A Recursive Algorithm to Hide Three Secret Images in One Image Using Wavelet Transform

Yasmin Muwafaq kassim
Engineering College, University of Al-Nahrain /Baghdad
Email:yasminma@yahoo.com

Received on: 2 /6/2011 & Accepted on: 3/11/2011

ABSTRACT

This paper presents an algorithm based on wavelet transform to hide three secret colored or gray-scale images with different sizes in one colored cover image. The algorithm takes level1 wavelet transformation for the cover image and level2 wavelet transformation for the coefficients resultant from level1. The algorithm begins to divide and transpose the secret images into multiple sub bands, then imbedding them into the coefficient parts resulting from level2. The embedding depends upon a variable threshold which begins with a very small value. Here the algorithm ensures the embedding of all the pixels values of the sub band, if it is not, the operation will be repeated with a larger threshold value until all the pixels are embedded. Also the pixel's value will not be embedded directly, the difference between the cover and the secret pixel value will be embedded instead of it after some manipulation (mathematical operations). All of these factors (divide and transpose the secret images, the variable threshold for each sub band and changes on the embedded pixels) increase the robustness and quality of the algorithm. The resultant stego image and the extracted secret images are very close to the original one with high PSNR, high Correlation, low Normal Absolute Error and low Maximum Difference.

Keywords: Image Steganography, Discrete Wavelet Transform, Stego Image, Threshold, PSNR(Peak Signal to Noise Ratio) and Correlation Test(Corr).

الخوارزمية التكرارية لإخفاء ثلاثة صور في صورة واحدة باستعمال التحويل الموجي

الخلاصة

هذا البحث يقدم خوارزمية لإخفاء ثلاثة صور سرية ملونة أو ذات التدرج الرمادي بمختلف الأحجام في صورة غطاء واحدة ملونة بعد تعريضها إلى المستوى الأول للتحويل الموجي و من ثم تعريض المعاملات الناتجة إلى المستوى الثاني للتحويل الموجي. الخوارزمية تبدأ بتقسيم الصور السرية الى حزم و تغيير مواضع النقاط فيها وبعد ذلك تبدأ بإخفاء الحزم داخل المعاملات الناتجة من التحويل الثاني بالاعتماد على حد متغير, هذا الحد يبدأ بقيمة صغيرة جدا و بعد ذلك تبدأ الخوارزمية بالتأكد من إن جميع نقاط الحزمة ممكن أن تخفى بهذا الحد و بخلافه تقوم بزيادة قيمة الحد و تستمر العملية

https://doi.org/10.30684/etj.30.2.5
2412-0758/University of Technology-Iraq, Baghdad, Iraq

السابقة إلى أن تخفى جميع نقاط الحزمة , عند عملية الإخفاء قيمة النقطة سوف لن تخفى بشكل مباشر و إنما الفرق بين قيمة نقطة معامل الغطاء و قيمة النقطة السرية هو الذي سوف يخفى و بعد إدخالها أيضا لبعض العمليات الرياضية. كل هذه العوامل سوف تزيد من قوة و كفاءة الخوارزمية . صورة الغطاء الناتجة و الصور السرية المستخرجة لها مشابه جدا للصوره الأصلية **مع** (PSNR) عالي , (Corr) عالي, (NAE) قليل و ال(MD) قليل.

الكلمات المرشدة: علم إخفاء الصور, التحويل الموجي المتقطع, الصورة المشفرة, الحد,مقياس نسبة أعلى موجة إلى الخطأ, مقياس الترابط.

INTRODUCTION

Steganographyis the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. The word steganography is derived from the Greek words "*stegos*" meaning "cover" and "*grafia*" meaning "writing" [1] defining it as "covered writing". In image steganography the information is hidden exclusively in images.

The idea and practice of hiding information has a long history. In *Histories* the Greek historian Herodotus writes of a nobleman, Histaeus, who needed to communicate with his son-in-law in Greece. He shaved the head of one of his most trusted slaves and tattooed the message onto the slave's scalp. When the slave's hair grew back the slave was dispatched with the hidden message [2]. In the Second World War the Microdot technique was developed by the Germans. Information, especially photographs, was reduced in size until it was the size of a typed period. Extremely difficult to detect, a normal cover message was sent over an insecure channel with one of the periods on the paper containing hidden information [3]. Today steganography is mostly used on computers with digital data being the carriers and networks being the high speed delivery channels.

Images are the most popular cover objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications. For these different image file formats, different steganographic algorithms exist.

Image steganography techniques can be divided into two groups: those in the Image Domain and those in the Transform Domain [2]. Image – also known as spatial – domain techniques embed messages in the intensity of the pixels directly, while for transform – also known as frequency – domain, images are first transformed and then the message is embedded in the image [4].

In the spatial domain, the secret message is inserted directly into the pixels. This method of data hiding results in distortions of the original image. The original image is not lossless and reversible. This results in a permanent distortion of the original image or hidden data. Hence, a lossless data embedding, which is also called reversible data embedding, is the need to embed invisible data into a digital image. In the frequency domain, the common well-known method for data hiding is based on discrete wavelet transform(DWT) [5].

DISCRETE WAVELET TRANSFORM

The wavelet transform is a technique for analyzing signals. It divides a signal into different frequency components each with different resolution. The Discrete Wavelet Transform (DWT) is used when a signal being sampled such as with digital image processing. An efficient way to implement DWT uses filters. The signal is filtered by Low Pass Filter LPF and High Pass Filter HPF then the signal is down sampled. The resulting low pass signal is called approximation and it is much similar to the original signal. The high pass signal is called details signal.

