CRYSTOMARKING

J SAI HARSHA

ACHARYA NAGARJUNA UNIVERSITY, GUNTUR, A.P, INDIA

Abstract—**We propose a novel approach for highly secure, stable and simplified communication system (Crystomarking).— In this paper we use cryptography, steganography and watermarking to bring a complex and secure system. Other main aim is to implement watermarking of several images under single cover.**

The paper also review the advantages of using Crystomarking and in particular watermarking of several (4) images under single cover reduces the bandwidth required to transmit the data.This paper describes the proposed algorithm, implementation and results of Crystomarking. Four images, a text message and single cover is used to test the system and the results are much efficient than existing systems.

Keywords—**RSA; LSB; DCT; DWT;**

I. INTRODUCTION

We are living in the era of information where billions of bits of data is created in every fraction of a second and with the advent of internet, creation and delivery of digital data (images, video and audio files, digital repositories and libraries, web publishing) has grown many fold. Since copying a digital data is very easy and fast too so, issues like, protection of rights of the content and proving ownership, arises.There are many encrypting systems to overcome copyright and also for security purpose.

The classification of encrypting system can be described by below Figure 1.

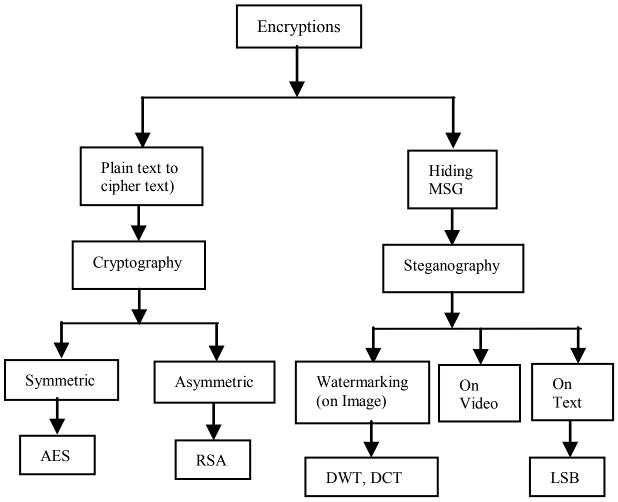

Fig. 1 Encryption Classification

For efficient protection against copyrights and also for secure and simple communication we proposed CRYSTOMARKING, it is a combination of Cryptography, Steganography, and Watermarking. This proposed system will satisfies most important aims of cyber world i.e., Security, Stability and Simplicity.

Security and Stability can be increased by complicating the encryption system. That is why Crystomarking is made so complex by using cryptography, steganography and watermarking together in encoding process. Stability can also be improved by changing the format of encrypted image into other format such as .mat, which cannot be accessed by others.

Simplicity means reducing the size of data that to be transmitted i.e. bandwidth conservation. In general to implement watermark on an image we need a cover (i.e. for single image we need single cover). If we want to implement watermark on two or more images we require two or more covers, this lead to increase the data size enormously. But in Crystomarking we use single cover for multiple images for watermarking, this in return reduces the size of datathat to be transmitted.

II. RSA CRYPTOGRAPHY

RSA cryptography is an asymmetric type of cryptography. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977.

A developer/user of RSA creates and then publishes a public key based on the two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message.

2.1 *Algorithm of Standard RSA*

1. Choose two very large random prime integers: p and q.
2. Compute n and φ(n): n = pq and φ(n) = (p-1)(q-1).

3. Choose an integer e, $1 < e < \varphi(n)$ such that: $\gcd(e, \varphi(n)) = 1$

(Where gcd means greatest common divisor)

4. Compute d, $1 < d < \varphi(n)$ such that: $ed(\bmod \varphi(n)) \equiv 1$

Where…

- The public key is (n, e) and the private key is (n, d)
- The values of p, q and φ(n) are private
- e is the public or encryption exponent
- d is the private or decryption exponent.

