021). Error correction techniques use coding algorithms (Senekane et al., 2021). The most commonly used ones are Single Parity Code (SPC), 2D Parity Code, Hamming code, and CRC.

SPC is a simple form of error detection coding. It is usually applied to 8-bit octets (bytes), which are the smallest units of the storage or communication protocol. Occasionally it can be applied to longer data strings. SPC checks whether the total number of "1" bits in the data string is odd or even, and adds an SPC bit to the end of the string. There are two variants of SPC; odd parity and even parity. In the case of even parity, the parity bit is added as 1 if the number of "1" bits is odd, and 0 if it is even (Figure 7). Thus, the sum of the "1" bits, including the parity bit, is an even number. In the case of odd parity, the parity bit is added as 0 if the number of "1" bits is odd, and as 1 if the number of even parity is odd. Thus, the sum of the "1" bits, including the parity bit, is an odd number.

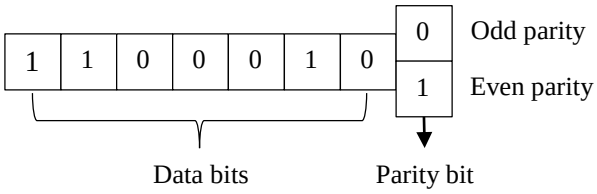


Figure 7. Even and odd SPC

In electronic systems, a system that creates and performs parity checks can be designed using XOR and NOT logic gates. Table 1 shows the basic inputs and outputs of a simple XOR gate. Table 2 shows the inversion of a NOT gate. A parity bit is calculated by applying XOR to all bits in order as shown in Figure 8.

Table 1. XOR Gate

1.Input	2.Input	Output
0	0	0
0	1	1
1	0	1
1	1	0

Table 2. NOT Gate

Input	Output
0	1
1	0

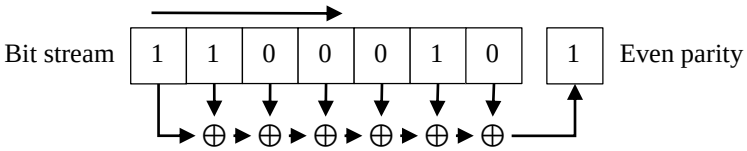


Figure 8. Even SPC with XOR Gates

2. METHOD

The digital image is represented by an array of N rows and M columns (Figure 9). Each cell of an image array is called a pixel. In the simplest case, each pixel is represented by a bit (1 or 0). This image is called a binary image (Şahin et al., 2013).

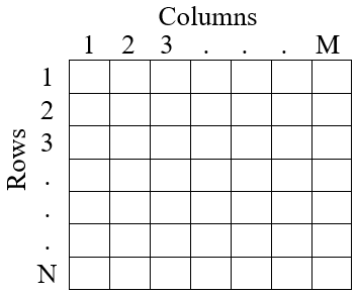


Figure 9. Representation of a digital image

Images in which each pixel is represented by 8 bits (1 byte) are called monochrome images. In a monochrome image, each pixel takes an integer value between 0-255. Values between 0-255 are shades of a single color. Color images are represented in RGB space and consist of 3 layers; Red, Green and Blue (Figure 10). Each layer is represented by 8 bits (a total of 24 bits) holding the tone information of its color.

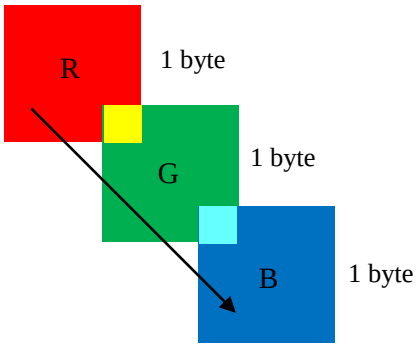


Figure 10. Representation of an RGB pixel (Macit and Koyun, 2019)

Let a digital image I represented by an array of M rows and N columns. Therefore, a digital image contains MxN pixels.

$$I = \{p_{ij} | 1 \leq i < M, 1 \leq j < N\}$$

Here; $p_{i,j}$ represents each element of the I matrix. 3 matrices should be created for each color plane of an RGB image (Macit and Koyun, 2019).

