

An Investigation for Steganography using Different Color System

Amira B. Sallow

[amira.bibo](mailto:amira.bibo@nawroz.edu.krd)

@nawroz.edu.krd

Nawroz University

Duhok, Iraq

Zahraa M. Taha

College of Computer Science and Mathematics

University of Mosul

Ahmed S. Nori

[ahmed.s.nori](mailto:ahmed.s.nori@uomosul.edu.iq)

@uomosul.edu.iq

Received on:2/9/2010

Accepted on:10/11/2010

ABSTRACT

Steganographic techniques are generally used to maintain the confidentiality of valuable information and to protect it from any possible theft or unauthorized use especially over the internet. In this paper, Least Significant Bit LSB-based Steganographic techniques is used to embed large of data in different color space models, such as (RGB, HSV, YCbCr, YIQ, YUV). The idea can be summarized by transforming the RGB value of the secret image pixels into three separate components into the pixels of the cover image.

The measures (MSE, SNR, PSNR) were used to compare between the color space models, the comparisons proved that steganography with color systems (RGB and HIS) shown a best results.

Keywords- Steganography, LSB.

التقصي في إخفاء المعلومات باستخدام نظام ألوان مختلف

أحمد سامي نوري

كلية علوم الحاسوب والرياضيات

جامعة الموصل

زهراء مازن طه

اميرة بيبو سلو

جامعة نورو

دهوك، العراق

الملخص

استخدمت طرائق تغطية المعلومات للحفاظ على خصوصية المعلومات ذات الأهمية وحمايتها من أي سرقة أو استخدام غير موثوق خاصة عند التعامل مع الانترنت. في هذا البحث، تم استخدام تقنية الإخفاء بواسطة الخانة الأقل أهمية Least Significant Bit LSB للتضمين كمية كبيرة من البيانات في عدة أنواع من النظم اللونية، مثل (RGB, HSV, YCbCr, YIQ, YUV).

تتلخص الفكرة بتحويل قيمة RGB للنقاط الضوئية للرسالة السرية إلى ثلاث مجاميع منفصلة لتضمن داخل النقاط الضوئية للغطاء. استخدمت المقاييس (MSE, SNR, PSNR) للمقارنة بين النظم اللونية المستخدمة في البحث واثبتت المقارنات ان استخدام طريقة تغطية المعلومات في الملفات ذات النظم اللونية (HIS and RGB) تعطي نتائج افضل.

الكلمات المفتاحية: تغطية المعلومات، الخانة الاقل اهمية.

1. Introduction:

Colors are specified for an image by using a color model. The color model's components must be mapped to values (to generate a color space) and these values combined in a meaningful way for each pixel in the image. In English, this means for an image represented by the RGB color model, each pixel needs a red, green and blue value assignment.

The purpose of color models is to organize colors in a standard form. Different models are used according to the user's need. These models are divided in two models: hardware oriented and color manipulation. The color models include RGB, HSI, HSV, YCbCr, YIQ, CMY, CMYK

Color is what you perceive when light within a certain wavelength range hits your eyes. This definition may sound a bit trite, but there is an important implication contained within: color does not exist in the physical world. Indeed, color is entirely subjective and dependent on the observer's eyes, brain and connecting hardware; color is truly in the brain of the beholder. In other words, color is not an absolute characteristic of an object, but a human perception. The color stimulants registered by the retina of the eye are made up of the energy distribution and spectral properties of the visible light passing through or reflected by an object.

color model is a mathematically consistent way of expressing colors. The mathematics of a color model allow you to express any perceived color as a combination of the components of the model. For example, a color represented in the RGB color model is a mixture of three values: a value for red, green, and blue.[1][2]