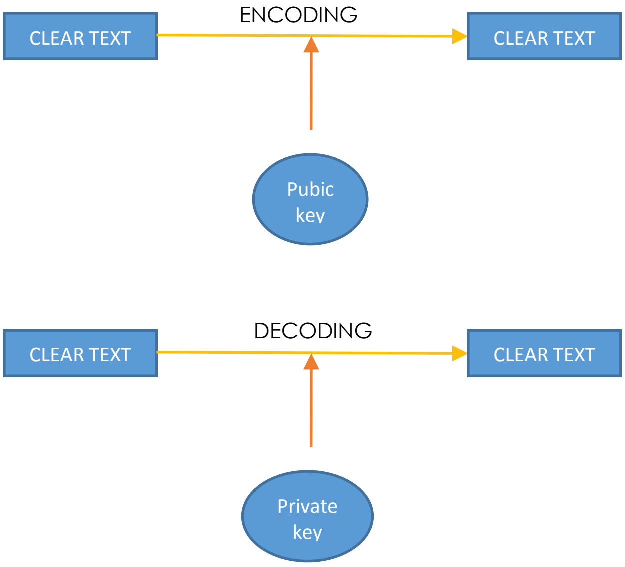

Fig. 2 RSA Block Diagram

Here private key != public key.

As RSA is asymmetric cryptography and it is also hard to decode this algorithm, this made me to choose RSA in my proposed system.

III. Steganography

The word steganography is derived from the Greek words stegos meaning cover and grafia meaning writing defining it as covered writing. **Steganography** is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message.

3.1 *LSB Steganography*

It is most popular and simplest method of steganography that is in practice. In LSB technique the least significant bits of each pixel arefliped by the message bits that to be encoded.

Suppose the first eight pixels of the original image have the following grayscale values:

11010010
01001010
10010111
10001100
00010101
01010111
00100110
01000011

To hide the letter C whose binary value is 10000011, we would replace the LSBs of these pixels to have the following new grayscale values:

11010011
01001010
10010110
10001100
00010100
01010110
00100111
01000011

Note that, on average, only half the LSBs need to change.

3.2 *LSB Embedding Algorithm*

The embedding process is as follows.

Inputs Cover image, stego-key and the text file
Output stego image

Steps:

1. Extract the pixels of the cover image.
2. Extract the characters of the text file.
3. Extract the characters from the Stego key.
4. Choose first pixel and pick characters of the Stego key and place it in first component of pixel.
5. Place some terminating symbol to indicate end of the key. 0 has been used as a terminating symbol in this algorithm.
6. Insert characters of text file in each first component of next pixels by replacing it.
7. Repeat step 6 till all the characters has been embedded.
8. Again place some terminating symbol to indicate end of data.
9. Obtained stego image.

Figure 3 shows the mechanism of LSB embedding technique.

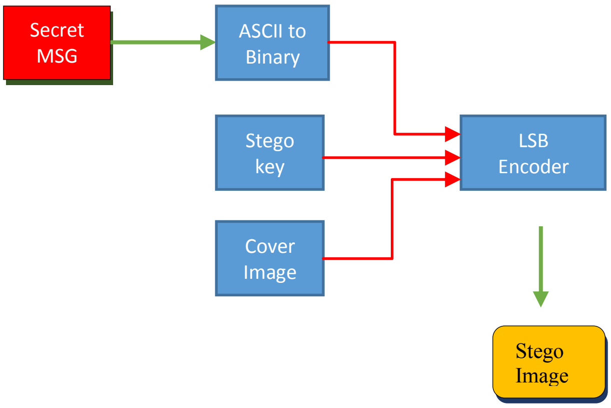

Fig. 3 LSB embedding mechanism

3.3 *LSB Extraction Algorithm*

The extraction process is as follows.

Inputs Stego-image file, stego-key
Output Secret text message.

Steps:

1. Extract the pixels of the stego image.
2. Now, start from first pixel and extract stego key characters from first component of the pixels. Follow Step3 up to terminating symbol, otherwise follow step 4.
4. If this extracted key matches with the key entered by the receiver, then follow Step 5, otherwise terminate the program.
5. If the key is correct, then go to next pixels and extract secret message characters from first component of next pixels. Follow Step 5 till up to terminating symbol, otherwise follow step 6.
6. Extract secret message.