$$R = \{r_{ij} | 1 \leq i < M, 1 \leq j < N\}, r_{ij} \in \{0,1,2,\dots,255\}$$

$$G = \{g_{ij} | 1 \leq i < M, 1 \leq j < N\}, g_{ij} \in \{0,1,2,\dots,255\}$$

$$B = \{b_{ij} | 1 \leq i < M, 1 \leq j < N\}, b_{ij} \in \{0,1,2,\dots,255\}$$

A pixel in an RGB image can be expressed as in table 3;

Table 3. Binary representation of an RGB pixel

			MSB							**LSB**
Color	**Layer**	**Decimal**	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1
	R	150	1	0	0	1	0	1	1	0
	G	120	0	1	1	1	1	0	0	0
	B	170	1	0	1	0	1	0	1	0

The leftmost bit of the color layer octet has the largest numerical value and called most significant bit (MSB), and the rightmost bit has the smallest numerical value and called least significant bit (LSB). Any change in the MSB of any of the R, G, and B layers of a pixel causes a color change which is easily detectable by HVS. However, even if all LSB bits are changed, it is not possible for the HVS to detect the color change of the pixel as shown in table 4. For this reason, LSBs can be used to hide some data. Hiding data into the LSBs is suitable for use in tampering detection, as it is susceptible to even a simple attack (Stoilov et al., 2021).

Table 4. Impact of MSB versus LSB on RGB layers

Original color	**MSBs changed in RGB**	**LSBs changed in RGB**

In this study; a fast and effective active tamper-detection method is proposed and implemented, in which the even SPC of all color layers is calculated and placed to its LSB. f is the boolean function with n variables to calculate parity;

$$f: \{0,1\}^n \rightarrow \{0,1\}$$

So; the sum of "1" bits in $x \in \{0,1\}^n$ vector is calculated as $f(x) = 1$. If the function f is represented with $\oplus$ (XOR);

$$f(x) = x_1 \oplus x_2 \oplus \dots \oplus x_n$$

Let the R, G and B layers of $p_{i,j}$ pixel which is at i. row and j. column of image I are respectively $r_{i,j}$, $g_{i,j}$ and $b_{i,j}$.

$r_{i,j} \in \{0,1,2,\dots,255\}$, $r_{i,j} = x_8x_7x_6x_5x_4x_3x_2x_1$

$g_{i,j} \in \{0,1,2,\dots,255\}$, $g_{i,j} = x_8x_7x_6x_5x_4x_3x_2x_1$

$b_{i,j} \in \{0,1,2,\dots,255\}$, $b_{i,j} = x_8x_7x_6x_5x_4x_3x_2x_1$

All the x_8s for the $p_{i,j}$ are MSBs and the x_1s for the $p_{i,j}$ are LSBs. Before the original image is distributed, the parity bits are embedded into the LSBs in the pre-processing stage;

$r_{i,j}(x_1) = r_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

$g_{i,j}(x_1) = g_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

$b_{i,j}(x_1) = b_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

If the image has been tampered with in an insecure environment, it is now easy to detect. The same method for this process is repeated for all pixels. If the 3 conditions below are met at the same time, $p_{i,j}$ is not tampered. Otherwise, this pixel is marked as tampered.

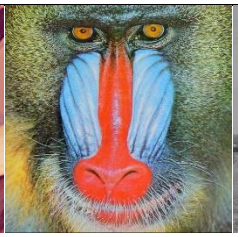
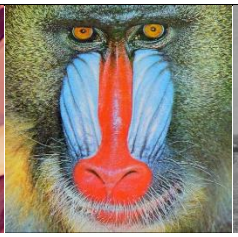
Condition 1: $r_{i,j}(x_1) = r_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

Condition 2: $g_{i,j}(x_1) = g_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

Condition 3: $b_{i,j}(x_1) = b_{i,j}(x_8 \oplus x_7 \oplus x_6 \oplus x_5 \oplus x_4 \oplus x_3 \oplus x_2)$

The proposed method is tested on three different images, which are often used in image processing articles. Table 5 shows original and pre-processed images.

