Journal of Al-Nahrain University Vol. 9(1), June, 2006, pp.34-40 Science

Image Steganography in De-Correlated Color Spaces

[1]Dr. Sarab M. Hameed, [1]Sura N. Abdula, [1]Fatin A. Dawood
[1]University of Baghdad,College of Science/Computer Science Department, University of Baghdad

Abstract

This paper introduces a new method for hiding information in de-correlated color space of true color image. Toward this goal, we utilize de-correlated color space to hide the grayscale image into another true color image. The method processes involve: convert the true color cover image into one of the de-correlated space, concealing the grayscale image into the cover image by exploiting the two least significant bits of specified color space, and hiding process in the de-correlated space was performed unsequentialy. Then convert the resulted image into correlated color space. Experimental results show this method provided an acceptable imperceptibility and capacity .Further, this method provides better security than the classical method.

Introduction

Steganography is the art and science of hiding information by embedding messages within other, seemingly harmless messages. The practice of sending secret messages is nothing new, and attempts to cover the messages by hiding them in something else (or by making them look like something else) have been made for millennia [1].

Modern steganography is generally understood to deal with electronic media rather than physical objects and texts [1]. This makes sense for a number of reasons. First of all, because the size of the information is generally (necessarily) quite small compared to the size of the data in which it must be hidden (the cover text), electronic media is much easier to manipulate in order to hide data and extract messages. Secondly, extraction itself can be automated when the data is electronic, since computers can efficiently manipulate the data and execute the algorithms necessary to retrieve the messages. Also, because there is simply so much electronic information available, there are a huge number of potential cover texts available in which to hide information, and there is a gargantuan amount of data an adversary attempting to find steganographically hidden messages must process. Electronic data also often includes redundant, unnecessary, and unnoticed data spaces which can be manipulated in order to hide messages. In a sense, these data spaces provide a sort of conceptual "hidden compartment" into which secret messages can be inserted and sent off to the receiver.

Image steganography has gotten more popular press in recent years than other kinds of steganography, possibly because of the flood of electronic image information available with the advent of digital cameras and high-speed internet distribution. Image steganography often involves hiding information in the naturally occurring "noise" within the image, and provides a good illustration for such techniques [1].

Most kinds of information contain some kind of noise. Noise can be described as unwanted distortion of information within the signal. Within an audio signal, the concept of noise is obvious. For images, however, noise generally refers to the imperfections inherent in the process of rendering an analog picture as a digital image. For example, the values of colors in the palette for a digital image will not only not be the exact colors in the real image, and the distribution of these colors will be also be imperfect.

There are many approaches to hiding the embedded file. The embedded file bits can be inserted in any order, concentrated in specific areas that might be less detectable, dispersed throughout the cover file, or repeated in many places. Careful selection of the cover file type and composition will contribute to successful embedding. A technique called substitution replaces cover file bits with embedded file bits. Since the replacement of certain bits in the cover file will be more detectable than the replacement of others, a smart decision has to be made as to which bits would make the best candidates for substitution. The number of bits in the cover file that get replaced will also affect the success of this method. In general, with each additional bit that is replaced the odd of detection increases, but in many cases more than one bit per cover file byte can be replaced successfully. Combining the correct selection of bits with analysis of the maximum number of bits to replace should result in the smallest possible impact to the statistical properties of the cover file.

One of the more common approaches to substitution is to replace the least significant bits (LSBs) in the cover file [1] [2]. This approach is justified by the simple observation that changing the LSB results in the smallest change in the value of the byte. One significant advantage of this method is that it is simple to understand and implement and many steganography tools available today use LSB substitution.

The Discrete Cosine Transform (DCT) is the keystone for JPEG compression and it can be exploited for information hiding. For one technique, specific DCT coefficients are used as the basis of the embedded file hiding. The coefficients correspond to locations of equal value in the quantization table. The embedded file bit is encoded in the relative difference between the coefficients. If the relative difference does not match the bit to be embedded, then the coefficients are swapped. This method can be enhanced to avoid detection if blocks that are drastically changed by swapping the coefficients are not used for hiding. A slight variation of this technique is to encode the embedded file in the decision to round the result of the quantization up or down [3] [4].