Figure 4 shows LSB extraction mechanism technique

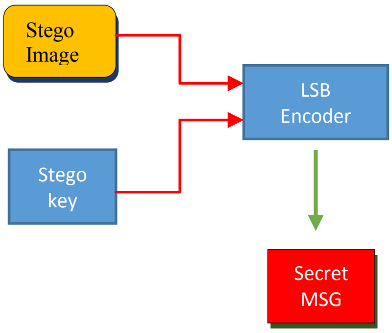

Fig. 4 LSB embedding mechanism

IV. EXISTINGWATERMARKINGSYSTEM (2 LEVEL DWT & DCT)

Presently for each secure image that to be transferred we use a cover single image even to transfer more images we required more cover images.

For each image we use watermarking, that watermarking involves three steps

1. Decomposition of colour image into its RGB channels.
2. Each colour channel is decomposed into three spatial components using DWT.
3. Applying DCT to convert these components into respective frequencies.

The input cover image is decomposed into its R, G and B colour channels. For each colour channel, spatial components (LH, HL, HH) are generated by using DWT transform. Further, frequency components can be generated by applying DCT to every spatial component. Among these frequencies, mid frequencies are used to embed the watermark. To enhance security, a code can be created in which the coefficients of one colour channel signify the indices of the other colour channel.

4.1 *Watermark Embedding Algorithm*

Figure 5 shows the block diagram of watermark embedding algorithm.

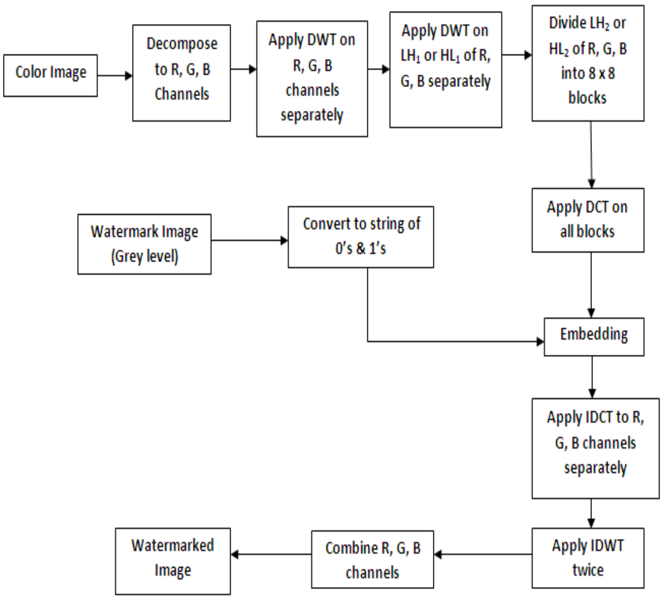
Color Image
Decompose to R, G, B Channels
Apply DWT on R, G, B channels separately
Apply DWT on LH_1 or HL_1 of R, G, B separately
Divide LH_2 or HL_2 of R, G, B into 8 x 8 blocks
Apply DCT on all blocks
Watermark Image (Grey level)
Convert to string of 0's & 1's
Embedding
Apply IDCT to R, G, B channels separately
Apply IDWT twice
Combine R, G, B channels
Watermarked Image

Fig. 5 Watermark embedding process

4.2 *Watermark Extraction Algorithm*

Figure 6 shows the block diagram of watermark extraction algorithm

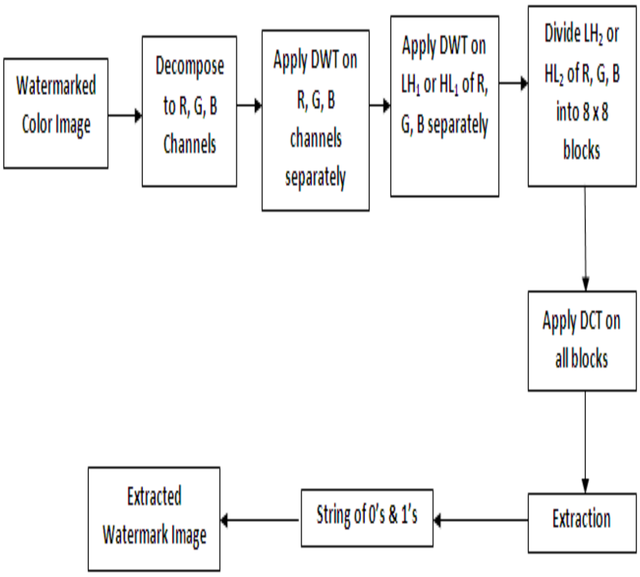
Watermarked Color Image
Decompose to R, G, B Channels
Apply DWT on R, G, B channels separately
Apply DWT on LH_1 or HL_1 of R, G, B separately
Divide LH_2 or HL_2 of R, G, B into 8 x 8 blocks
Apply DCT on all blocks
Extraction
String of 0's & 1's
Extracted Watermark Image