Table 5. Original vs. pre-processed images

	Peppers	Lena	Baboon	Cameraman	Boat
Original					
Pre-processed					

One of the performance criteria of the tampering algorithm is imperceptibility. A pre-processed image should be indistinguishable from the original by HVS. There are many methods to measure the similarity between two images. In this study, Peak signal-to-noise ratio (PSNR) and structural

similarity index (SSIM) methods were used to mathematically express how far the original image from the pre-processed image.

PSNR examines the noise between two different images using Mean Square Error (MSE). Let I_p is the pre-processed image and x_i and y_i are the samples of them;

$$MSE(I, I_p) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2$$

The equation below shows the calculation of PSNR. Here; L is the dynamic range of the allowed image pixel.

$$PSNR = 10log_{10}\frac{L^2}{MSE}$$

SSIM measures the similarity between two images. SSIM is much closer to HVS. SSIM first calculates three parameters; luminosity $l(I, I_w)$, degradation $c(I, I_w)$, and degradation $s(I, I_w)$.

$$l(I, I_w) = \left(\frac{2\mu_I\mu_{I_w} + k_1}{{\mu_I}^2 + {\mu_{I_w}}^2 + k_1}\right)$$

$$c(I, I_w) = \left(\frac{2\sigma_I\sigma_{I_w} + k_2}{{\sigma_I}^2 + {\sigma_{I_w}}^2 + k_2}\right)$$

$$s(I, I_w) = \left(\frac{2\sigma_{II_w} + k_3}{\sigma_I + \sigma_{I_w} + k_3}\right)$$

SSIM is calculated in the equation below after calculating l,c and s.

$$SSIM(I, I_w) = l(I, I_w)^{\alpha}.c(I, I_w)^{\beta}.s(I, I_w)^{\gamma}$$

3. RESULTS

Three test images are chosen as test images which are used as test images in almost all image processing studies because of their specific texture and other properties. Similarity results between original and pre-processed test images are shown in table 6. As it is shown, SSIM values are close to 1. This means that generated watermark for active tamper detection is greatly imperceptible by HVS. Based on PSNR measurements, the standard LSB method usually produces values of 40 dB or more. The LSB method is therefore a popular method of hiding data in the image. If the PSNR value obtained with the LSB technique is over 50dB, the method is quite successful (Setiadi, 2021).

Table 6. Similarity results of test images

	PSNR	**SSIM**
Peppers	51.2026	0.9998
Lena	51.148	0.9998
Baboon	51.1607	0.9996
Cameraman	51.0919	0.9955
Boat	51.1621	0.9964

The proposed method is compared with other active tamper detection methods at the watermarking stage. Accordingly, the maximum and minimum values for the SSIM and PSNR results obtained in the test images used in other methods are given in Table 7.

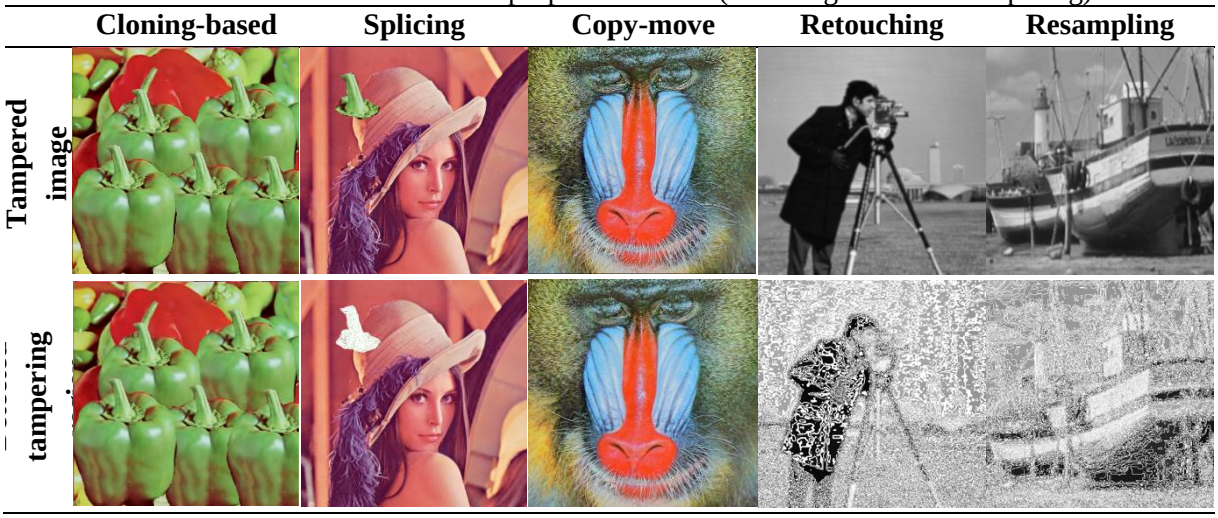
Table 7. Comparison of similarity results