There are several types of wavelet (such as Haar , Daubechies, Coifman, symlets, and Morlet). For images Two_Dimensional Discrete wavelet Transform 2D_DWT decomposes image into multi levels of independent information. Images will be transformed in each level of decomposition to four bands, one low information image and three details images, When the DWT is used to decompose the resulting subspaces (a, h, v and d) then the resulting transform is called Discrete Wavelet Packet Transform DWPT. The four subspaces low-low (approximation), low-high (horizontal), high-low (vertical) and high-high (diagonal) are depicted in Figure (1). While Figure(2) shows an image with its level-1 DWT. Viewing Figure(2) gives two indications, the first is that the most information of an image is contained in the approximation subspace 'a' of DWT and little information is contained in each of the details subspaces 'h' ,'v', and 'd'.

So that the secret image data should be placed in details subspaces of DWT of the cover image and not in the approximation in order not to deteriorate the cover image. The second indication is that the coefficients values of the details sub spaces are low, since they appear black in Figure(2_b). So that the data to be embedded in these subspaces should be low in order to make the resulting stego_image more similar to the cover image [6].

DESCRIPTION OF THE PROPOSED WORK

This research presents two algorithms. The first one embeds three color secret images in one cover color image and the second embeds three grayscale secret image in one color cover image. The two algorithms depend upon two principles sub algorithms, the first one is Divide and Transpose algorithm and the second is Embedding algorithm.

Divide and Transpose Algorithm

This algorithm takes an input image and divides the image into three sub bands longitudinally as output. It is not normal dividing, the divide operation interfered with transposition ciphering which means that the position of the pixels will be changed to increase the security and robustness of the algorithm. The transposition is done in such away that each plate is divided into 3 sub bands: the first sub band includes column (i) to column (n) incremented by 3, the second sub band includes column (i+1) to n incremented by 3 and finally the third sub band includes column i+2 to n, the preceding steps has been programmed as a function in MATLAP called div_trans as follows:

```
Function [s1 s2 s3]= div_trans(image)
s=1;
j=1;
[m n]=size(image);

for i=1:m

j=1;
for s=1:(n/3)
sfb1(i,s)=image(i,j);
j=j+3;
end
j=1;
for s=1:(n/3)
sfb2(i,s)=image(i,j+1);
j=j+3;
    end
   j=1;
    for s=1:(n/3)
     sfb3(i,s)=image(i,j+2);
     j=j+3;
    end

end
s1=sfb1;
s2=sfb2;
s3=sfb3;
```

3.2 Embedding algorithm

The embedding process is recursive, the embedding depends upon a threshold value. The threshold value is not constant like the traditional algorithms, it begins with very small value which represents the difference between the pixel will be embedded and the pixel will be removed. If all the sub band pixel values find pixels with such threshold, the embedding is done and if not, this algorithm will be repeated with a larger threshold and so on. The variable threshold will increase the robustness of the algorithm (because not all sub bands will be embedded with the same threshold) besides finding the ideal pixels for embedding**.**

```
function [stegosb,embed]=Embed_Secret(csb,ssb)
=0;                              / counter to ensure that all secret pixels are embedded
      threshold=0.01;                / begin with very minimum value to guarantee
ideal embedding
      [z1 z2]=size(ssb);             / z1 & z2 represent the number of rows and
columns of secret sub band
f
    while f~=(z1*z2)                 /to ensure that all pixels are embedded

      s1=1;                          /s1 and s2 are counter variables for secret sub
band pixels
      s2=1;
      flag=0;                        / flag to ending the function loops if all secret
pixels are embedded
      cover=csb;
      secret=ssb;
[m   n]=size(cover);             / m & n represent the number of rows and columns of
cover sub band(csb)

       for i=1:m
         for j=1:n

if  abs(cover(i,j))< threshold)
          embed(i,j)=1;
          f=f+1;
          diffvec(i,j)=cover(i,j)-secret(s1,s2);
          cover(i,j)=diffvec(i,j)*0.001;
          s2=s2+1;
            if s2==z2+1
             s2=1;
             s1=s1+1;
             if s1==z1+1
             flag=1;
             break;
             end
            end
          end
end
        if flag==1
        break
        end
       end
      f=0;
```

```
      Threshold=threshold+0.02;     / increment the value of threshold if not all
pixels are embedded.
      End
Stegosb=cover;                      / here the function will ended with giving the value
of cover to
    stegosb
```

/To ensure that s2 is through the boundary of the ssb and if the /condition is true, s2 will return 1 and s1 will be incremented to take /the second row with all values of columns.

/The second condition will check if s1 arrive to maximum value, the /flag will be true to end the iterations because all the secret pixels are /embedded through the cover but if not, the iterations (for loops) will /be ended but with out embedding all pixels and the big iteration will /continue with increasing the value of threshold to increase the chance /of embedding.

```
    Function [stegosb,embed]=Embed_Secret(csb,ssb)

      f=0;                          / counter to ensure that all secret pixels are
embedded
       Threshold=0.01;              / begin with very minimum value to guarantee
ideal embedding
       [z1 z2]=size (ssb);          / z1 & z2 represent the number of rows and
columns of secret sub band

While f~= (z1*z2)                 /to ensure that all pixels are embedded

      s1=1;                         /s1 and s2 are counter variables for secret sub
band pixels
      s2=1;
      Flag=0;                       / flag to ending the function loops if all secret
pixels are embedded
cover=csb;
      secret=ssb;
      [m   n]=size(cover);          / m & n represent the number of rows and
columns of cover sub band(csb)
        for i=1:m
for j=1:n
          if  abs(cover(i,j))< threshold)
          embed(i,j)=1;
          f=f+1;
          diffvec(i,j)=cover(i,j)-secret(s1,s2);
          cover(i,j)=diffvec(i,j)*0.001;
          s2=s2+1;
if s2==z2+1
```

```
        s2=1;
        s1=s1+1;
        if s1==z1+1
        flag=1;
        break;
        end
       end
      end

     end
     if flag==1
     break
     end
    f=0;
    threshold=threshold+0.02;     / increment the value of threshold if not all pixels
are embedded.
    end

   stegosb=cover;                  / here the function will ended with
```

giving the value of cover to stegosb
To ensure that s2 is through the boundary of the ssb and if the /condition is true, s2 will return 1 and s1 will be incremented to take /the second row with all values of value, the /flag will be true to end the iterations because all the secret pixels are /embedded through the cover but if not, the iterations (for loops) will /be ended but with out embedding all pixels and the big iteration will /continue with increasing the value of threshold to increase the chance /of embedding.
/ columns.
/The second condition will check if s1 arrive to maximum