Fig. 6 Watermark embedding process

V. Proposed System& Experimental Results

Our proposed method will provide secure and efficient communication. This method uses single cover image to secure more confidential images.

In this project we considered four secure images and single text that is Crystomarked under single cover.

Our proposed method involves six steps

1. Apply RSA cryptography on text
2. Combine four images into one image.
3. Generate an integer from password entered.
4. Apply LSB steganography of text on combined image from the pixel at obtained integer.
5. Use 2 level DWT & DCT method of watermarking on blue component of cover.
6. Finally use LBS steganography to embed password and its length into cover.

5.1 *Appling RSA Cryptography on Text*

Use the standard RSA algorithm to encrypt the plain text message. The code for RSA encryption is as below.

```
e=dec2bin(e);
k = 65535;
c  = Msg;
cf = 1;
cf=mod(c*cf,n);
for i=k-1:-1:1
    c = mod(c*c,n);
    j=k-i+1;
if e(j)==1
         cf=mod(c*cf,n);
end
end
cipher=cf;
```

- *Experimental Inputs*

p = 11
q = 13

Considered message in this paper is

" Images A and B are taken by Mangalyan and Images C and D are taken by NASA rover "

- *Results*

Total	n	=	143
Public key	e	=	7
φ		=	120
private key	d	=	103

Output of RSA encryption for given input is

"S;&>PbAb;!dbBb;1>b□;D>!bybM;!&;y;!b;!dbS; &>PbYb;!dbb;1>b□;D>!b ybNAb1-O>1 "

5.2 *Considered Confidential Images*

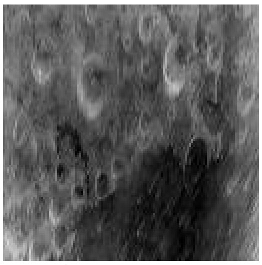

Fig. 7 Image A

Fig. 8 Image B

Fig. 9 Image C

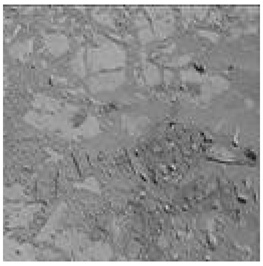

Fig. 10 Image D

5.3 *Combined Four Images*

1. Take four confidential images A, B, C and D.
2. Resize each image into 128*128 resolution.
3. Create a new image Z with size 256*256.
4. Replace the pixels of new Z image with pixels of four images, such that first pixel with A, next pixel with B repeat this process until all pixels in A and B are completed.
5. Now replace the remaining pixels of Z with pixels of C and D as in the point 4 until all pixels in C and D are completed.

```
size=128*128;
for i=1:size
    Z(2*i-1)=A(i);
    Z(2*i)=B(i);
end
for i=1:size
    Z(2*(i+size)-1)=C(i);
    Z(2*(size+i))=D(i);
End
```

The result of above algorithm is shown in figure 12.

Fig. 11Created Image Z

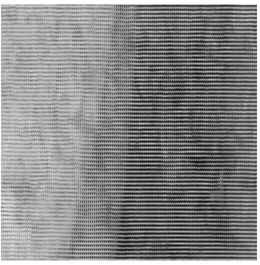

Fig. 12 Compound Image Z

5.4 *Generate an integer from Password entered*

Use the below logic to generate an integer using length of password and length of text message. This integer play a dominant role in providing security for encryption.

