International Science and Technology Journal
المجلة الدولية للعلوم والتقنية
Volume 37 العدد
Part 1 المجلد
المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ

http://www.doi.org/10.62341/hawe0728

| | | |
|---|---|---|
| **Received** | **2025/07/02** | **تم استلام الورقة العلمية في** |
| **Accepted** | **2025/07/26** | **تم قبول الورقة العلمية في** |
| **Published** | **2025/07/28** | **تم نشر الورقة العلمية في** |

# Extension of ATD Steganographic Method to Work with Color Images

**Hajer A.Alaswed**
College of Industrial Technology - Misurata, Libya
Hajer87ed@gmail.com

## Abstract

This paper proposed to extend the ATD steganographic method to work with color images and conduct an experimental study of the modified algorithm. In the original ATD algorithm, which works on grayscale images, The main idea is embedding the secret message as the ternary number and in each pixel of cover image is embedding two ternary numbers.

For color images. we propose embedding four ternary numbers within each pixel of the cover image. As is well known, color images are composed of three primary color channels: red, green, and blue. The final color is generated by blending specific intensities of these three channels. Accordingly, three embedding combinations (112,121,211) were proposed to determine how the four ternary numbers are distributed among the red, green, and blue channels of each pixel. Each such combination corresponds to approximately 8 BPP for example, using the combination 211 means embedding two ternary numbers in the red channel (R), and one ternary number in each of the green (G) and blue (B) channels.

By using the Matlab program, we compared the results on eight color images selected as cover images to evaluate the performance of the proposed algorithm. Image quality metrics, such as Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE), were used to evaluate the algorithm's performance according to our analysis, the value of PSNR is not affected by the scheme but depends on image; PSNR is in range from 34.50 dB to 37.57 dB in same embedding capacity 8 BPP. Moreover, SNR values across the different embedding combinations, suggesting comparable performance among them.

**Keywords:** Steganography, ATD, Peak Signal to Noise Ratio (PSNR), Embedding Capacity, Bit Per Pixel (BPP).

International Science and Technology Journal
المجلة الدولية للعلوم والتقنية
Volume 37 العدد
Part 1 المجلد
المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ
http://www.doi.org/10.62341/hawe0728

# تطوير خوارزمية ATD لتعمل على الصور الملونة

**هاجر أحمد الأسود**

قسم الهندسة الالكترونية – كلية التقنية الصناعية – مصراته – ليبيا

**الملخص**

في هذه الورقة، تم تطوير خوارزمية ATD لتعمل مع الصور الملوّنة، حيث أُجريت دراسة تحليلية وتجريبية على الخوارزمية في النسخة الأساسية من خوارزمية ATD، التي تُطبَّق على الصور الرمادية، تتمثل الفكرة الرئيسية في إخفاء الرسالة السرية بعد تحويلها إلى النظام الثلاثي(Ternary Number) ، بحيث يُخفى رقمين ثلاثيين من الرسالة السرية في كل بكسل من بكسلات صورة الغلاف. بالنسبة للصور الملوّنة، فقد اقترحنا إخفاء أربعة أرقام ثلاثية داخل كل بكسل من بكسلات صورة الغلاف. وكما هو معلوم، تتكوّن الصور الملوّنة من ثلاث قنوات لونية رئيسية: الأحمر(Red) ، والأخضر(Green) ، والأزرق(Blue) ، حيث يتكوّن اللون النهائي من دمج درجات محددة من هذه القنوات الثلاث. بناءً على ذلك، تم اقتراح ثلاث توليفات (112، 121، 211) لتحديد عدد الأرقام الثلاثية المُخفاة في كل قناة لونية من البكسل (الأحمر، الأخضر، الأزرق). كل واحدة من هذه التوليفات تكون بسعة إخفاء 8 بت لكل بكسل فعلى سبيل المثال، عند استخدام التوليفة (211)، فهذا يعني إخفاء رقمين ثلاثيين في القناة الحمراءRED، و رقم ثلاثي واحد في كل من القناتين الخضراء Green و الزرقاء Blue.