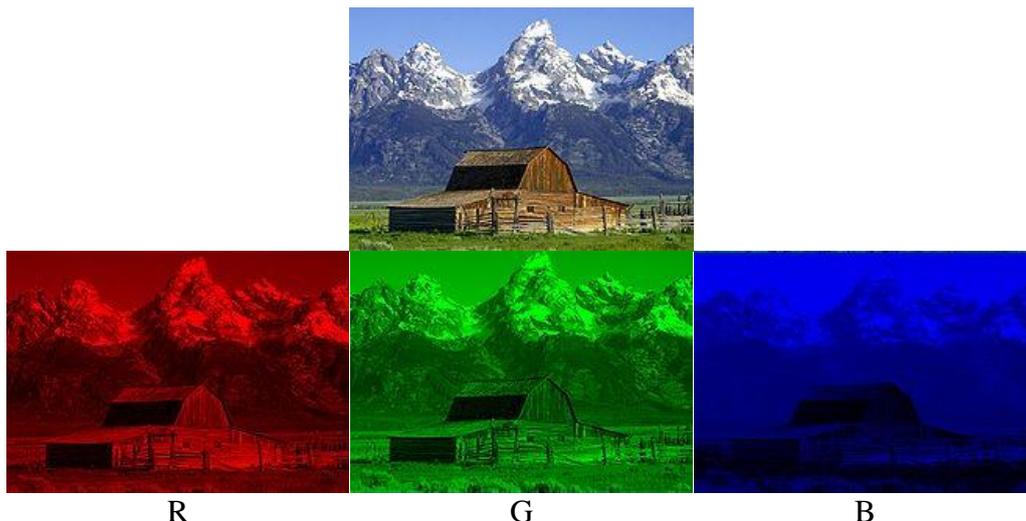
2. Color Model types

2.1 RGB color model

The **RGB color model** is an additive color model in which Red, Green, and Blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, Red, Green, and Blue.

The main purpose of the **RGB** color model is for the sensing, representation, and display of images in electronic systems, such as televisions and computers, though it has also been used in conventional photography. Before the electronic age, the **RGB** color model already had a solid theory behind it, based in human perception of colors, See figure(1).

RGB is a *device-dependent* color space: different devices detect or reproduce a given **RGB** value differently, since the color elements (such as phosphors or dyes) and their response to the individual R, G, and B levels vary from manufacturer to manufacturer, or even in the same device over time. Thus an **RGB** value does not define the same *color* across devices without some kind of color management.[3][4]

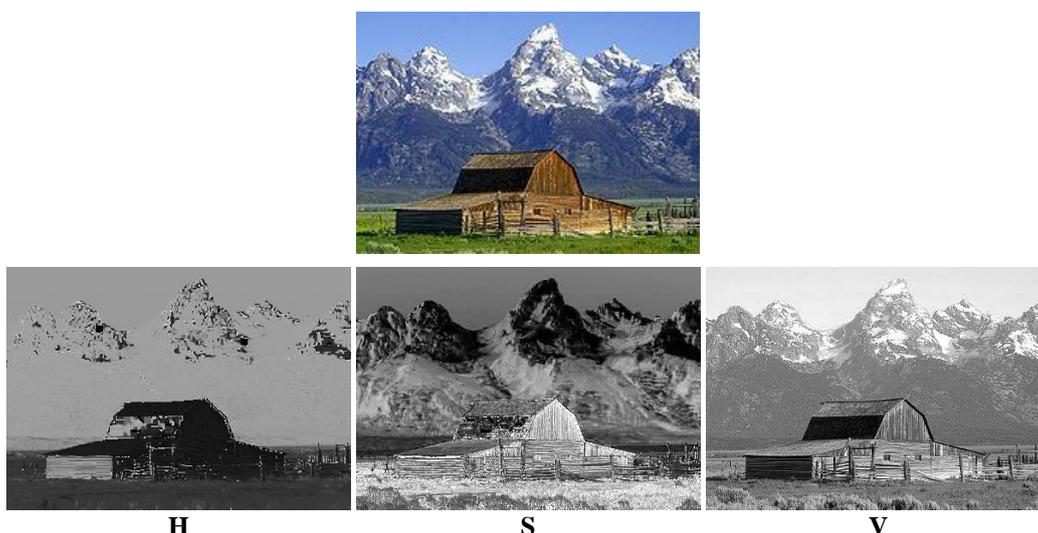


Figure(1): An image along with its R, G, and B components

2.2 HSL AND HSV COLOR MODEL

In the **HSB** (Hue, Saturation , Brightness) and **HSL** (Hue , Saturation, Lightness) models, you specify Hue or color as an angle from 0 to 360 degrees on a color wheel, and Saturation , Brightness , and Lightness as percentages.

Saturation is the intensity of a color . at 100 percent Saturation, a color is pure; at 0 percent Saturation, the color is white , black or gray. Lightness or Brightness is the percentage of black or white that is mixed with a color. A Lightness of 100 percent will yield a white color; 0 percent is black; the pure color has a percent Lightness, See figure (2).