Let P = length of password;
L = length of text message;
Then logic is

```
Int1 = 8*L*P;
Integer = numel(cover) - Int1;
```

➢ *Experimental inputs and Results*

Password	-	***harsha***
Length of password (P)	-	***6***
Length of text message (L)	-	***80***

Int1	=	8 * 6 * 80	=	***3840***
numel (cover)	-	256*256	=	***4194304***
Integer			=	***4190464***

Without knowing the password on can't decode the encryption.

5.5 *Apply LSB Steganography in RSA encrypted Message*

Use the standard LSB Steganography as mention previously to embed RSA output (cipher text) on image Z.

Here Cover Image is compound image Z.
Text message is RSA output.
The code for LSB steganography is as below

```
for i=Integer:Integer+L
    pc=dec2bin(msg(i),8);
for j=1:8
        h=Z(j+i*8);
        a=dec2bin(h,8);
        a(8)=pc(j);
        r=bin2dec(a);
        Z(j+i*10)=r;
end
end
```

Result of LSB Steganography is shown in figure 14.

➢ *Experimental inputs and Results*

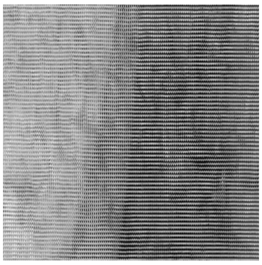

Fig. 13 Cover Image Z

+

S;&>PbAb;!db
Bb;1>b□;D>!by
bM;!&;y;!b;!db
S;&>PbYb;!dbb
;1>b□;D>!b

(Cipher text)

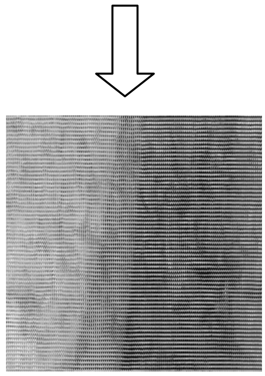

Fig. 14 Stego Image Z

5.6 *Apply 2 level DWT & DCT method of watermarking on blue component of cover*

Modify the existing watermark system as prescribed below.

A. *Watermark embedding over cover*

Step involved in watermark embedding are

1) Select any color image as cover image, denote it by 'I'. Obtain R, G and B channels of cover image 'I'.
2) Apply DWT to B channel separately to get the multi-resolution sub-bands LL_1, HL_1, LH_1, and HH_1.
3) Apply DWT again to HL_1 (or LH_1) sub-bands of R, G and B channels and select HL_2 (or LH_2)sub-bands of B channel.(Decomposition is continued only up to 2 levels as the energy becomes 0 at the third level.)
4) Divide the HL_2 (or LH_2) sub-bands of R, G and B channels into blocks of size 4X4.
5) Apply DCT to each of the blocks obtained in previous step.
6) Convert the watermark 'w' into string of 0's and 1's.
7) To embed bits, first calculate average of middle band coefficients of first block of B component.
8) Repeat this process for all bits of the watermark.
9) Apply IDCT to the blocks of B channels.
10) Apply IDWT for 2 levels to B channels.
11) Combine R, G and B channels to get watermarked image 'WI'

Modified watermark embedding process is as shown in figure 15 (Watermark Embedding in proposed system)

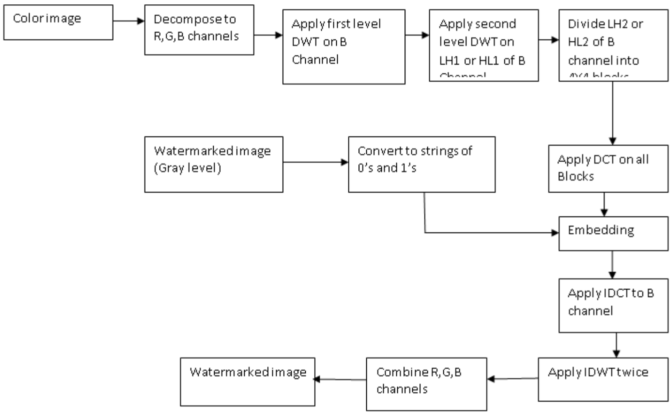
Color image
Decompose to R,G,B channels
Apply first level DWT on B Channel
Apply second level DWT on LH1 or HL1 of B Channel
Divide LH2 or HL2 of B channel into 4X4 blocks
Watermarked image (Gray level)
Convert to strings of 0's and 1's
Apply DCT on all Blocks
Embedding
Apply IDCT to B channel
Watermarked image
Combine R,G,B channels
Apply IDWT twice