أجريت التجارب باستخدام برنامج MATLAB ماتلاب على 8 صور ملوّنة تم استخدامها كصور غلاف، لتقييم أداء الخوارزمية تم استخدام مقاييس جودة الصورة مثل PSNRوMSE من خلال مقارنة النتائج المتحصّل عليها، تبيّن أن قيمة PSNR لا تتأثر بالتوليفة المُستخدمة في عملية الإخفاء، بل تعتمد بشكل رئيسي على الصورة المُستخدمة كصورة غلاف؛ حيث كانت قيم PSNR بين 34.50 dB و 37.57 dB عند نفس سعة الإخفاء(8 BPP) أما بالنسبة لقيمة SNR، فقد أظهرت النتائج تقاربًا كبيرًا بين مختلف التوليفات المستخدمة.

**الكلمات المفتاحية:** إخفاء المعلومات، خوارزمية ATD، نسبة الإشارة إلى الضوضاء (PSNR)، سعة التضمين، بت لكل بكسل (BPP).

http://www.doi.org/10.62341/hawe0728

## A. Introduction

Steganography is the art of embedding information with in other forms of media such as text, images, videos, or audio in such a way that the presence of the hidden information is unlikely to be detected [1].

In steganography, there are two main categories of techniques: spatial domain and frequency domain. In the spatial domain, the actual pixel values of the cover image are directly modified to embed the secret information. In contrast, the frequency domain involves transforming the cover object into another domain, such as the Fast Fourier Transform (FFT), to obtain frequency coefficients. These coefficients are then manipulated to embed the hidden data [2].

Steganography can be classified into several types, including text steganography, audio steganography, video steganography, protocol steganography, and image steganography. In image steganography, the secret data is embedded into an image, which may be a color image, grayscale image, or binary image.

We consider here spatial domain methods. There are many steganographic schemes based on direct replacement like Least Significant Bit (LSB) [11] and Algorithm with Ternary Digits (ATD) [10] scheme; each it has own security and complexity. The main aim of each is to embed a large amount of secret data with minimal impact on the cover object, which means more bits per pixel (BPP) embedding capacity with good image quality.

In 2006, Chin, Wei-Liang, and Chia-Chen proposed a scheme for digitally compressed images based on Side Match Vector Quantization (SMVQ) [3]. In this method, the cover image is compressed using SMVQ, and a compressed image is generated. The SMVQ-compressed cover image is then divided into non-overlapping blocks, and the secret data is embedded into these blocks. This approach achieves a larger secret data capacity, better visual quality, and a higher compression rate compared to other methods based on SMVQ.

In 2007, Yoon, Chan, and Eun proposed a scheme for embedding a color or grayscale image into a true color image.[4] This scheme uses three different types of color secret images: a color image based on a 256-color palette, and a grayscale image. The secret image is converted to a binary representation, and the secret data is then protected by encrypting it using the DES algorithm. Each 8-bit

http://www.doi.org/10.62341/hawe0728

encrypted data is then divided into three bits: 3 bits, 2 bits, and 3 bits. This method achieves high image quality.

In 2015 Hemalatha, Dinesh and Renuka proposed a method for embedding audio in color image by using the wavelet transform [2]. In this method, the cover image is displayed in YCbCr format, and then the Cb, Cr, and occult audio components are transferred to the wavelet domain using the Intra-wavelet Transform (IWT). The approximate occult audio coefficient is embedded in the second and third bits of the high-frequency coefficients Cb and Cr. This method demonstrates high stego image quality.

In 2012 Taur, Lin, lee and Tao proposed a method for hiding data in DNA sequences based on table lookup substitution (TLSM) [5]. This method aims to improve the performance of a data embedding technique known as substitution. The base-t TLSM encrypts the secret message using the radix t to take full advantage of the substitution table. In the extended TLSM (ETLSM), the number of selectable substitution tables is increased by adding additional characters, significantly improving the security of the TLSM. This method performs well in terms of capacity and security.

In 2013 Kiruba and Karthikeyan proposed a method for detection of adaptive pixel pair matching in color images and grayscale images [6]. This technique is based on pixel pair matching (PPM) for data hiding. The basic idea of PPM is to use the values of pixel pairs as a reference configuration and search for coordinates in the neighboring group of this pair according to a specific message number. In this method, the maximum payload capacity is 1.161 bits per second. Therefore, this method achieves the best image quality with the least distortion.

In 2013 Arnab, Rajat and Sudipta proposed a method by using Sudoku puzzle for embedding a secret message in the color image [7]. In this method, the cover image is divided into equal-sized blocks, each of size 64. In each block, a letter of the secret message is embedded in every three pixels. Thus, this system is more robust with fewer computations.