The other factor which will increase the robustness of the algorithm is that it does not store the secret pixels value directly in to the cover, it stores the difference between the cover and the secret pixel value after some manipulation and will be represented as difference vector (diffvec()). The algorithm of embedding method is programmed by MATLAB as a function called Embed_Secret with two inputs and two outputs. The first input is the cover sub band(csb) and the second one is the secret sub band (ssb). The output from the function is the stego sub band (stegosb) in which ssb is hidden inside it and the other one is the embed vector typed as embed() represents the indexes of embedded pixels in the stego cover and it is required in the reconstruction operation to reconstruct the secret images. This vector will filled by 1 value in the indexes of allocation to know how to extract the secret images by inversing the operation of embedding.
Embed_Secret function is described in the preceding page with many notes to clarify the whole operation.

Embedding three color images in one color image algorithm

The embedded process is described as follows:

1. Split the cover image C into three plates (red, green and blue) represented as Cr, Cg and Cb.
2. Apply level-1 wavelet decomposition for the three plates of cover image to result 4 bands for each plate represented as (Dcr1,Dcr2,Dcr3 and Dcr4) for red plate, (Dcg1,Dcg2,Dcg3 and Dcg4) for green plate and (Dcb1,Dcb2,Dcb3 and Dcb4) for blue plates.
3. Apply level-2 wavelet decomposition for all the coefficient parts for all plates to result 36 bands (27 as coefficient bands and 9 as approximation bands represented as (Dcr21, Dcr22, Dcr23 and Dcr24) , (Dcr31, Dcr32, Dcr33 and Dcr34) and (Dcr41, Dcr42, Dcr43 and Dcr44) for red plate and the same for green and blue plate but with alternative names as described in figure(3).
4. Split the three secret images Sf, Sc and St in to 9 planes (each image in to three planes red, green and blue) resulting Sfr, sfg, sfb, scr, scg, scb, str, stg and stb.
5. Divide and transpose the nine plate of secret images by using Divide and transpose algorithm in such away that each plate is divided into 3 sub bands: the first sub band includes column (i) to column (n) incremented by 3, the second sub band includes column (i+1) to n incremented by 3 and finally the third sub band includes column i+2 to n incremented by 3 resulting 27 sub bands (sfr1,sfr2,sfr3,sfg1,sfg2,sfg3,sfb1,sfb2,sfb3,scr1,scr2,scr3,scg1,scg2,scg3,scb1,scb2,scb3,str1,str2,str3,stg1,stg2,stg3,stb1,stb2 and stb3).
6. Each of the preceding sub bands is embedded into the coefficient parts described in step 3 (27 secret sub bands in 27 cover band) by the embedding algorithm described above in such a way that all the red sub bands of the three secret images will embedded into red plate coefficients of C and this is repeated for green and blue sub bands.
7. Now all the sub bands are embedded into the cover image so the reconstruction process will begin from level-2 to level-1 and then from level-1 to result stego imge.

The schematic diagram of the whole operation is described in Figure (3), The similar colors refers to the embedding operation, the author choose to put similar colors instead of arrows to prevent arrows intersection which lead to reader confusing.

Embedding three Gray scale secrets images in one color image algorithm

The algorithm steps is described as follows:

1. Split the cover image C into three plates (red, green and blue) represented as Cr, Cg and Cb.
2. Apply level-2 wavelet decomposition for the coefficient parts (Dcr2,Dcr3 and Dcr4) resulting (Dcr21, Dcr22, Dcr23 and Dcr24) , (Dcr31, Dcr32, Dcr33 and Dcr34) and (Dcr41, Dcr42, Dcr43 and Dcr44) and the same for green and blue plate.

3. Divide and transpose the three secret images in such away that each secret is submitted to the divide and transpose algorithm twice to result nine bands for each secret like that: (sf11,sf12,sf13,sf21,sf22,sf23,sf31,sf32 and sf33) for the first secret, (sc11,sc12,sc13,sc21,sc22,sc23,sc31,sc32 and sc33) for the second secret and (st11,st12,st13,st21,st22,st23,st31,st32 and st33) for the third secret.
4. Each of the preceding sub bands is embedded into the coefficient parts of C like below, this is the embedding operation of the first secret image only and all the operation is obvious in Figure(4):
   Sf11 → Dcr22
   Sf12 → Dcr23
   Sf13 → Dcr24
   Sf21 → Dcg22
   Sf22 → Dcg23
   Sf23 → Dcg24
   Sf31 → Dcb22
   Sf32 → Dcb23
   Sf33 → Dcb24
5. Now all the sub bands are embedded (by the same preceding embedding algorithm) into the cover image so the reconstruction process will begin from level-2 to level-1 and then from level-1 to stego image.

   The schematic diagram of the whole operation is described in Figure(4). The similar colors refers to the embedding operation, the author choose to put similar colors instead of arrows to prevent arrows intersection which lead to reader confusing.

Extracted of secret images

The result of the the two cases described in section 3.3 and 3.4 is stego image. This stego image will be transmitted over a channel to the receiver which has the extracting algorithm to extract secret images. This algorithm depends also on two subalgorithms which are redivide and retranspose algorithms which has the inverse steps of sections 3.1 and reembedd algorithm which has the inverse steps of embed algorithm described in section 3.2.