Figure(2): An image along with its H , S and V components

HSL stands for *Hue, Saturation, and Lightness*, and is often also called **HLS**. **HSV** stands for *Hue, Saturation, and value*, and is also often called **HSB** (*B* for *Brightness*). A third model, common in computer vision applications, is **HSI**, for *Hue,*

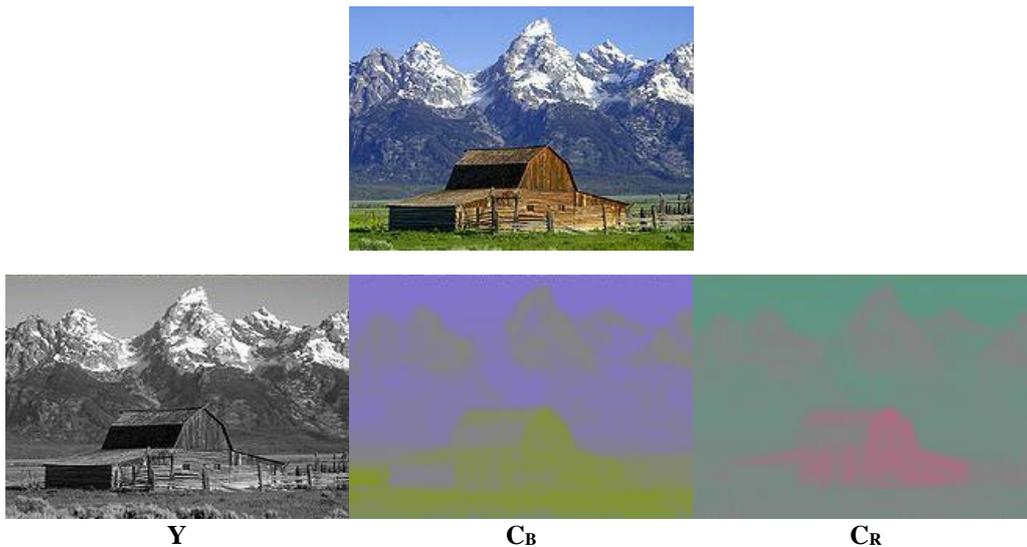
Saturation, and Intensity. Unfortunately, while typically consistent, these definitions are not standardized, and any of these abbreviations might be used for any of these three or several other related cylindrical models.

HSL, HSV, HSI, or related models are often used in computer vision and image analysis for feature detection or image segmentation. The applications of such tools include object detection, for instance in robot vision; object recognition, for instance of faces, text, or license plates; content-based image retrieval; and analysis of medical images. [3][4]

$$\begin{aligned}
 HSI = \begin{cases} H = \begin{cases} \theta & \text{if } B \leq G \\ 360 - \theta & \text{if } B > G \end{cases} \\ \text{where } \theta = \cos^{-1} \left\{ \frac{\frac{1}{2} [(R - G) + (R - B)]}{\left[(R - G)^2 + (R - B)(G - B) \right]^{\frac{1}{2}}} \right\} \\ S = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)] \\ I = \frac{1}{3} (R + G + B) \end{cases} \dots(1) \\
 RGB = \begin{cases} R = I \left[1 + \frac{S * \cos(H)}{\cos(60^\circ - H)} \right] \\ G = 3I - (R + B) \\ B = I(1 - S) \end{cases} \dots(2)
 \end{aligned}$$

2.3 YCbCr color model

YCbCr is sometimes abbreviated to YCC. Y'CbCr is often called YPbPr when used for analog component video, although the term Y'CbCr is commonly used for both systems, with or without the prime. Y'CbCr is often confused with the YUV color space, and typically the terms YCbCr and YUV are used interchangeably, leading to some confusion; when referring to signals in video or digital form, the term "YUV" mostly means "Y'CbCr", See figure (3).