In 2015 Jheng, Chen and Huang proposed a method for data hiding based on histogram medication over ternary computers [8]. They proposed two data hiding methods: ternary data hiding (TDH) and cryptographic ternary data hiding (C-TDH). In both methods, the secret data was ternary (in NAF format). Therefore, the TDH method achieves a higher signal-to-noise ratio (PSNR), while the C-

http://www.doi.org/10.62341/hawe0728

TDH method achieves a larger amount of secret data than the TDH method.

In 2016 Mehdi, Ainuddin and Anthony proposed a scheme for data embedding usingparity bit pixel value differencing (PBPVD) and improved rightmost digit replacement (IRMDR).[9]. In this method, the cover image is divided into non-overlapping pixel blocks, and then the parity bit value difference (PBPVD) and iRMDR value difference in each block are calculated by calculating the difference between the pixel values in the blocks. If the block difference value lies in the L_0 plane, iRMDR is applied; otherwise, PBPVD is applied.

**B. Algorithm with Ternary Digits (ATD)**

In this Algorithm, the secret data consists of digits {0, 1, 2}, represented as a ternary string. two ternary digits from the secret data are embedded into each pixel of the grayscale cover image [10].

- ATD Embedding Algorithm

**Inputs**: $S$ ternary secret message $S = \{s_k | 0 \leq k \leq |S|\ s_k \in \{0,1,2\}\}$, I is cover image it is size [M N] the number of rows is M, the number of columns is N, $I = \{v_{ij} | 0 \leq v_{ij} \leq 255\}$; the $i^{th}$, $j^{th}$ cover pixel is $v_{ij}$.

**Output**: Stego image, SI [M N].

**Step0:** $k = 0$

**Step1**: convert the pixel value $\boldsymbol{v_{ij}}$ to binary $B_7, B_6 \dots \dots B_0$ according to equation (1)

$$v_{ij} = \sum_{r=0}^{7} B_r * 2^r \qquad (1)$$

**Step2**: Divide $\boldsymbol{v_{ij}}$into two subparts $sub1_{ij} = B_7, B_6, B_5, B_4, B_3, B_2$; and $sub2_{ij} = B_1, B_0$.

**Step3**: check overflow/underflow for $sub1_{ij}$ , $sub2_{ij}$ According to the next relations

$$\grave{sub1}_{ij} = \begin{cases} 000001 & sub1_{ij} = 000000 \\ 111110 & sub1_{ij} = 111111 \\ sub1_{ij} & Otherwise \end{cases} \qquad (2)$$

$$\grave{sub2}_{ij} = \begin{cases} 01 & sub2_{ij} = 00 \\ 11 & sub2_{ij} = 11 \\ sub2_{ij} & Otherwise \end{cases} \qquad (3)$$

http://www.doi.org/10.62341/hawe0728

Step4: Embed the first ternary digit ($s_k$) in $sub1`_{ij}$ based on the following cases:

Case 1:$if\ mod(sub1`_{ij}, 3) = s_k$

$$sub1_{ij}^{stego} = sub1`_{ij} \quad (4)$$

Case 2:$if\ mod(sub1`_{ij} + 1,3) = s_k$

$$sub1_{ij}^{stego} = sub1`_{ij} + 1 \quad (5)$$

Case 3:$if\ mod(sub1`_{ij} - 1,3) = s_k$

$$sub1_{ij}^{stego} = sub1`_{ij} - 1 \quad (6)$$

**Step5**: Construct $v`_{ij}$ by using equation (7)

$$v`_{ij} = sub1_{ij}^{stego} * 2^2 + sub2`_{ij} \quad (7)$$

**Step6**: $if\ (k = k + 1) < |S|$ go to Step7 else go to Step9.

**Step7**: Embed the second ternary digit ($s_k$) into $\boldsymbol{v`_{ij}}$, based on the following:

Case1:$if\ mod(v`_{ij}, 3) = s_k$

$$v_{ij}^{stego} = v`_{ij} \quad (8)$$

Case 2:$if\ mod(v`_{ij} + 1,3) = s_k$

$$v_{ij}^{stego} = v`_{ij} + 1 \quad (9)$$

Case 3:$if\ mod(v`_{ij} - 1,3) = s_k$

$$v_{ij}^{stego} = v`_{ij} - 1 \quad (10)$$

**Step8**: If all secret messages are embedded go to Step 9, else $k = k + 1$ then go to step1.

**Step9**: End.

- Extraction Algorithm

Inputs: Stego image is SI with size [M N], the number of rows is M, the number of columns is N, $SI = \{SI_{ij} | 0 \le SI_{ij} \le 255\}$; the $i^{th}$, $j^{th}$ cover pixel is $SI_{ij}$.