Peak signal-to-noise Ratio

Peak Signal-to-Noise Ratio is used to measure the difference between two images. It is defined as

$$PSNR = 20 * \log10(b/rms)....(1)$$

where b is the largest possible value of the signal (typically 255 or 1), and rms is the root mean square difference between the two images. The PSNR is given in decibel units (dB), which measures the ratio of the peak signal and the difference between two images. An increase of 20 dB corresponds to a ten-fold decrease in the rms difference between two images.

There are many versions of signal-to-noise ratios, but the PSNR is very common in image processing, probably because it gives better-sounding numbers than other measures[9].

Normal Absolute Error (NAE)

It is the division between summation and errors(the difference between the original pixel and distorted pixel) of all pixel's value, it is computed in MATLAB as:

```
error = origImg - distImg;
NAE = sum(sum(abs(error))) / sum(sum(origImg));
```

Maximum Difference (MD)

It is the maximum error value obtained by decreasing the distorted pixel value from the original one and it is calculated as

```
error = origImg - distImg;
MD = max(max(error));
```

Correlation

Similarity test is the correlation between the cover-image and stego-image. When the stego-image is perceptually similar to the original cover-image, then the correlation equals one. The correlation can be calculated as shown below :

$$Corr=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(X(i,j)-\overline{X})(Y(i,j)-\overline{Y})}{\sqrt{\left[\sum_{i=1}^{M}\sum_{j=1}^{N}(X(i,j)-\overline{X})^2\right]\left[\sum_{i=1}^{M}\sum_{j=1}^{N}(Y(i,j)-\overline{Y})^2\right]}} \quad .... (2)$$

where M and N: height and width of the two images (because the two images must be of the same size).

i and j: row and column numbers.

X(i, j): the original image.

Y(i,j): modified image.

$\overline{X}$, $\overline{Y}$: Meaning original and modified image respectively [10]; calculated by

$$\overline{X} = \frac{\sum_{i}^{M}\sum_{j}^{N} X(i,j)}{M \times N} \quad ..(3)$$

$$\overline{Y} = \frac{\sum_{i}^{M}\sum_{j}^{N} Y(i,j)}{M \times N} \quad ..(4)$$

RESULTS AND DISCUSSION

This section presents an example for implementing the demonstrated algorithm (which is programmed with MATLAB R2010a) as displayed in Figure(5). Tables(3-7) and Figures(6-9) are results for implementing the same algorithm for multiple standard images to show the difference between embedding three gray-scale images and three colored images with different sizes.

Some notes about the results (tables results and related figures) are illustrated as follows:

1. The results in table (3) and table(4) indicate that there is an inverse relation between the increasing in image size and the test measure(PSNR and Correlation).

2. Since the work chooses the 9 most dark parts in the cover image for embedding (each of them has size of 128*128=442368 pixels), the maximum size for embedding three colored images in one colored image is 220*220 and the maximum size for embedding three gray-scale images is 380*380. The justification is demonstrated in tables (1) and(2).

3.The measure tests will be decreased when there are many details in the cover picture like Baboon in table(5) is less than lena in table(6).

4. There is a case in which the PSNR of all secret images will be "infinity" and the correlation will be "1" for some image sizes as displayed in some rows in table(3) for gray_scale images and table(4) for color images, and this is because there are no lost in pixels through embedding operation, all these image sizes are divisible by 3 according to the algorithm used while all image sizes in the rest of the tables are not divisible by 3 which results loss of some columns, as example for image size of 100*100, 100/3=33.33 (not divisible) while 108/3=36 (divisible), also the value of NAE and MD is 0 for the same reason.

5. The value of NAE increase with the increasing of the size of secret images embedding in the cover but it is still very small value. For example in table(3), the value is 0.0202 for maximum size of secret images which is very small value and considered ideal (0 is ideal).

6. The value of MD also increased with the increasing of the size of secret images and in the maximum size it's value is 0.1212 which is considered very good value.

5. CONCLUSIONS

The presented algorithm is very efficient for hiding three secret images (colored or gray) in one colored image, no body can notice that there is a difference between the stego image and the original one, beside that the extracted secret images have high quality with high PSNR and Corr.

if some information arrives that shows there are secret images inside the stego one, here the robustness of the algorithm will prevent the intruder from extracting the images unless he knows the threshold for each sub band, the difference vector, the manipulation of the pixel's value and finally the way of dividing and transposing the image and it is impossible to know all of these.

REFERENCES

[1] Kipper, G. " Investigator's Guide to Steganography ", AUERBACH, 2003.

[2] Silman, J. "Steganography and Steganalysis: An Overview", *SANS Institute*, 2001.

[3] Jamil, T. "Steganography: The art of hiding information is plain sight", *IEEE* Potentials, 1999.

[4] Lee & L. H. Chen, Y. K. "High capacity image steganographic model", Visual Image Signal Processin*g*(147:03), June 2000.

[5] Huang, H. Y. "A lossless data hiding based on discrete haar wavelet transform", 2010 10th IEEE International Conference on Computer and D
[6] Jasim, A. A. "Gray Scale Image Hiding Using Wavelet Packet Transform", Iraq J. Electrical and Electronic Engineering(Vol.5 No.1) , 2009.
[7] Nag, A. S. Biswas, D. Sarkar and P. P. Sarkar, "A novel technique for image steganography based on Block-DCT and Huffman Encoding", International Journal of Computer Science and Information Technology(Volume 2, Number 3), June 2010.
[8] Lin and E. J. Delp E. T.,"Review of Data Hiding in Digital Images", Purdue university, west Lafayehe, Indiana, 1999.
[9] Keelan,B. W. "Handbook of image quality" , CRC Press, 2002.