Figure(3): An image along with its Y, C_B, and C_R components

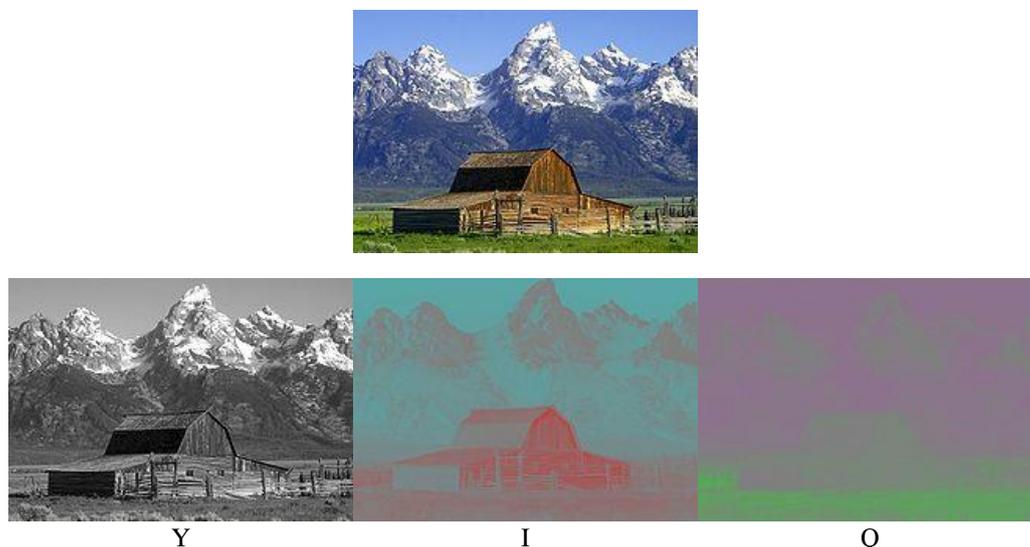
YCbCr or **Y'CbCr**, sometimes written **YCbCr** or **Y'CbCr**, is a family of color spaces used as a part of the color image pipeline in video and digital photography systems. *Y'* is the Luma component and *C_B* and *C_R* are the Blue-difference and Red-difference Chroma components. *Y'* (with prime) is distinguished from *Y* which is luminance, meaning that light intensity is non-linearly encoded using gamma correction. **Y'CbCr** is not an absolute color space, it is a *way of encoding* RGB information. The actual color displayed depends on the actual RGB colorants used to display the signal.[2][4]

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \dots(3)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 0.00456621 & 0 & 0.00625893 \\ 0.00456621 & -0.00153632 & -0.00318811 \\ 0.00456621 & 0.00791071 & 0 \end{bmatrix} \begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} - \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \quad \dots(4)$$

2.4 YIQ COLOR MODEL

YIQ IS THE COLOR SPACE USED BY THE NTSC COLOR TV SYSTEM, EMPLOYED MAINLY IN NORTH AND CENTRAL AMERICA, AND JAPAN. THE **Y** COMPONENT REPRESENTS THE LUMA INFORMATION, AND IS THE ONLY COMPONENT USED BY BLACK-AND-WHITE TELEVISION RECEIVERS. **I** AND **Q** REPRESENT THE CHROMINANCE INFORMATION, SEE FIGURE (4). [1][4][5]



Figure(4): An image along with its Y, I, and Q components

The **YIQ** system is intended to take advantage of human color-response characteristics. The eye is more sensitive to changes in the orange-blue (**I**) range than in the purple-green range (**Q**), therefore less bandwidth is required for **Q** than for **I**. The **YIQ** representation is sometimes employed in color image processing transformations. For example, applying a histogram equalization directly to the

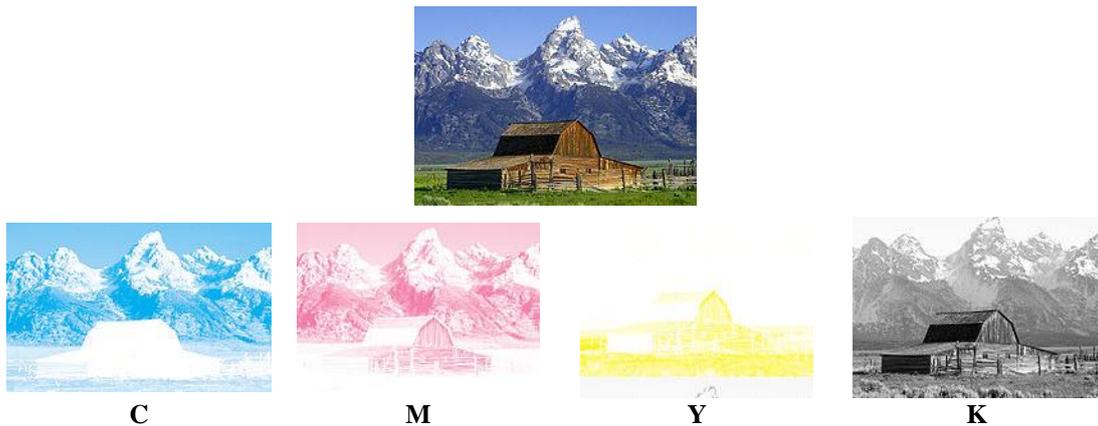
channels in an RGB image would alter the colors in relation to one another, resulting in an image with colors that no longer make sense. Instead, the histogram equalization is applied to the Y channel of the **YIQ** representation of the image, which only normalizes the Brightness levels of the image. The formulae approximate the conversion between the **RGB** color space and **YIQ** is as follow:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.29900 & 0.58700 & 0.11400 \\ 0.595716 & -0.274453 & -0.321263 \\ 0.211456 & -0.522591 & 0.311135 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \dots(5)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0.9563 & 0.6210 \\ 1 & -0.2721 & -0.6474 \\ 1 & -1.11070 & 1.7046 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad \dots(6)$$