Fig. 15 Watermark Embedding in proposed system

B. *Watermark Extraction*

Steps involved in watermark extraction from cover are

1) Obtain R, G and B channels of watermarked image ‘WI’.
2) Apply DWT to B channel to obtain the multi-resolution sub-bands LL_1, HL_1, LH_1, and HH_1.
3) Apply DWT again to HL_1 sub-bands of B channel and select HL_2 sub- B channel.
4) Divide the HL_2 sub-bands of B channel into blocks of size 4×4.
5) Apply DCT to B block obtained in previous step.
6) Water marking bits are extracted from first block of B channel and repeat this process until extraction of all bits from all 4X4 blocks
7) Then apply IDCT to all 4X4 blocks to get the watermarked image.

Modified watermark extraction process is as shown in figure 16 (Watermark Embedding in proposed system)

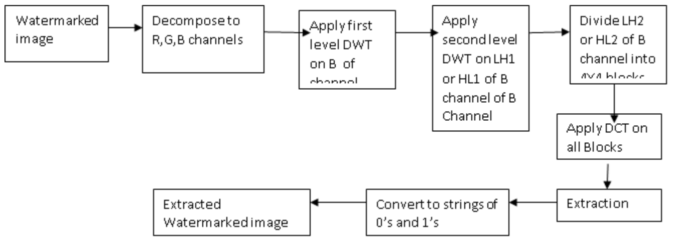
Watermarked image
Decompose to R,G,B channels
Apply first level DWT on B of channel
Apply second level DWT on LH1 or HL1 of B channel of B Channel
Divide LH2 or HL2 of B channel into 4X4 blocks
Apply DCT on all Blocks
Extraction
Convert to strings of 0’s and 1’s
Extracted Watermarked image

Fig. 16 Watermark Extraction in proposed system

➢ *Experimental inputs and Results*

Cover image and Stego image as Watermark image are shown in figures 17 and figure 18 respectively.

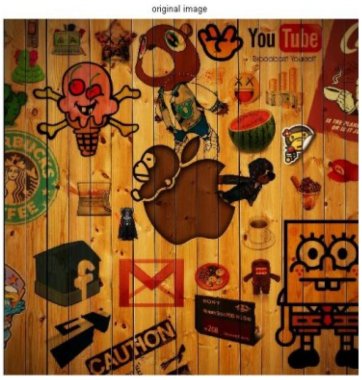

Fig. 17 Cover Image

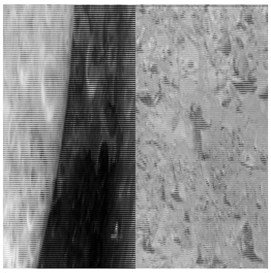

Fig. 18 Stego Image (Watermark)

Resultant of Watermarking is shown in figure 19

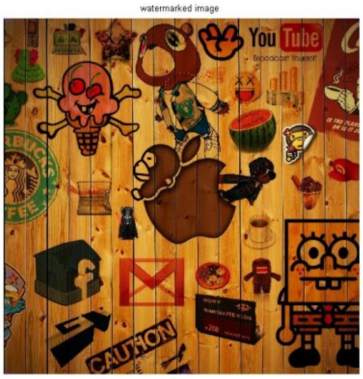

Fig. 19 Resultant of watermarking

5.7 *Finally use LBS steganography to embed password and its length into cover*

At final stage apply LSB steganography to embed password and its length on the watermarked cover image. The embedding procedure starts from pixel at the ‘Integer’ hat is obtained in previous step.

➢ *Experimental inputs and Results*

Password	-	harsha
Length of password (P)	-	6
Integer	-	4190464

Result of LSB Steganography on Watermarked image (Crystomarking output) is shown in the figure 20.

Fig. 20 Crystomarking output

This Crystomarked output can send to receiver through unreliable medium. The receiver should do the reverse process (decrypting) of encoding process to decode four confidential images and a text message by providing the same password (which is used in encryption).