Table(1) Pixel's number variation for embedding colored secret images pixels in one cover band

Size of secret images	The no. of pixels which will be embedded of each sub band of the secret image	The max. no. of pixels which will be embedded in each coefficient part of the cover
100*100	29700	442368
200*200	356400	442368
220*220	433620	442368

Table (2) Pixel's number variation for embedding gray-scale Secret images pixels in one cover band

Size of secret images	The no. of pixels which will be embedded of each sub band of the secret image	The max. no. of pixels which will be embedded in each coefficient part of the cover
100*100	89100	442368
200*200	356400	442368
300*300	433620	442368
380*380	433620	442368

Table (3) Results for embedding figure (6) secret's picture Into Lena cover image

	Lena+f84+86+f89 Size=512*512 Colored	Secret(f84) Gray_scale	Secret(f86) Gray_scale	Secret(f89)Gray_scale
Size of secret images=100*100				
PSNR (dB)	103.9936	70.0902	68.6269	68.7056
CORRELATION	1.000	0.9893	0.9848	0.9870
NAE	0.0003	0.0105	0.0131	0.0118
MD	0.0335	0.9961	0.9922	0.9961
Size of secret images=108*108				
PSNR(dB)	103.2061	Infinity	infinity	infinity
CORRELATION	1.00	1.000	1.000	1.000
NAE	0.0009	0	0	0
MD	0.0334	0	0	0
Size of secret images=200*200				
PSNR(dB)	96.4921	70.0853	68.6051	68.6917
CORRELATION	0.9999	0.9894	0.9848	0.9870
NAE	0.0034	0.0105	0.0131	0.0118
MD	0.0363	1	1	1
Size of secret images=225*225				
PSNR(dB)	95.2379	Infinity	infinity	infinity
CORRELATION	0.9999	1.000	1.000	1.000
NAE	0.0045	0	0	0
MD	0.0368	0	0	0
Size of secret images=300*300				
PSNR(dB)	92.5844	70.0847	68.5987	68.6985
CORRELATION	0.9999	0.9894	0.9848	0.9870
NAE	0.0082	0.0105	0.0131	0.0118
MD	0.0378	1	1	1
Size of secret images=360*360				
PSNR(dB)	88.8363	Infinity	infinity	infinity
CORRELATION	0.9997	1.000	1.000	1.000
NAE	0.0140	0	0	0
MD	0.0584	0	0	0
Size of secret images=370*370				
PSNR(dB)	87.4141	75.7326	73.8321	74.3167
CORRELATION	0.9995	0.9971	0.9955	0.9964
NAE	0.0161	0.0029	0.0037	0.0032
MD	0.0737	1	1	1
Size of secret images=380*380				
PSNR(dB)	84.6762	72.8512	71.0539	71.4285
CORRELATION	0.9992	0.9944	0.9914	0.9931
NAE	0.0202	0.0055	0.0072	0.0062
MD	0.1212	1	1	1

	baboon+f100+f102+f105 Size=512*512 Colored	Secret(f100) Gray_scale	Secret(f102)Gray_scale	Secret(f105)Gray_scale
Size of secret images=360*360				
PSNR(dB)	78.9585	infinity	infinity	infinity
CORRELATION	0.9970	1.000	1.000	1.000
NAE	0.0411	0	0	0
MD	0.1490	0	0	0
Size of secret images=380*380				
PSNR(dB)	75.3077	71.3609	74.9197	72.8616
CORRELATION	0.9932	0.9912	0.9947	0.9944
NAE	0.0593	0.0072	0.0055	0.0056
MD	0.2472	1	0.9686	0.8706

Table (4) Results for embedding figure (7) secret's picture into Lena cover image

	Lena+1c+2c+3c Size=512*512 Color cover	Secret(1c) Colored	Secret(2c) colored	Secret(3c)colored
Size of secret images=100*100				
PSNR(dB)	98.0585	69.9745	73.4080	70.2575
CORRELATION	1.000	0.9886	0.9910	0.9873
NAE	0.0024	0.0111	0.0096	0.0118
MD	0.0364	0.8322	0.7254	0.8930
Size of secret images=180*180				
PSNR(dB)	92.2101	infinity	infinity	infinity
CORRELATION	0.9998	1.0000	1.0000	1.0000
NAE	0.0090	0	0	0
MD	0.0376	0	0	0
Size of secret images=200*200				
PSNR(dB)	90.6800	69.9739	73.3906	70.2503
CORRELATION	0.9998	0.9886	0.9910	0.9873
NAE	0.0115	0.0111	0.0096	0.0118
MD	0.0419	0.8352	0.7607	0.8930

Size of secret images=210*210				
PSNR(dB)	88.4343	infinity	infinity	infinity
CORRELATION	0.9996	1.0000	1.0000	1.0000
NAE	0.0146	0	0	0
MD	0.0705	0	0	0
Size of secret images=220*220				
PSNR(dB)	84.2288	73.4009	76.8207	73.6730
CORRELATION	0.9991	0.9948	0.9959	0.9942
NAE	0.0209	0.0050	0.0043	0.0053
MD	0.1373	0.8361	0.7436	0.8930

Table(5) Results for embedding figure (8) secret's picture Into Baboon cover image

	baboon+f100+f102+f105 Size=512*512 Colored	Secret(f100) Gray_scale	Secret(f102) Gray_scale	Secret(f105) Gray_scale
Size of secret images=360*360				
PSNR(dB)	78.9585	infinity	infinity	infinity
CORRELATION	0.9970	1.000	1.000	1.000
NAE	0.0411	0	0	0
MD	0.1490	0	0	0
Size of secret images=380*380				
PSNR(dB)	75.3077	71.3609	74.9197	72.8616
CORRELATION	0.9932	0.9912	0.9947	0.9944
NAE	0.0593	0.0072	0.0055	0.0056
MD	0.2472	1	0.9686	0.8706