2.5 CMYK color model

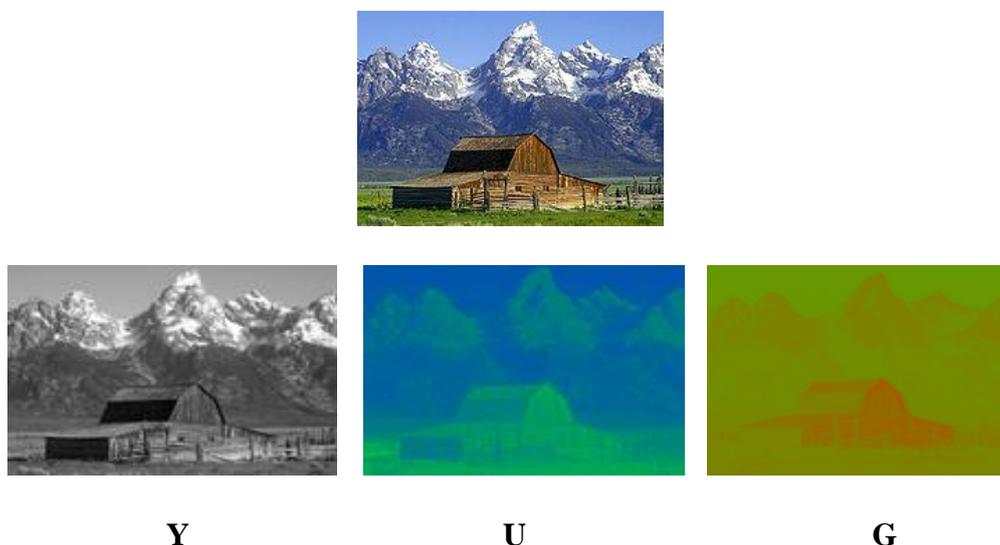
Unlike RGB, which is an additive color model, **CMYK** is a subtractive color model. Typically used in printing, **CMYK** assumes that the background is white, and thus subtracts the assumed Brightness of the white background from four colors: Cyan, Magenta, Yellow, and black (called “key”). Black is used because the combination of the three primary colors (**CMY**) doesn’t produce a fully saturated black. **CMYK** can produce the whole spectrum of visible colors thanks to the process of half-toning, whereby each color is assigned a Saturation level and miniscule dots of each of the three colors are printed in tiny patterns so that the human eye perceives a certain color. Like RGB, **CMYK** is device-dependent. There’s no straightforward formula to convert **CMYK** color to RGB colors or vice versa, so conversion is typically dependent upon color management systems. ColoRotate easily converts one system to the other, See figure (5). [3][4]



Figure(5): An image along with its C, M, Y and K components

2.6 YUV COLOR MODEL

YUV is a color space typically used as part of a color image pipeline. It encodes a color image or video taking human perception into account, allowing reduced bandwidth for chrominance components, thereby typically enabling transmission errors or compression artifacts to be more efficiently masked by the human perception than using a "direct" RGB-representation, See figure (6).



Figure(6): An image along with its Y, U, and V components

Other color spaces have similar properties, and the main reason to implement or investigate properties of $Y'UV$ would be for interfacing with analog or digital television or photographic equipment that conforms to certain $Y'UV$ standards.