**Output**:$\boldsymbol{s}$ ternary secret message.

**Step0:** $k = 0$

**Step1**: convert the pixel value $SI_{ij}$ to binary $B_7B_6 \dots \dots B_0$ according to equation (1)

**Step2**: Divide $SI_{ij}$ into two subparts $sub_1 = B_7B_6B_5B_4B_3B_2$; and $sub_2 = B_1B_0$.

**Step3**: Extract the first ternary number from $sub_1$, according equation (11)

$$s_k = mod(sub_1, 3) \quad (11)$$

http://www.doi.org/10.62341/hawe0728

**Step4**: $if\ (k = k + 1) < |S|$ go to Step5 else go to Step7.
**Step5**: Extract second ternary number of $SI_{ij}$, According to the formula (12)

$$s_k = mod(SI_{ij}, 3) \quad (12)$$

**Step6**: $\boldsymbol{k = k + 1, if\ k < |S|}$ go to Step1 else go to Step7.
**Step7**: End.

**C. Numerical Example for ATD Embedding and Extraction**

Let the cover pixel values be $(\mathbf{135\ 137\ 138})_{\mathbf{10}}$, and the ternary message be $(\mathbf{010212})_{\mathbf{3}}$. We will embed two ternary digits into each cover pixel: 01 into 135, 02 into 157, and 12 into 138.
Embed 01 into cover pixel $v_1 = 135$
**Step1**: Convert cover pixel to binary

$$v_1 = (135)_{10} = (10000111)_2$$

**Step2**: Divide binary value of cover pixel into $sub_1, sub_2$

$$sub_1 = (100001)_2 = (33)_{10}$$
$$sub_2 = (11)_2 = (3)_{10}$$

**Step3**: Check overflow/underflow for $sub_1, sub_2$; the result after Check overflow/underflow according to (2) and (3):

$$sub_1 = (33)_{10}$$
$$sub_2 = (01)_2 = (2)_{10}$$

**Step4**: Embed the first ternary secret number $S_1 = (0)_3$ to $sub_1$

$$sub_1\ mod3 = 33\ mod\ 3 = 0 = S_1$$
$$sub_1^{stego} = 33$$

**Step5**: $\grave{v_1} = 33 * 2^2 + 2 = 134$.
**Step6**: Read the second ternary number $S_2 = (1)_3$ and embed it in $\boldsymbol{\grave{v_1}}$

$$v_1\ mod3 = 134\ mod\ 3 = 2\ \neq S_2$$
$$(v_1\ + 1)mod3 = (134 + 1)mod\ 3 = 0\ \neq S_2$$
$$(v_1\ - 1)mod3 = (134 - 1)mod3 = 133\ mod3 = 1 = S2$$

Hence, $\boldsymbol{v_1}$ stego=133
Embed $(\mathbf{02})_{\mathbf{3}}$ into cover pixel $\boldsymbol{v_2}$=137.
**Step1**: Convert cover pixel to binary

$$v_2 = (137)_{10} = (10001001)_2$$

**Step2**: Divide binary value of cover pixel into $\boldsymbol{sub_1, sub_2}$

$$sub_1 = (100010)_2 = (34)_{10}$$
$$sub_2 = (01)_2 = (1)_{10}$$

**Step3**: Check overflow/underflow for $\boldsymbol{sub_1, sub_2}$according to (2) and (3); no need for change in this step.
**Step4**: Embed $S_3 = (0)_3$ into $sub_1$:

$$sub_1\ \text{mod}3 = 34\ mod\ 3 = 1 \neq S_3$$

http://www.doi.org/10.62341/hawe0728

$$sub_1 \bmod 3 = (34+1)\ mod\ 3 = 2 \neq S_3$$
$$sub_1 \bmod 3 = (34-1) \bmod 3 = 33 \bmod 3 = 0 = 33.$$

Hence: sub1stego=33

**Step5**: $v_2^{`} = 33 * 2^2 + 1 = 133.$

**Step6**: Read the next ternary number $S_4 = (2)_3$and embed it in $v_2^{`}$:

$$v_2 \bmod 3 = 133\ mod\ 3 = 1\ \neq S_4$$
$$v_2 \bmod 3 = (133+1) mod\ 3 = 2 = S_4$$

Hence,

$$v_2^{stego} = 134$$

Embed $(12)_3$ into Cover pixel$v_3$=138.