5.8 *Decoding Process*

Steps:

1. Decode password by LSB Steganography decryption.
2. Watermark Extraction of Z image from cover.
3. Decode cipher text from Z image using LSB decryption from the pixel at ‘Integer’.
4. Use RSA decryption to decode text message from cipher text.

A. LSB decryption

LSB decryption is exactly the reverse process of LSB encryption. The code for LSB decryption is as below.

```
for k=Integer:Integer+L
      for j=1:8
            z=Z(j+k*8);
            z1=dec2bin(z,8);
            u(j)=z1(8);
end
        mssg(k)=bin2dec(u);
end
  msg=char(mssg);
```

B. RSA decryption

RSA decryption is exactly same as RSA encryption but with cipher text as input and by using private key instead of public key. The code for RSA decryption is as below.

```
d=dec2bin(d);
k = 65535;
c  = cipher;
cf = 1;
cf=mod(c*cf,n);
for i=k-1:-1:1
    c = mod(c*c,n);
    j=k-i+1;
if e(j)==1
          cf=mod(c*cf,n);
end
end
msg=cf;
```

C. Experimental Results

The decoding of Crystomarking is as shown in figures below.

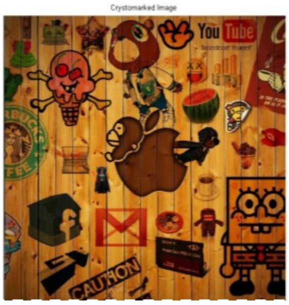

Fig. 21 Crystomarked image Received

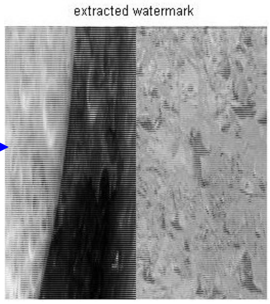

Fig. 22Extracted Watermark image Z

S;&>PbAb;!db
Bb;1>b□;D>!by
bM;!&;y;!b;!db
S;&>PbYb;!dbb
;1>b□;D>!b

[Cipher text]

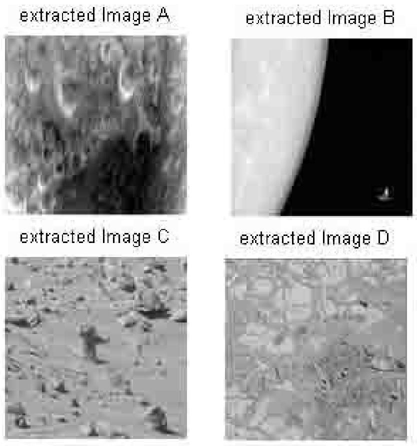

Fig. 21 Extracted confidential images A, B, C & D from Z

VI. COMPARISON OF RESULTS

In my proposed system we need single cover to transmit many images (4).

In this paper I used 4 images of each 120 KB, 2KB of text and a cover image of 1MB. The resultant output size is 1MB (Crystomarked output).

In general four images require 4 covers i.e. 4 images of each 120KB require 4MB of covers (4 covers of each 1MB).

Thus data is compressed logically which results in efficient communication.

TABLE 1: COMPARISON BETWEEN CRYSTOMARKING (PROPOSED SYSTEM) AND GENERAL WATERMARKING SYSTEM

	No of images to hide	Size of total image to hide	No of covers	Size of covers	Total size of data	Band width conserved
Crystomarking	4 (120KB each)	480 KB	1 (1 MB each)	1 MB	1504 KB	3072 KB
General Watermarking	4	480 KB	4	4 MB	4576 KB	0 KB

TABLE 2: COMPARISON BETWEEN ORIGINAL COVER AND WATERMARKED IMAGES

Mean Square Error	Peak Signal to Noise Ration	Ratio of Squared Norm	Normalized correlation	Standard Correlation
3.7443e-04 ~ 0	82.3971	1	1.0000	1.0000

VII. Functions Used For Proposed System & Execution Results

TABLE 3: Describes the MATLAB Function Used in the Proposed System and Its Execution Results.