The scope of the terms $Y'UV$, YUV , $YCbCr$, $YPbPr$, etc., is sometimes ambiguous and overlapping. Historically, the terms YUV and $Y'UV$ were used for a specific analog encoding of color information in television systems, while $YCbCr$ was used for digital encoding of color information suited for video and still-image compression and transmission such as MPEG and JPEG. Today, the term YUV is commonly used in the computer industry to describe file-formats that are encoded using $YCbCr$.

The $Y'UV$ model defines a color space in terms of one luma (Y') and two chrominance (UV) components. The $Y'UV$ color model is used in the NTSC, PAL, and SECAM composite color video standards. Previous black-and-white systems used only luma (Y') information. Color information (U and V) was added separately via a sub-carrier so that a black-and-white receiver would still be able to receive and display a color picture transmission in the receiver's native black-and-white format.

Y' stands for the luma component (the Brightness) and U and V are the chrominance (color) components; luminance is denoted by Y and luma by Y' – the prime symbols (') denote gamma compression, with "luminance" meaning perceptual (color science) Brightness, while "luma" is electronic (voltage of display) Brightness.[4]

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.29900 & 0.58700 & 0.11400 \\ -0.14713 & -0.28886 & 0.43600 \\ 0.61500 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \dots(7)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.13983 \\ 1 & -0.39465 & -0.58060 \\ 1 & 2.03211 & 0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \quad \dots(8)$$

3. RELATED WORK

- *Adnan Abdul-Aziz Gutub, he* use an improved technique that takes the advantage of the 24 bits in each pixel in the RGB images using the two least significant bits of one channel to indicate existence of data in the other two channels.[6]
- *Rajanikanth Reddy Koppola, he* use LSB-based steganography technique to embed large amount of data in RGBA images while keeping the perceptual degradation to a minimum. And use the YIQ color space model to transform the RGB value of the secret image pixel into three separate components, which are then embedded into the least significant bits of each color pixel in the cover image. [5]
- *Sos S. Agaian and Juan P. Perez,* they propose a steganographic approach for palette-based images. That method had the advantage of embedding secure data, within the index, the palette or both, using special sorting scheme. In the index, information can be embedded using a double pseudo random key that selects blocks and pixels within the image randomly. A stego-bit is embedded at each pixel of the selected blocks. Embedding is accomplished by selecting the closest color using a weighted distance measure that contains the desired modulus-bit. This algorithm also gives you the option of selecting the best color models (i.e. YCbCr and HSV) for embedding purposes.[7]

4. STEGANOGRAPHY

Steganography is the art of hiding communication. The idea is to hide a message inside some cover media without modifying the perception of the media. As an example, many image files contain redundant information. Altering this information does not affect the visual appearance of the image. This fact can be used to create a steganographic system. By embedding a message into the redundant bits of an image, we can send secret messages by posting the images on the web.

Different types of steganographic techniques have been used that employ invisible inks, microdots, character arrangement, digital signatures, covert channel, and spread spectrum communications. Electronic steganography techniques use digital ways of hiding and detecting processes. Steganography is different than cryptography and watermarking although they all have overlapping usages in the information hiding processes. Steganography security hides the knowledge that there is information in the cover medium. Watermarking is different from steganography in its main goal. Watermarking aim is to protect the cover medium from any modification with no real emphasis on secrecy.

It can be observed as steganography that concentrates on high robustness and very low or almost no security . Steganography, in general, may have different

applications. For example, steganography can be utilized for posting secret communications on the Web to avoid transmission or to hide data on the network in case of a violation. It can be useful for copyright protection, which is, in reality, digital watermarking. Copyright protection is to protect the cover medium from claiming its credit be others, with no real emphasis on secrecy. Stego Applications can involve "ownership evidence, fingerprinting, authentication and integrity verification, content labeling and protection, and usage control".[1][2][8]

5.The Proposed Hiding Algorithm

In this paper ,a steganographic algorithm was designed and implemented. Besides, different color models were implemented and programmed to improve the steganographic technique that deals with all these models .

The algorithm can be summarized as:

Step 1: Read color cover_image like C.

Step 2: Image resize to be as secret image size (512x512)

Step 3: Read binary secret_image like S.

Step 4: Convert color made of C to one of the following color model.