**Step1**: Convert cover pixel to binary

$$v_3 = (138)_{10} = (10001010)_2$$

**Step2**: Divide binary value of cover pixel into $sub_1, sub_2$

$$sub_1 = (100010)_2 = (34)_{10}$$
$$sub_2 = (10)_2 = (2)_{10}$$

**Step3**: Check overflow/underflow for $sub_1$, $sub_2$ according to (2) and (3); no need for changes in this step.

**Step4**: Embed $S_5 = (1)_3$ into $sub_1$**:**

$$sub_1 \bmod 3 = 34\ mod\ 3 = 1 = S_5$$

Hence*:*

$$sub_1^{stego} = 34.$$

**Step5**: $v_3^{`} = 34 * 2^2 + 1 = 137.$

**Step6**: Read the next ternary number $S_6 = (2)_3$and embed it in $v_3^{`}$

$$v_3^{stego} = 137\ mod\ 3 = 2\ = S_6$$
$$v_3^{stego} = 137$$

The stego pixel are (133, 134, 137).

In the extraction, if we have stego pixel value (133, 134, 137)

stego pixel$v_1$=133.

**Step1**: Convert cover pixel to binary

$$v_1 = (133)_{10} = (10000101)_2$$

**Step2**: Divide the binary value of the cover pixel into $sub_1, sub_2$

$$sub_1 = (100001)_2 = (33)_{10}$$
$$sub_2 = (01)_2 = (1)_{10}$$

**Step3**: Extract first ternary number $S_1$ From $sub_1$according to (11)

$$S_1 = 33\ mod\ 3 = 0$$

**Step4**: Extract second ternary number $S_2$ From $v_1$according to (12)

$S_2 = 133\ mod\ 3 = 1$

The first part of the secret message $(01)_3$is restored.

stego pixel$v_2$=134.

http://www.doi.org/10.62341/hawe0728

**Step1**: Convert cover pixel to binary

$$v_2 = (134)_{10} = (10000110)_2$$

**Step2**: Divide the binary value of cover pixel into $sub_1, sub_2$

$$sub_1 = (100001)_2 = (33)_{10}$$

$$sub_2 = (10)_2 = (2)_{10}$$

**Step3**: Extract $S_3$ from $sub_1$ according to (11)

$$S_3 = 33 \; mod \; 3 = 0$$

**Step4**: Extract $S_4$ From $v_2$ according to (12)

$$S_4 = 134 \; mod \; 3 = 2$$

The second part of the secret message $(02)_3$ is restored.

stego pixel $v_3$=137.

**Step1**: Convert cover pixel to binary

$$v_3 = (137)_{10} = (10001001)_2$$

**Step2**: Divide the binary value of the cover pixel into $sub_1, sub_2$

$$sub_1 = (100010)_2 = (34)_{10}$$

$$sub_2 = (01)_2 = (1)_{10}$$

**Step3**: Extract $S_5$ From $sub_1$ according to (11)

$$S_5 = 34 \; mod \; 3 = 1$$

**Step4**: Extract $S_6$ From $v_3$ according to (12)

$$S_6 = 137 \; mod \; 3 = 2$$

The third part of the secret message $(12)_3$ is restored.

Finally, we embedded $(010212)_3$ into $(135,137,138)_{10}$ and the ternary secret message $(010212)_3$ is restored from the stego pixel $(133,134,137)_{10}$.

**D. Experimental results and analyses**

Color images with a size of 512×512 were used as cover images in the experiments shown in Figure 7.

