Final Report of Project

LSB Steganography

By: Swaroop Devaraju

Instructor: Dr. Belkasim

CSc6260

I. Abstract

In digital steganography, sensitive messages may be concealed by manipulating and storing information in the least significant bits of an image or a sound file. In the context of an image, if a user were to manipulate the last two bits of a color in a pixel, the value of the color would change at most +/- 3 value places, which is likely to be indistinguishable by the human eye. The user may later recover this information by extracting the least significant bits of the manipulated pixels to recover the original message.

II. Introduction

LSB can also stand for least significant byte. The meaning is parallel to the above: it is the byte (or octet) in that position of a multi-byte number which has the least potential value. If the abbreviation's meaning least significant byte isn't obvious from context, it should be stated explicitly to avoid confusion with least significant bit.

In computing, the least significant bit (LSB) is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The LSB is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digits further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position.

It is common to assign each bit a position number, ranging from zero to N-1, where N is the number of bits in the binary representation used. Normally, this is simply the exponent for the corresponding bit weight in base-2 (such as in $2^{31}..2^{0}$). Although a few CPU manufacturers assign bit numbers the opposite way (which is not the same as different endianness), the term least significant bit itself remains unambiguous as an alias for the unit bit.

By extension, the least significant bits (plural) are the bits of the number closest to, and including, the LSB.

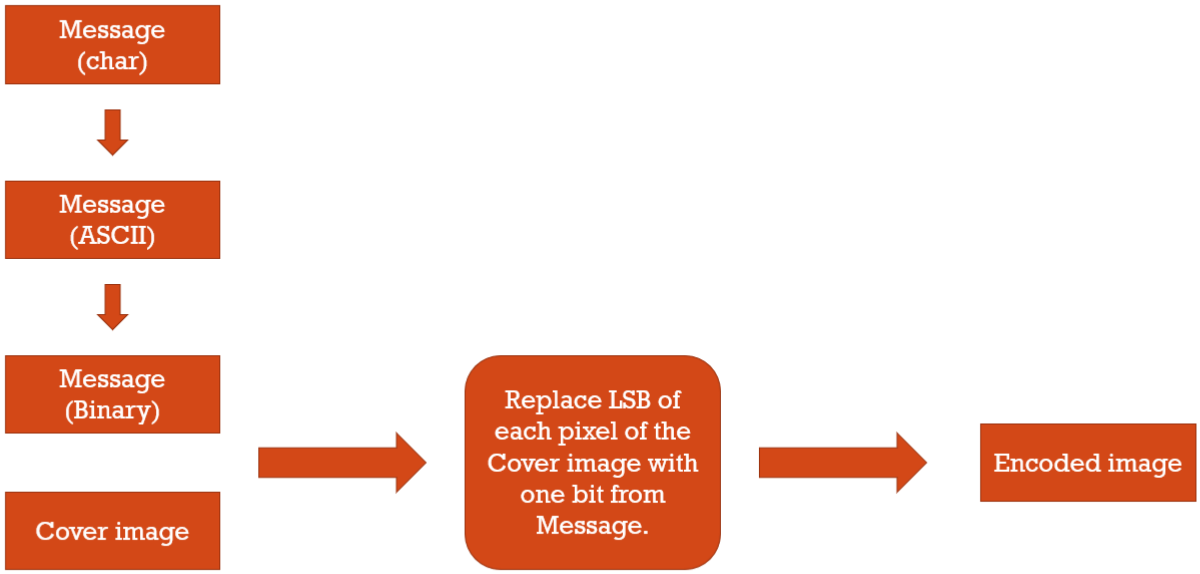
The least significant bits have the useful property of changing rapidly if the number changes even slightly. For example, if 1 (binary 00000001) is added to 3 (binary 00000011), the result will be 4 (binary 00000100) and three of the least significant bits will change (011 to 100). By contrast, the three most significant bits (MSBs) stay unchanged (000 to 000).

This table illustrates an example of decimal value of 149 and the location of LSB. In this particular example, the position of unit value (decimal 1 or 0) is located in bit position 0 (n=0). MSB stands for Most Significant Bit, while LSB stands for Least Significant Bit.

Binary (Decimal: 149)	1	0	0	1	0	1	0	**1**
Bit weight for given bit position n (2^n)	2^7	2^6	2^5	2^4	2^3	2^2	2^1	$\mathbf{2^0}$
Bit position label	MSB	___	___	___	___	___	___	**LSB**

III. LSB Steganography

- Encoding
  - Encoding is the process where the message is embedded inside the image.