- a) RGB
- b) YCbCr
- c) HSV
- d) YIQ
- e) YUV

Step 5: For converting from RGB to any of these color model, We use the equations Number (1),(3),(5),(7).

Step 6: Choose the best slide image from C to hide the secret image in it, and call it Z. From testing we discover that the best slide to hide for each model is:

IF *RGB* then $Z=R$
 IF *Ycbcr* then $Z= Cb$
 IF *HSV* then $Z=V$
 IF *YIQ* then $Z=Y$
 IF *YUV* then $Z=Y$

Step 7: $[r, c] = \text{size}(Z)$

For $i=1$ to r // $r = \text{row}$

For $j=1$ to c // $c = \text{column}$

hide $S(i,j)$ pixel in the LSB of $Z(i,j)$

end

end

Step 8: Convert Z back to RGB using equation (2), (4), (6), (8).

Step 9: Store the steg image.

6.The Proposed Extraction Algorithm

As mentioned in the previous paragraph, here the operation working in reverse manner, so to get the secret message the steps should be in the following sequence :

Step 1: Read steg image to Xc .

Step 2: For converting from RGB to any of these color model, We use the equations Number (1),(3),(5),(7)

Step 3: Select the slide where the secret message was hide , and call it Xz . From testing

we discover that best slide to hide for each model is:

IF *RGB* then $X_z=R$
 IF *Ycbcr* then $X_z = Cb$
 IF *HSV* then $X_z = V$
 IF *YIQ* then $X_z = Y$
 IF *YUV* then $X_z = Y$

Step 4: $[r,c] = \text{size}(X_z)$

For $i=1$ to r // $r = \text{row}$

For $j=1$ to c // $c = \text{column}$

Extract LSB of $X_z(i,j)$ and put it $A(i,j)$

end

end

Step 6: Store the secret image

7. Performance Measurements

To know the amount of difference between the original image and the target image(Stego-image),three kinds for performance measurements between the two images are used,which are

[4][5]:

1- **Mean square error** or **MSE** is one of many ways to quantify the difference between an estimator and the true value of the quantity being estimated and it's equation is:

$$MSE = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} \text{steg_im}(x, y) - \text{cover_im}(x, y) \quad \dots(9)$$

2- **Signal-to-noise ratio** (often abbreviated **SNR** or **S/N**) is a measure used in science and engineering to quantify how much a signal has been corrupted by noise. It is defined as the ratio of signal power to the noise power corrupting the signal.

$$SNR = 10 \cdot \log_{10} \left(\frac{\sum_{x=0}^{m-1} \sum_{y=0}^{n-1} (\text{cover_im}(x, y))^2}{\sum_{x=0}^{m-1} \sum_{y=0}^{n-1} (\text{steg_im}(x, y) - \text{cover_im}(x, y))^2} \right) \quad \dots(10)$$

3- **peak signal-to-noise ratio**, often abbreviated **PSNR**, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.

$$PSNR = 20 \cdot \log_{10} \left(\frac{(\text{Max value of Gray level})}{RMSE} \right) \quad \dots(11)$$

$$RMSE = \sqrt{MSE} \quad \dots(12)$$

8. Results

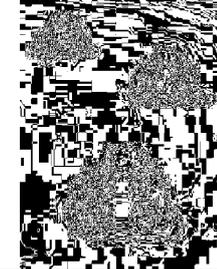
See figures (7)(8)(9) and (10), also tables (1)(2)(3) and (4).

	Cover Image	Stego Image	Secret Image	Extracted Secret Image
RGB				
HSV				
Ycber				
YIQ				
YUV				

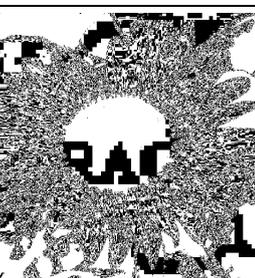
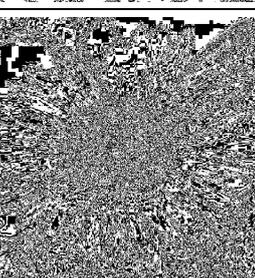
Figure(7): Final Results For Applying Different Color Models Using Sample (1)

	Cover Image	Stego Image	Secret Image	Extracted Secret Image
RGB				
HSV				
Ycbr				
YIQ				
YUV				

Figure(8): Final Results For Applying Different Color Models Using Sample (2)