In this paper, we propose an algorithm to hide message (grayscale image) in de-correlated color space of a true color image. The proposed algorithm utilizes the weakness in human visual system, which has a low sensitivity in random pattern changes and luminance. The proposed steganographic algorithm attempts to satisfy the following:

1. Imperceptibility (including both visual imperceptibility and statistical imperceptibility).
2. Security (how difficult it is to break the embedded image).
3. Capacity (how much information can be hidden in a certain media).

The rest of this paper is organized as follows. Section two presents description of the de-correlated color space. Section three clarifies the proposed algorithm. Section four depicts some stego output and summarizes results. Finally, Section five introduces conclusions.

De-correlated Color Space

A Color space is a three-dimensional body used to represent some color or a particular choice of three coordinates that describe color. Many three-dimensional color spaces have been proposed and adapted in different applications in the last few decades. All of them take different approaches to represent the trichromatic nature of light to satisfy their objective [5].

Most color models in use today are oriented either toward hardware such as (RGB mainly for use with color CRT monitor, YIQ for use in the broadcast TV color system, YUV for use in the PAL and NTSC systems of television broadcasting, CMY used in many color printer), or toward applications where color manipulation is a goal (such as in the creation of color graphics for animation)

The RGB system is an example of a color model (space), which is a formal system for defining and representing colors. A somewhat synonymous term is photometric interpretation. The computer video displays are based on an RGB photometric interpretation of color. It is a simple and robust color definition used in computer systems and the Internet to help ensure that a color is correctly mapped from one platform to another without significant loss of color information. RGB uses three numerical components to represent a color. This color space can be thought of as a three-dimensional coordinate system whose axes correspond to the three components, R or red, G or green, and B or blue. RGB corresponds most closely to the behavior of the human eye.

When a typical three channels image is represented in Red, Green, and Blue (RGB) space, there will be correlations between the different channels' values. For example, most pixels will have large values for the red and green channel if the blue channel is large. This implies that if one wants to change the appearance of a pixel's color in a coherent way, he must modify all color channels in random. This complicates any color modification process [6]. Hence, it is desirable to find an orthogonal color space without correlations between the axes. As a result, RGB color information should be transformed into a mathematical space that decouples the brightness information from the color information. After this is done, the image information consists of a one-dimensional brightness or luminance space and a two-dimensional color space [7].

YIQ Color Space

In the development of the United States color television system, the N.T.S.C.(National Television System(s) Committee) formulated a color coordinate system composed of three tri-stimulus values Y-I-Q, a luminance component (Y) and two chrominance or color components: inphase (I) and quadrature (Q) components. The Y is a measure of the luminance of a color, the remaining two tri-stimulus values I and Q jointly describe the hue (that what we normally think of as "color", for example green, blue, or orange) and saturation (the measure of how much white is in the color) [7] [8].

This model was designed to separate chrominance from luminance, this was a requirement in the early days of color television when black-and-white sets still were expected to pick up and display what were originally color pictures. The Y-channel contains luminance information (sufficient for black-and-white television sets) while the I, and Q channels carried the color information. A color television set would take these three channels, Y, I, and Q, and map the information back to R, G, and B levels for display on a screen [8]. Transformations from RGB to YIQ and vice versa are expressed in equation (1) and equation (2) [9]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.595 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \ldots\ldots(1)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.00 & 0.956 & 0.62 \\ 1.00 & -0.272 & -0.647 \\ 1.00 & -1.108 & 1.705 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \ldots\ldots(2)$$

YUV Color Space

The YUV is a format that was first developed for color televisions in order to be compatible with the black and whites and is widely used throughout Europe. Y stands for luminance (or Luma), U (Cb) is the color difference for blue and V (Cr) is the color difference for red. In black and white televisions only the Y component is shown [10].