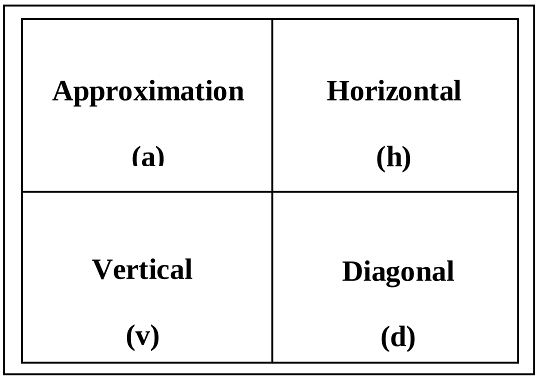
Table(6) Results for embedding figure (9) secret's picture into Lena cover image

	lena+cameraman+barbara+lena Size=512*512 Colored	Secret(cameraman) Gray_scale	Secret(barbara) Gray_scale	Secret(lena) Gray_scale
Size of secret images=360*360				
PSNR(dB)	88.8711	infinity	infinity	infinity
CORRELATION	0.9997	1.000	1.000	1.000
NAE	0.0140	0	0	0
MD	0.0581	0	0	0
Size of secret images=380*380				

PSNR(dB)	84.6923	76.5283	77.0598	76.0309
CORRELATION	0.9992	0.9948	0.9950	0.9940
NAE	0.0201	0.0058	0.0053	0.0055
MD	0.1200	0.6706	0.7412	0.8392

Table (7)Results for embedding figure (5) secret's picture into Clown cover image

	Clown+1c+2c+3c Size=512*512 Color cover	Secret(Peppers) colored	Secret(lena) colored	Secret(baboon)colored
Size of secret images=210*210				
PSNR(dB)	88.4476	infinity	infinity	infinity
CORRELATION	0.9996	1.0000	1.0000	1.0000
NAE	0.0154	0	0	0
MD	0.0728	0	0	0

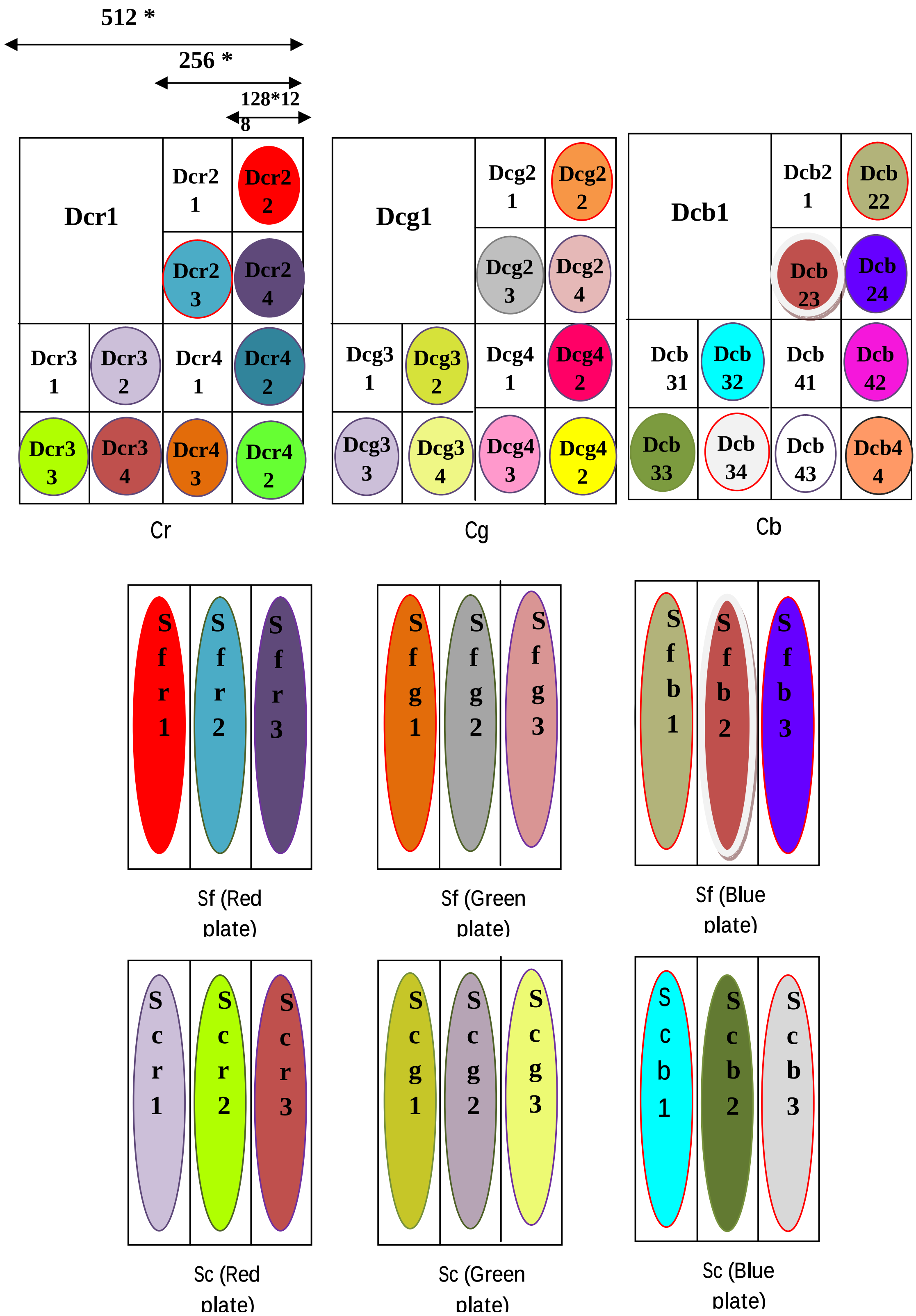


Figure(1) Level-1 2D_DWT representation of an image

(a)

(b)

Figure (2)An image with its DWT
(a) The image (b) level_1 decomposition for the cover image for red plate