Test Function	Expected Result	Observed Result	Pass/ Fail
Main	Loads four confidential images and a cover image	A,B,C,D & Cover images are loaded	Pass
RSA_encryption()	Loads text and encrypt into cipher text	Cipher text is produced	Pass
Combine4()	Combine 4 images in one	Image Z is created	Pass
Int_gen()	To generate an integer	Integer is generated	Pass
LSB_Stego_encrypt()	Embed the cipher message on image Z	LSB Stenography is applied	Pass
Watermark_encrypt()	Watermark Z on cover	2 level DWT & DCT is applied	Pass
Watermark_decrypt()	Extract Z image from Watermarked cover	Z image is extracted	Pass
LSB_Stego_decrypt()	Extract cipher text from Z	Cipher text is extracted	Pass
RSA_decyption()	Loads Cipher text and convert to plain text	Text message is extracted	Pass
Split4()	Extract 4 images from Z	A,B,C & D images are extracted	Pass

VIII. Conclusion and Future Work

This paper introduces the efficient and secure method of communication in unreliable environment that can replace the old techniques. This method is secure enough, reliable and simple to use. No need for special configuration (hardware) to implement it. Thus this system is most efficient, secure and stable communication system than any other.

This proposed system (CRYSTOMARKING) can be made more secure and efficient by adding best lossless compression algorithm and also by providing a biometric or face recognition as an authentication to access this algorithm.

Acknowledgments

I would like to acknowledge the help from all my classmates and faculty especially Dr. P. Siddaiah (Principal& H.O.D), Dr. E. Srinviasa Reddy (Dean), Er.S.Nagakishore Bhavanam (Asst Prof), Ms. Alekhya and Mr.Goutham Bommidi.

References

[1] Stefan Katzenbeisser and Fabien A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking," Artech house, Computer security series, pp. 15-23, 97-109, 2000.
[2] Neil F.Johnson, Zoran Duric and Sushil Jajodia, "Information Hiding, Steganography and Watermarking-Attacks and Counter Measures" Kluwer academic publisher, pp. 15-29, 2003.
[3] Vladimir Britanak, Patrick C. Yip and K.R. Rao, "Discrete Cosine and Sine Transforms", Boston: Academic Press, 2006.
[4] Gonzalez, Woods, and Eddins, "Digital image processing using MATLAB
[5] Guangmin Sun, Yao Yu, " DWT Based Watermarking Algorithm of Color Images", Second IEEE
[6] Conference on Industrial Electronics and Application",2007, PP 1823-1826.
[7] Bruce S, "Applied Cryptography", 2nd John Wiley and Sons, Inc. 1996
[8] Douglas R. Stinson "Cryptography Theory and Practice", Chapman & Hall/CRC Press, 3rd Edition, pp. 211-214, 2006
[9] Rivest R, Shamir A and Adelman L, "A Method for Obtaining Digital Signature and Public Key Cryptosystems", Communications of the ACM, 21, pp. 120-126, 1978
[10] Nan Li, "Research on Diffie – Hellman Key Exchange Protocol", IEEE 2nd International Conference on Computer Engineering and Technology, 2010, Volume 4, pp 634 – 637
[11] Xin Zhou, Xiaofei Tang, "Research and Implementation of RSA Algorithm for Encryption and Decryption", IEEE, 6th International Forum on Strategic Technology, pp- 1118 – 1121
[12] R . L. Rivest, A. Shamir and L. Adleman, "On Digital Signatures and Public Key Cryptosystems", Technical Memo 82, Laboratory for Computer Science, Massachusetts Institute of Technology, April 1970
[13] N . Provos, "Defending Against Statistical Steganography," Proc 10th USENEX Security Symposium 2005.
[14] Steven W. Smith , The Scientist and Engineer's Guide to Digital Signal Processing.M. G. J. Fridrich. Practical steganalysis of digital images - state of the art. Security and Watermarking of Multimedia Contents IV, 4675:1–13, 2002.

J. Sai Harsha is a Final Year BTech (ECE) student at**Acharya Nagarjuna University, Guntur (A.P), India**. He has published and presented several papers in different national and international conferences and journals. His research areas are Digital Image Processing, Network Security and Communications.