YUV uses a matrixes combination of Red, Green and Blue to reduce the amount of information in the signal. The Y channel describes Luma (slightly different than Luminance), the range of values between light and dark. Luma is the signal seen by black and white televisions. The U (Cb) and V (Cr) channels subtract the Luminance values from Red (U) and Blue (V) to reduce the color information. These values can then be reassembled to determine the mix of Red, Green and Blue. Some deeper research into YUV reveals two reasons why Blue always looks so crummy when extracted from video images. The U channel ranges from Red to Yellow, the V channel ranges from Blue to Yellow. Because Yellow is Red and Green, Red is essentially sent three times, Green twice and Blue only once. Reconstructing the Luminance component reveals another reason Blue suffers, the Blue channel is only 11% of Luminance [11]. The RGB to YUV conversion and transformation from YUV back to RGB model are defined in equation (3) and equation (4):

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \ldots\ldots(3)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.140 \\ 1.0 & -0.394 & -0.581 \\ 1.0 & 2.028 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \ldots\ldots\ldots\ldots(4)$$

lαβ Color Space

As Ruderman et al. mentioned, lαβ color space was developed to minimize correlation between the three coordinate axes of the color space. The color space provides three decorrelated, principal channels corresponding to an achromatic luminance channel l and two chromatic channels α and β, which roughly correspond to yellow-blue and green red opponent channels. Small changes in one channel impose minimal effect on values of other two [12].

The RGB color space is initially converted to LMS space, which corresponds to the bands of sensitivity of the human cones and was thus, used Runderman et al. as the basis to define lαβ space. This initial transformation corresponds to multiplication by a 3 x 3 matrix as expressed in equation (5) [6]:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \ldots\ldots(5)$$

The conversion from LMS to lαβ is performed through the sequential application of the logarithmic transformation expressed in equation (6) and the linear transformation expressed in equation (7).

$$[L, M, S] = [\log L, \log M, \log S] \ldots\ldots\ldots\ldots\ldots\ldots(6)$$

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \ldots(7)$$

Convert from lαβ color space RGB color space, first, convert lαβ to LMS using the equation(8):

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} \ldots(8)$$

Then, after raising the pixel values to the power ten to go back to linear space, we can convert the data from LMS to RGB using equation (9).

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.467 & 3.587 & 0.119 \\ -1.218 & 2.380 & -0.162 \\ 0.049 & 0.243 & 1.20 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \ldots\ldots(9)$$

The Proposed Algorithm

In this section, we describe a simple algorithm for embedding grayscale image into anther true image. The previous published image steganogrphic methods relay mainly on hiding information on the RGB components of the cover. However, the proposed algorithm relays on concealing grayscale image into the de-correlated color space (e.g., YIQ, YUV, and lαβ) of the cover image. The steps of the proposed steganographic algorithm can simply be stated as follows:

1. Convert the cover image to a de-correlated color space (e.g. YIQ, YUV, or lαβ).
2. Go through the embedded grayscale image in scan line order (i.e. form top to bottom and from right to left) in steps of three bytes at a time.
3. Go through the cover image in scan line order and hide the embedded bytes as:
    - 3.1. Hide the first byte into 2 LSBs of cover luminance components (e.g. Y or l) as: 1st two bits into 2 LSBs of 1st luminance byte, 2nd two bits into 2 LSBs of 2nd byte, 3rd two bits into 2 LSBs of 3rd bye, and 4th two bits into 2 LSBs of 4th byte.
    - 3.2 Hide the second embedded byte into the two LSBs of the first chromatic component of the cover image (e.g., I, U, or α) in the same manner as in 3.1.
    - 3.3 Hide the third embedded byte into the two LSBs of the second chromatic component of the cover image (e.g., Q, V, or β) in the same manner as in 3.1.
4. The last eight bytes of the cover image are utilized to conceal the width and height of the embedded image.
5. Finally, convert the resulted image (stego-image) from its current de-correlated color space into correlated color space.

The embedded image can be extracted from the receiver side, by reversing the steps above. First, the received stego-image should be converted to de correlated color space. To extract the first byte of embedded image, accumulate the 2 LSBs of the first four Y or l component, and then extract the second byte from the first four I (U or α) component, and extract the third byte from the first four Q (V or β) component. By repeating, these three extraction steps *height * width* times, we can obtain the embedded image.

Experimental Results

This section reports some of the experimental results obtained with the proposed algorithm as depicted in figure (1). In the following experiments, we are analyzing the proposed algorithm attributes including imperceptibility, capacity, and security.

Imperceptibility takes advantages of human psycho-visual redendancy. It is difficult to quantify. For image steganography, exiting metrics to measure imperceptibility include Mean Square Error (MSE) as equation (10).

$$MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(f_{ij} - g_{ij} \right)^2 \ldots\ldots\ldots\ldots\ldots\ldots(10)$$

Where M, and N are the height and width of the cover image, f_{ij} is the pixel value of the cover image, g_{ij} is the pixel value of the stego-image. The larger value of MSE represents the low imperceptibility and verse versa.