	Cover Image	Stego Image	Secret Image	Extracted Secret Image
RGB				
HSV				
Yeber				
YIQ				
YUV				

Figure(9): Final Results For Applying Different Color Models Using Sample (3)

	Cover Image	Stego Image	Secret Image	Extracted Secret Image
RGB				
HSV				
Ycbr				
YIO				
YUV				

Figure(10): Final Results For Applying Different Color Models Using Sample (4)

Table (1): comparisons between the different color models

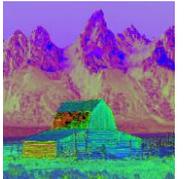
Images					
	Measures				
	RGB	HSV	YCBCR	YIQ	YUV
MSE	0.132913	0.361792	0.756934	82.905178	10.963295
SNR	50.266757	45.984566	42.155010	6.028304	6.754600
PSNR	56.895131	52.546211	49.340224	28.944987	37.731393

Table (2): comparisons between the different color models

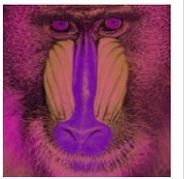
Images					
	Measures				
	RGB	HSV	YCBCR	YIQ	YUV
MSE	0.132977	0.355536	0.799234	82.435430	15.241652
SNR	50.555823	46.279980	42.448595	6.470313	5.503946
PSNR	56.893054	52.621966	49.104065	28.969665	36.300483

Table (3): comparisons between the different color models

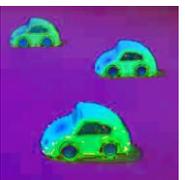
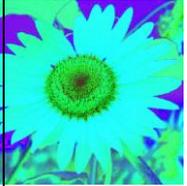
Images					
	Measures				
	RGB	HSV	YCBCR	YIQ	YUV
MSE	0.372122	0.298155	0.457161	80.338722	12.047230
SNR	46.784296	48.471867	44.468233	6.136386	6.182842
PSNR	52.423945	53.386376	51.530112	29.081554	37.321932

Table (4): comparisons between the different color models

Images					
	Measures				
	RGB	HSV	YCBCR	YIQ	YUV
MSE	0.137831	0.298155	0.708468	82.435430	44.943413
SNR	51.772122	48.471867	44.166045	6.470313	4.923764
PSNR	56.737323	53.386376	49.627600	28.969665	31.604143

9. Conclusions

The imperceptibility of steganography is one of the most important measures that evaluate the performance of the steganography algorithm. The global evaluation models (MSE, SNR, and PSNR) which are widely used by most researchers were presented and used to evaluate the performance of steganography algorithm effectively.

Experimental results demonstrate that our algorithm achieves high embedding capacity while preserving good image quality and high security; especially when using the color models (RGB and HSI). The important and distinctive features in the proposed method are to minimize the degradation of stego image.

REFERNCE

- [1]. Nelson, P. , 2007, “The Photographer’s Guide to Color Management: professional techniques for consistent results”, Amherst Media , Korea.
- [2]. Rattanapitak, W. and Udomhunsakul, S. , 2010 , “Comparative Efficiency of Color Models for Multi-focus Color Image Fusion”, Hong Kong.
- [3]. Cheddad, A. , Condell, J. , and Curran, K. , 2009 “A Skin Tone Detection Algorithm for an Adaptive Approach to Steganography “ , School of Computing and Intelligent Systems, Faculty of Computing and Engineering , University of Ulster, Londonderry, United Kingdom.
- [4]. Vaughan, T. , 2007, “ Multimedia Making it Work” seventh edition, Osborne-McGraw Hill.
- [5]. Koppola, R.R. , 2009 , “ A HIGH CAPACITY DATA-HIDING SCHEME IN LSB-BASED IMAGE STEGANOGRAPHY” , Master thesis , University of Akron
- [6]. Gutub ,A.A. , 2010 , “Pixel Indicator Technique for RGB Image Steganography” , Center of Excellence in Information Assurance (CoEIA), King Saud University, Saudi Arabia.
- [7]. Aгаian, S.S. , and Perez , J.P. , 2004 , “NEW PIXEL SORTING METHOD FOR PALETTE BASED STEGANOGRAPHY AND COLOR MODEL SELECTION “ ,The University of Texas, San Antonio,
- [8]. Gonzalez, R.C. , and Woods, R.E. , 2002, “Digital Image Processing”, Prentice-Hall, USA.