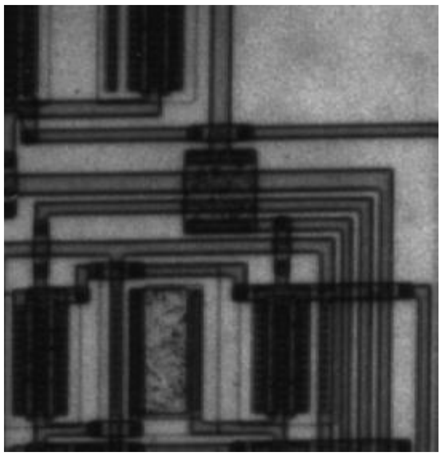


- Decoding
  - Decoding is a process where the message is retrieved from the encoded image.

IV. **Results and Conclusion**

I have used circuit.tif image as the cover image and a text file to retrieve the message to be encoded.

Cover Image



Message:
"CSC8221: Optical/Wireless networks
Homework 3

1. Compare the following concepts (RWA: Routing and Wavelength Assignment, RSA: Routing and Spectrum Allocation):
   a) RWA in in wavelength routed networks vs. RWA in waveband switching networks
   b) RSA in wavelength routed networks vs. RSA in FlexGrid networks

2. Compare the advantages and disadvantages of link protection, path protection and segment protection.

3. Comparing to Protection schemes, why Restoration schemes can be more efficient in terms of resource utilization.

4. Explain the basic idea of dedicated protection and shared protection.
   Compare dedicated link protection with dedicated path protection, which one is more practical and why?

5. In Optical Burst Switching (OBS) Networks, one key issue is to resolve the burst contention.
   (a) Explain when burst contention will happen.

(b) Provide at least 3 methods that can be employed to reduce data loss due to the burst contention, and briefly explain the idea of each method.

6. In OBS networks, the control packet is sent in advance before the corresponding data burst by an offset time.

Answer the following questions:

(a) Why is the offset time required?

(b) Assume we divide the burst into two classes: Class A, and Class B. We assign a larger offset time for the burst in Class A.

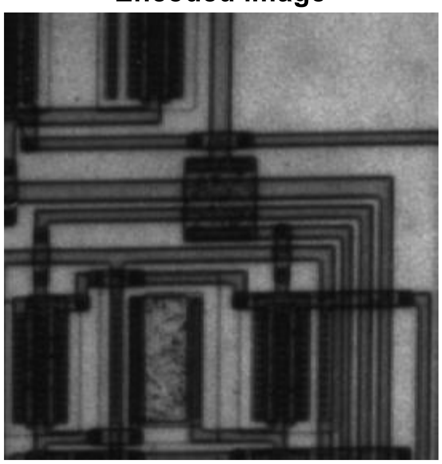
Explain why then the bursts in Class A have higher priority in the channel usage.

In other words, the bursts in Class A have less chance to be dropped due to contention.

7. (a). What is the maximum data rate that can be supported on a 10 MHz noiseless channel if the channel uses eight-level digital signals?

(b). What signal to noise ratio (in dB) is required to achieve 10 Mbps through a 5MHz channel?”

Encoded image



V. Future work

Future work will be an extension having one or more of the below:

- Hiding an image.
- Encrypt the message before embedding.
- Embed the message -
  - randomly.
  - in frequency domain.
- Embed multiple bits per pixel.

```
clear;

c = imread('circuit.tif');
figure;
imshow(c);
title('Cover Image');

fid =
fopen('C:\Users\swaro\Dropbox\sdevaraju1\Project\LSB_Stegan
ography\HW.txt','r');
F = fread(fid);
message = char(F);
fclose(fid);

message = strtrim(message);
m = length(message) * 8;
AsciiCode = uint8(message);
binaryString = transpose(dec2bin(AsciiCode,8));
binaryString = binaryString(:);
N = length(binaryString);
b = zeros(N,1);

for k = 1:N
    if(binaryString(k) == '1')
        b(k) = 1;
    else
        b(k) = 0;
    end
end

s = c;
height = size(c,1);
width = size(c,2);

%s(1,1) = uint8(m/8);
%s(1,2) = uint8(2);

k = 1;
for i = 1 : height
    for j = 3 : width
        if(k<=m)
            LSB = mod(double(s(i,j)),2);
            s(i,j)=s(i,j)+LSB+b(k);
            k=k+1;
```

```
        end

    end
end
imwrite(s,'output.png');

figure;
imshow(s);
title('Encoded image');




clear;

s = imread('output.png');
height = size(s,1);
width = size(s,2);

m = 14064;%double(
x = 2;

k = 1;
for i = 1 : height
    for j = 3 : width
        if (k <= m)
            b(k) = mod(double(s(i,j)),x);
            k = k + 1;
        end
    end
end
for k = 1 : m
     binaryVector(k) = b(k);
end
binValues = [ 128 64 32 16 8 4 2 1 ];
binaryVector = binaryVector(:);
if mod(length(binaryVector),8) ~= 0
    error('Length of binary vector must be a multiple of
8.');
end
binMatrix = reshape(binaryVector,8,m/8);
display(binMatrix);
textString = char(binValues*binMatrix);
disp(textString);
```