Capacity represent the amount of information can be embedded in the cover image. The proposed algorithm consumes four bytes from the cover image to hide one byte of the embedded image, so we can quantize the capacity of the proposed algorithm as equation (11):

$$\frac{(MN/4)}{M.N_1} \quad (11)$$

Where M, and N are the height and width of the cover image multiplied by three (since true color image has one byte for each R, G, and B), and M_1, N_1 are the height and width of the embedded image.

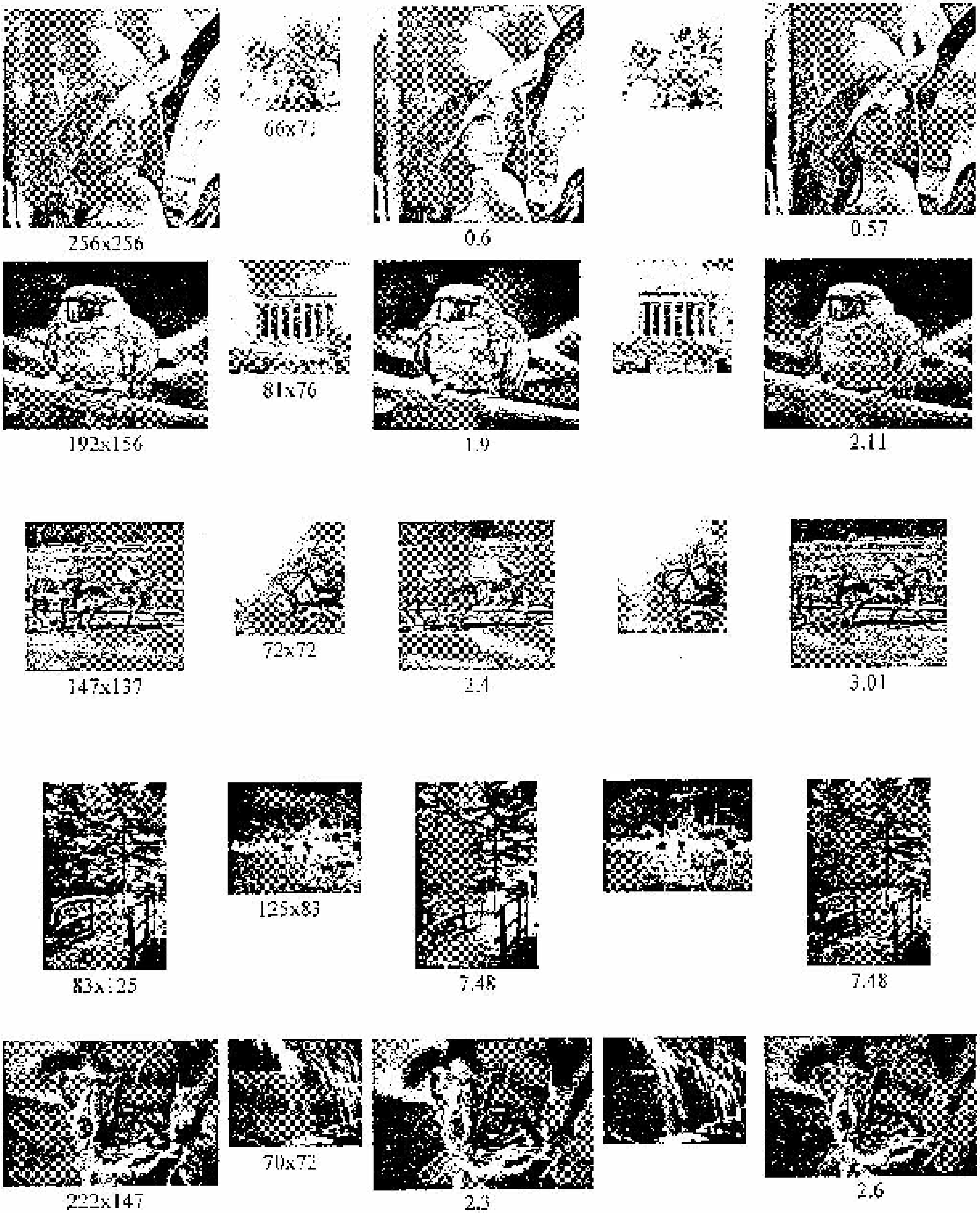
Security is based on of hiding grayscale images in de-correlated color space by converting the color space of the cover image to a de-correlated space and extracting the embedded bytes from the de correlated components in unsequential order.