Method	Max SSIM	Min SSIM	Max PSNR	Min PSNR
The proposed method	0.9998	0.9955	51.2026	51.0919
Taha et al, 2018	0.9914	0.9649	36.0161	28.3141
Vaishnavi and Subashini, 2015	NaN	NaN	51.1483	50.6246
Wang and Chen, 2007	NaN	NaN	44.56	38.86
Rawat and Raman, 2011	NaN	NaN	51.1552	50.7261
Golea, 2019	0.9970	0.9815	57.8021	48.0818

The results figured in Table 7 show satisfactory imperceptibility results of the proposed method. In every case, the PSNR and SSIM values are greater than other proposed methods.

The proposed method is implemented with MATLAB software. Tampering attacks have been performed on pre-processed images with various image editing software, and the results are shown in table 8.

Table 8. Test results of the proposed method (white regions show tampering)



It is clearly seen that the proposed method is successful in tamper detection in geometric and enhancement-based attacks. However, it is not successful in content-based attacks. Also, the method is able to detect and show the tampered region of the image. The PSNR and SSIM values, in which the original and pre-processed images are mathematically compared, show that the proposed method has the invisibility expected from an active method.

Due to its nature, the SPC method can detect a 1-bit change very quickly. However, in the event of possible tampering, there is also the possibility that more than one bit of a color layer will change at the same time.

Let P is the probability of tamper detection of image I and P_r, , and P_b are the probabilities of tamper detection in R, G, and B layers respectively. $P_r = \frac{1}{2}$, $P_g = \frac{1}{2}$ and $P_b = \frac{1}{2}$. Given that, the probability of failing to detect tampering in a single pixel;

$$P' = P_r . P_g . P_b = \frac{1}{2} . \frac{1}{2} . \frac{1}{2} = \frac{1}{8}$$

In other words, it can be expressed as 12.5%. Malicious tampering with images usually takes place on consecutive pixels. In this case, the probability of failing to detect tampering on n consecutive pixels;

$$P'_n = (\frac{1}{8})^n$$

4. CONCLUSIONS

In this study, we proposed an active tamper detection method. The proposed method offers low processing complexity and can be applied on high-resolution images even using simple processors. In active tamper detection applications, it is expected that the image in the unsafe environment will not be understood as being preprocessed. Therefore, the image is expected to be as close to the original as possible. We used SSIM and PSNR measures to calculate the similarity of the processed image to the original image and achieved greater scores than other methods in the literature.

In active tamper detection methods, no criteria have been proposed in the literature for the performance measure of after-attack tamper detection or image recovery. The only performance criteria after attacks is HVS. We clearly see that the proposed method is successful in tamper detection in geometric and enhancement-based attacks. However, it is not successful in content-based attacks. The proposed method has a very low probability of failing to detect tampering region. In addition, as the number of tampered pixels increases, this probability decreases.

REFERENCES

Alamro, L. and Nooraini, Y. (2017). Copy-move forgery detection using integrated DWT and SURF, Journal of Telecommunication, Electronic and Computer Engineering (JTEC), Vol. 9, pp. 67-71.

Chennamma, H. R. and Madhushree, B. (2022). A comprehensive survey on image authentication for tamper detection with localization, Multimedia Tools and Applications 2002, DOI: 10.1007/s11042-022-13312-1.

Deshpande, P. and Kanikar, P. (2012). Pixel Based Image Forgery Detection Techniques, International Journal of Engineering Research and Applications, Vol. (2)3, pp.539-543.

Dirik, A. E. and Memon, N. (2009). Image tamper detection based on demosaicing artifacts, 16th IEEE International Conference on Image Processing (ICIP), pp. 1497-1500, DOI: 10.1109/ICIP.2009.5414611.