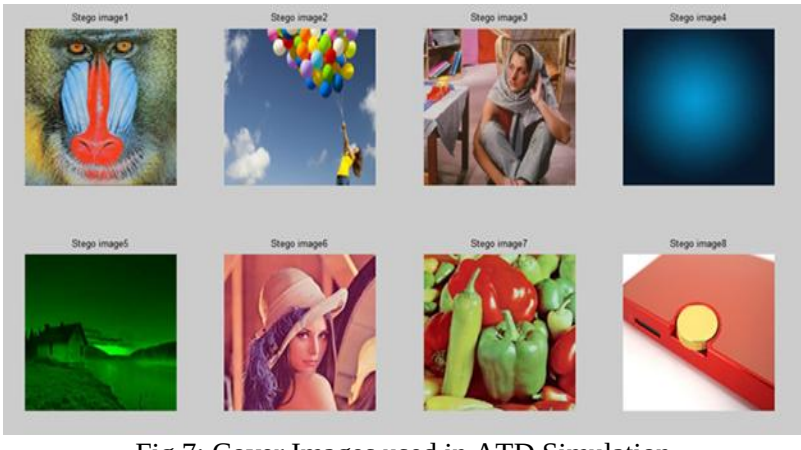


Fig 7: Cover Images used in ATD Simulation

http://www.doi.org/10.62341/hawe0728

The secret message in this experiment is fixed. We converted the secret message to base 3 and embedded four ternary digits (8 bits) into each pixel in different schema 211,121, and 112.
For example, if we used this schema 211 this means than, two ternary numbers are in red and one ternary number in Green and Blue. In addition, we calculated the PSNR, SNR only for modified pixel.

**Table 1 PSNR ATD in different construction results**

| Images | _ schema211 | _ schema121 | _ schema112 |
|---|---|---|---|
| Stego_image1_Banda | 37.4 | 37.4 | 37.4 |
| Stego_image2_Baboon | 37.7 | 37.7 | 37.6 |
| Stego_image3_Barbra | 37.6 | 37.6 | 37.6 |
| Stego_image4_Blue | 37.6 | 37.5 | 37.5 |
| Stego_image5_Green | 34.5 | 34.5 | 34.5 |
| Stego_image6_Lean | 37.6 | 37.6 | 37.6 |
| Stego_image7_Peppers | 37.4 | 37.4 | 37.4 |
| Stego_image8_Red | 36.8 | 36.8 | 36.8 |

From figure 8 and table 1 we show that the value of PSNR in ATD are affected by the image, and the value of PSNR in this algorithm is in range between 34.5 dB and 37.6 dB; the minimum value we get in the Stego_image5_Green image as 34.5 dB in all schema.

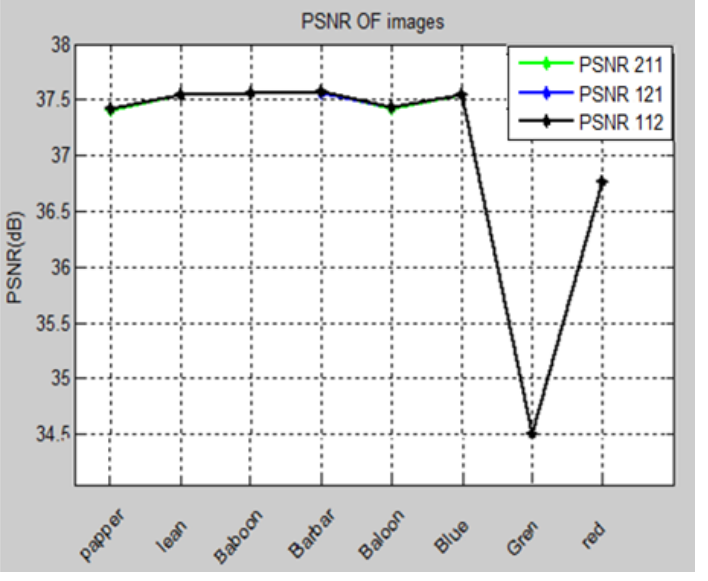


Fig 8: PSNR for ATD in Different Construction

http://www.doi.org/10.62341/hawe0728

In SNR criteria the small value we get in Stego_image5_Green image it value 7.1 dB and Stego_image6_Lean image has the large value of SNR it get 16.5 dB as figure 9.