Conclusions

In this paper, we have introduced a new method for hiding grayscale images in the de-correlated color components of true color images. It is an extension to the classical steganographic algorithm that depends on RGB color space, the proposed algorithm based on de-correlated color space. Based on the experiments, we can conclude that the proposed algorithm has acceptable imperceptibility visually and statically than the classical method. Further, it has high security level than the classical methods, and the capacity is the same as the classical methods. Future work is concerned with improving the security of the scheme by using cryptography techniques and also improving the efficiency by using data compression techniques.

References

1. Katzenbeisser Stefan and Fabien A.P. Petitcolas (eds). Information Hiding: Techniques for Steganography and Digital Watermarking. Boston,:Artech House, 2000.
2. D. Kahn " The history of steganography" in the first workshop on information hiding, R. Anderson, Ed., vol. 1174 of lecture notes in computer science, pp. 1-5, Springer Verlag 1996.
3. W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data Hiding", IBM Systems Journal, vol. 35, no. NOS 3&4, pp.313-315, 1996.
4. Johnson, Neil F. and Sushil Jajodia. "Exploring steganography: Seeing the Unseen." IEEE computer, pp. 26-34, February 1998.
5. Christopher C. Yang and S. H. Kwork, "Efficient Gamut Clipping for Coloe Image Processing Using LHS and YIQ", Optical Engineering, Vol. 42, No. 3, March 2003.
6. Reinhard E. Ashikmin M. B. Gooch, and P. Shirely, "color transfer between images" IEEE Computer Graphics and Applications, Pages 34-41, 2001.
7. Umbagh S. E. "Computer Vision and Image Processing", Prentice-Hall, Inc., 1998.
8. Konzlewski, T., "Theory perceive the color", SEII EM-MI 2002.
9. Hearn, D., Baker, M.P., "Computer Graphics", Prentic Hall, Second Edition 1994.
10. Ponggas D. Gunter F., Vignard A., and Billard A. "Mobile Cameras for the eye of Robort", 2004. Available at URL: http://www.hrn_nodids.effl.ch/research/hardware/report/Pongas.short.pdf.
11. Maller J. "RGB and YUV Color", 2003, available at URL: http://www.joemaller.com/fcp/fxscript_yuv_color.html
12. D.L. Ruderman, T. W. Cronin, and C. C. Chiao. Statistics of cone responses to natural images: Implication for visual coding. J. Opt. Soc. Am A, 15(8):2036-2045, 1998.



256x256 66x71 0.6 0.57

192x156 81x76 1.9 2.11

147x137 72x72 2.1 3.01

83x125 125x83 7.48 7.18

222x147 70x72 2.3 2.6

Figure1. Comparison results between the proposed and classical steganographic methods. 1st column is the cover color image with its size. 2nd column is the embedded grayscale image with its size, 3rd column is our results with its MSE, 4th extraction embedded image from stego image, and 5th column is classical results with its MSE.

المستخلص

يقدم هذا البحث طريقة جديدة لإخفاء المعلومات في الفضاء اللوني الغير مترابط. تم الاستفادة من الفضاء اللوني الغير مترابط لاخفاء صورة رمادية في صورة ذات الالوان الحقيقية . تتضمن هذه الطريقة تحويل الصورة ذات الالوان الحقيقية الى واحد من الفضاءات اللونية الغير مترابطة (مثل YIQ, YUV, lαβ) ثم استغلال البتين الموجودة في اقصى اليمين لاخفاء الصورة الرمادية، وتتم عملية الاخفاء بصورة غير متسلسلة. بعد ذلك يتم تحويل الصورة الى الفضاء اللوني المترابط. اوضحت نتائج التجارب بان هذه الطريقة اعطت دقة وسعة خزنية جيدة بالاضافة الى امنية افضل من الطريقة التقليدية.