Golea, N.E.H. (2019). ROI-based fragile watermarking for medical image tamper detection, International Journal of High Performance Computing and Networking, Vol. 13, No. 2, pp. 199-209, DOI: 10.1504/IJHPCN.2019.097508

Granty, R. E. J., Aditya, T. S. and Madhu, Shankar, S. (2010). Survey on Passive Methods of Image Tampering Detection, Proceedings of the International Conference on Communication and Computational Intelligence, pp. 431-436, T.N.,India.

Gulivindala, S. and Rao, C.S. (2013). Tampering Detection Algorithms: A Comparative Study, International Journal of Engineering Research and Development, e-ISSN: 2278-067X, p-ISSN: 2278-800X, Vol. (7)5, pp. 82-86.

Macit, H.B. and Koyun, A. (2019). Tamper Detection and Recovery on RGB Images, International Conference on Artificial Intelligence and Applied Mathematics in Engineering, pp. 972-981, DOI: 10.1007/978-3-030-36178-5_86.

Mishra, M. and Adhikary, M.C. (2013). Digital Image Tamper Detection Techniques - A Comprehensive Study, International Journal of Computer Science and Business Informatics, ISSN: 1694-2108, Vol. (2)1.

Qureshi, M.A. and Deriche, M. (2015). A bibliography of pixel-based blind image forgery detection techniques, Signal Processing: Image Communication, Vol. 39, pp. 46-74, DOI:10.1016/j.image.2015.08.008.

Rawat, S. and Raman, B. (2011). A chaotic system based fragile watermarking scheme for image tamper detection, International Journal of Electronics and Communications, Vol. 65, pp. 840-847

Senekane, M., Mafu, M., Maseli, M. and Taleme, B.M. (2021). A quantum algorithm for single parity check code, IEEE 2021 Africon, DOI: 10.1109/AFRICON51333.2021.9570857.

Setiadi, M. S. (2021). PSNR vs SSIM: imperceptibility quality assessment for image steganography, Multimedia Tools and Applications, Vol. 80, pp. 8423-8444, DOI:10.1007/s11042-020-10035-z

Shashidhar, T.M. and Ramesh, K.B. (2017). Reviewing the Effectivity Factor in Existing Techniques of Image Forensics, International Journal of Electrical and Computer Engineering, Vol. (7)6, pp.3558-3569, ISSN: 2088-8708, DOI: 10.11591/ijece.v7i6.pp3558-3569.

Stoilov, P.S., Hristov, G. and Zahariev, P. (2021). Analysis Of The Least Significant Bit Subtitution Algorithm For Image Stenography, Proceedings Of University Of Ruse, Vol. 60, Book. 3.2, pp. 207-213.

Şahin, A., Buluş, E. and Sakallı, M.T. (2006). 24-Bit Renkli Resimler Üzerinde En Önemsiz Bite Ekleme Yöntemini Kullanarak Bilgi Gizleme, Trakya Univ J Sci, Vol. 7(1), pp. 17-22, ISSN 1305-6468.

Taha, B.T., Ngadiran, R., Ehkan, P. and Sultan, M.T. (2018). Image Tamper Detection and Recovery Using Lifting Scheme Based Fragile Watermarking", Journal of Theoretical and Applied Information Technology, Vol. (96)8, ISSN: 1992-8645.

Wang, M.S. and Chen, W.C. (2007). A majority-voting based watermarking scheme for color image tamper detection and recovery, Computer Standards & Interfaces, Vol. 29, pp. 561–570.

Vaishnavi, D. and Subashini, T.S. (2015). Image Tamper Detection based on Edge Image and Chaotic Arnold Map, Indian Journal of Science and Technology, Vol. (8)6, pp. 548–555.

Yeap, Y.Y., Sheikh, U.U. and Rahman, A. (2018). Image forensic for digital image copy move forgery detection, IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), DOI:10.1109/CSPA.2018.8368719.

Zulfira, F., Nuha, H.H., Sudiharto, D.W. and Utomo, R.G. (2021). Modified Bit Parity Technique for Error Detection of 8 Bit Data, Proceedings of 9'th International Conference on Information and Communication Technology, pp. 517-521.