512 *
256 *
128*128
Dcr1
Dcr21
Dcr22
Dcr23
Dcr24
Dcr31
Dcr32
Dcr33
Dcr34
Dcr41
Dcr42
Dcr43
Dcr42
Dcg1
Dcg21
Dcg22
Dcg23
Dcg24
Dcg31
Dcg32
Dcg33
Dcg34
Dcg41
Dcg42
Dcg43
Dcg42
Dcb1
Dcb21
Dcb22
Dcb23
Dcb24
Dcb31
Dcb32
Dcb33
Dcb34
Dcb41
Dcb42
Dcb43
Dcb44
Cr
Cg
Cb
Sfr1
Sfr2
Sfr3
Sf (Red plate)
Sfg1
Sfg2
Sfg3
Sf (Green plate)
Sfb1
Sfb2
Sfb3
Sf (Blue plate)
Scr1
Scr2
Scr3
Sc (Red plate)
Scg1
Scg2
Scg3
Sc (Green plate)
Scb1
Scb2
Scb3
Sc (Blue plate)

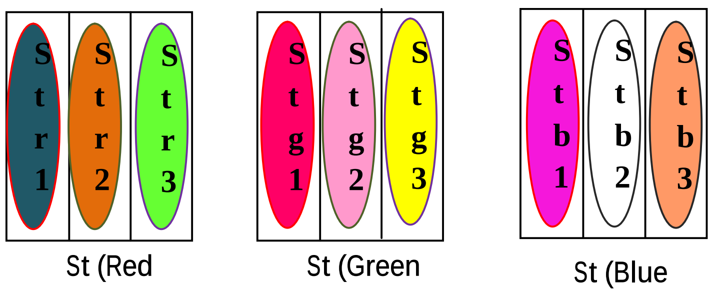
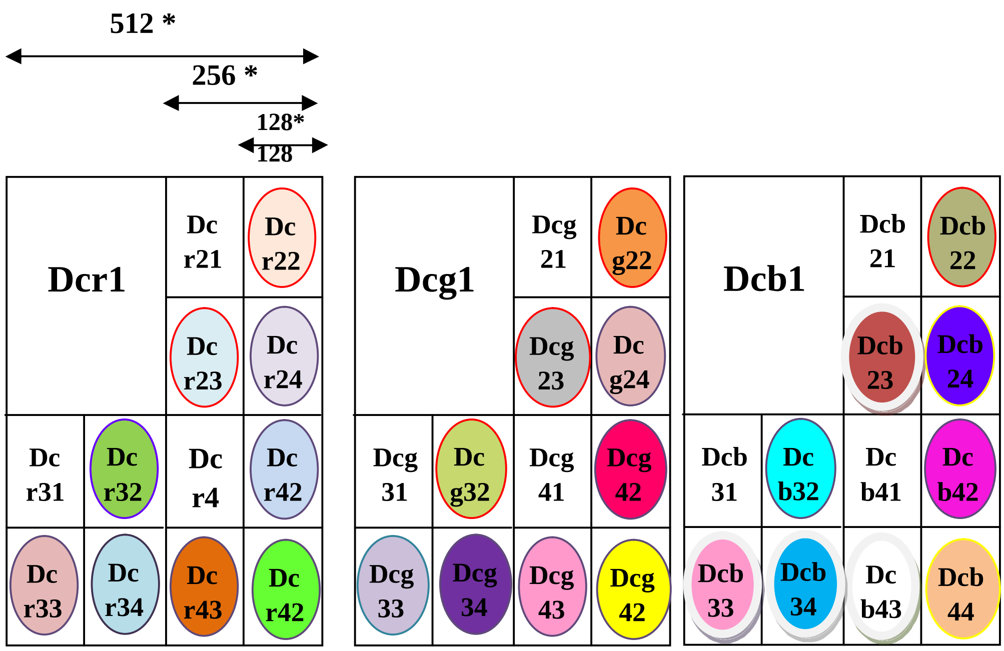
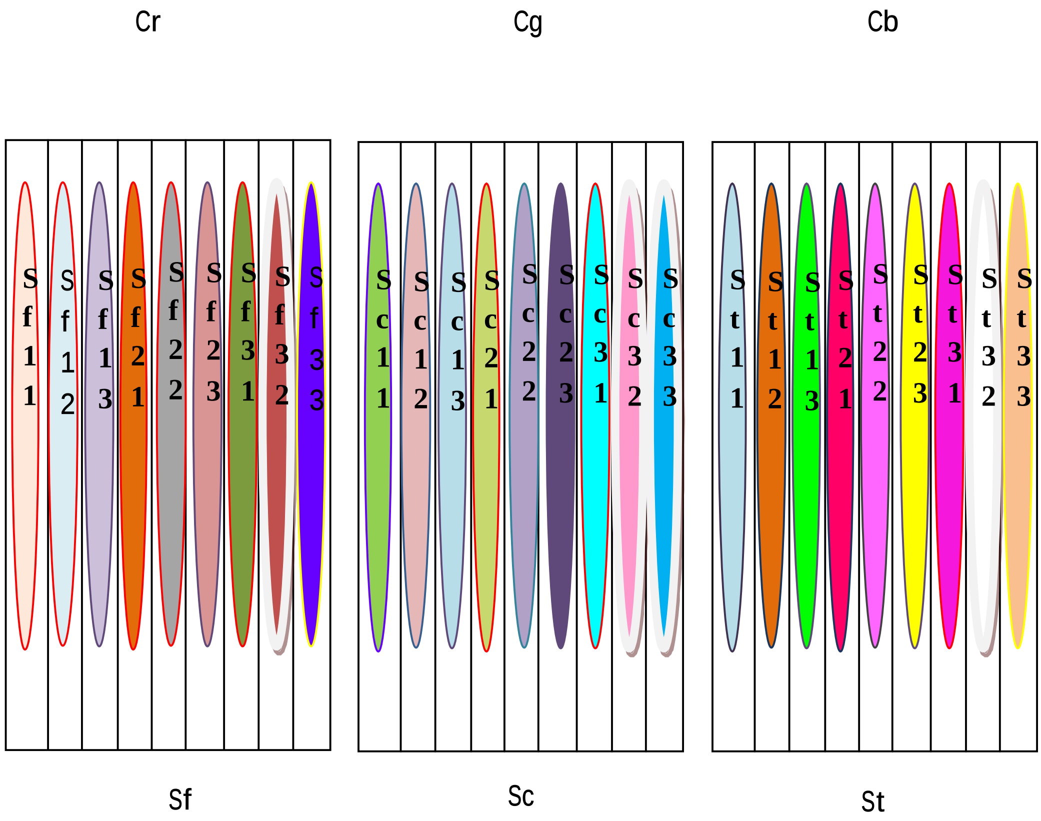