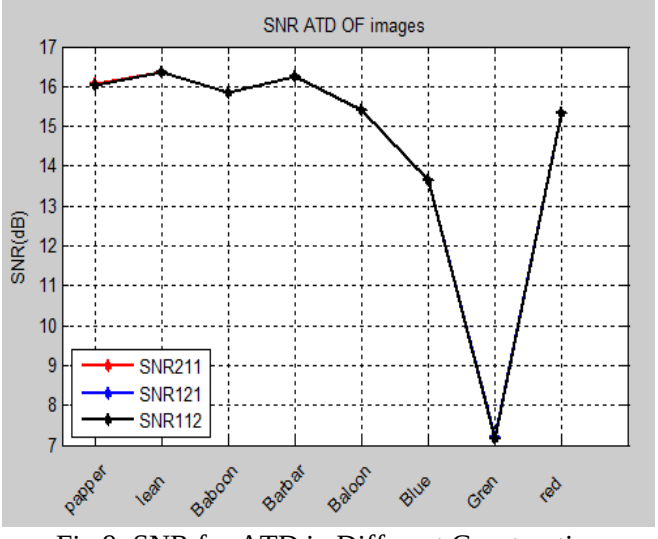


Fig 9: SNR for ATD in Different Construction

## E. CONCLUSION AND FUTURE WORK

We found that the PSNR behavior for color images was not as expected for the same embedding capacity, different embedding combinations applied to the same image resulted in varying PSNR values. In future work, we plan to conduct a more detailed study and analysis of PSNR behavior in the ATD method, with the aim of improving its performance under higher embedding capacities.

## F. References

[1] Agrawal, D, & Samidha, D. (2013, January). Random Image Steganography in Spatial Domain. *IEEE, International Conference*. pp. 1-3. Vol 12. No.34.

[2] Acharya, U. D, Hemalatha, S, & Renuka, A. (2015, March). Wavelet Transform Based Steganography Technique to Hide Audio Signals in Image. *Procedia Computer Science*, pp. 272-281. Vol 47. No.4.

[3] Chang, C. C., Tai, W. L., & Lin, C. C. (2006, October). A Reversible Data Hiding Scheme Based on Side Match Vector Quantization. *IEEE, Transactions on Circuits and Systems for Video Technology*, pp.1301-1308. Vol16. No.10.

http://www.doi.org/10.62341/hawe0728

[4] Yu, Y. H., Chang, C. C., & Lin, I. C. (2007, April). A New Steganographic Method for Color and Grayscale Image Hiding. *Computer Vision and Image Understanding*, pp.183-194. Vol107. No.3.

[5] Taur, J. S., Lin, H. Y., Lee, H. L., & Tao, C. W. (2012, October). Data Hiding in DNA Sequences Based On Table Lookup Substitution. *International Journal of Innovative Computing, Information and Control*, pp. 6585-6598. Vol 8. No.10.

[6] Kiruba, K., & Karthikeyan, S. (2013, February). Reliable Detection of Adaptive

Pixel Pair Matching in Color and Grayscale Images. *IEEE, In Information*

*Communication and Embedded Systems (ICICES)*. pp. 943-946. Vol 5. No.9.

[7] Maji, A. K., Pal, R. K., & Roy, S. (2014, February). A Novel Steganographic Scheme Using Sudoku. *IEEE, In Electrical Information and Communication Technology (EICT)*, pp. 1-6. Vol 14. No28.

[8] Jheng, Y. Z., Chen, C. Y., & Huang, C. F.(2015, September) Reversible Data Hiding Based on Histogram Modification over Ternary Computers. *Journal of Information Hiding and Multimedia Signal Processing*, pp.938 -955. Vol6. No4.

[9] Hussain, M., Wahab, A. W. A., Ho, A. T., Javed, N., & Jung, K. H. (2016, October). A Data Hiding Scheme Using Parity-Bit Pixel Value Differencing and Improved Rightmost Digit Replacement.*Signal Processing: Image Communication*, pp. 44-57. Vol50. No.12.

[10] Xu, W. L., Chang, C. C., Chen, T. S., & Wang, L. M. (2016, May). An Improved Least-Significant-Bit Substitution Method Using The Modulo Three Strategy. *Displays*, pp.36-42. Vol 42. No.17.

[11] Hegde, R., & S, J. (2015, July). Design and Implementation of Image Steganography by Using LSB Replacement Algorithm and Pseudo Random Encoding Technique. *International Journal on Recent and Innovation Trends in Computing and Communication*, pp.4415 - 4420. Vol 3. No.7.