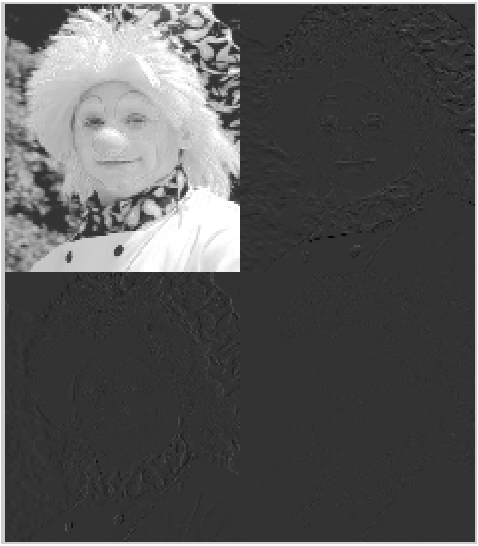
Figure (3) Schematic Diagram of Embedding Three Color Image In One Color Image





I … matic Diagra … ing Three Gra …
Secret Images In One Image

(a)Cover image



(b)level_1 decomposition for the cover image for red plate

(c)

level _2 decomposition for the coefficients of red plate

(d)

Stego image

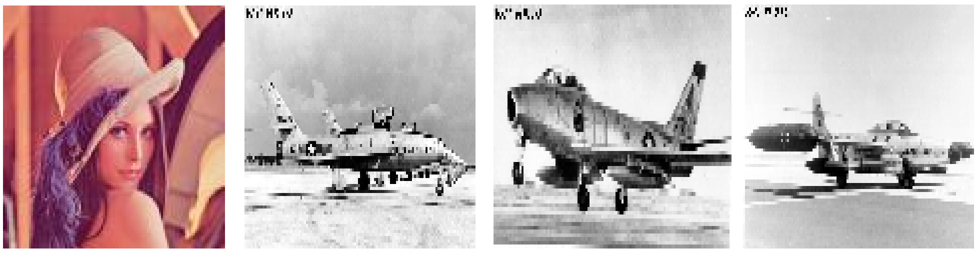
Figure (5) (a-d) Embedding three secret colored images in one color image

(e) Three secret colored images

(f) the three secret images after dividing and transposing

(g) the three extracted images

Figure (5) (e-g) Embedding three secret colored images in one colored image

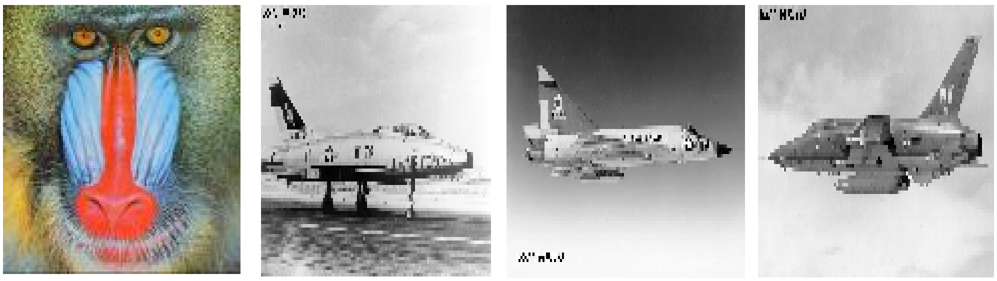


(a) Lena (Cover) (b) F84 (Secret 1) (c) F86 (Secret 2) (d) F89 (Secret3)

Figure (6) Embedding three gray-scale images into Lena cover image

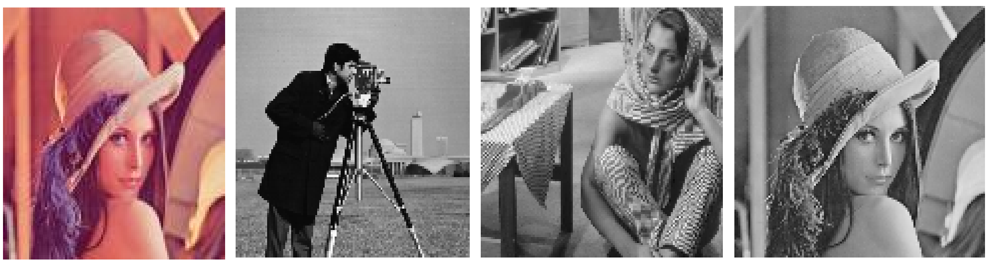
(a) Lena (Cover) (b) 1C (Secret 1) (c) 2C (Secret 2) (d) 3C (Secret3)

Figure (7) Embedding three colored-scales images into Lena cover image



(a) Baboon (b) F100 (Secret (c) F102 (Secret 2) (d) F105

Figure (8) Embedding three gray-scale images into Baboon cover image



(a) Lena (Cover) (b) Cam (Secret (c) Barbara (d) Lena (Secret3)

Figure (9) Embedding three gray-scale images into Lena cover image