Digital Image Processing

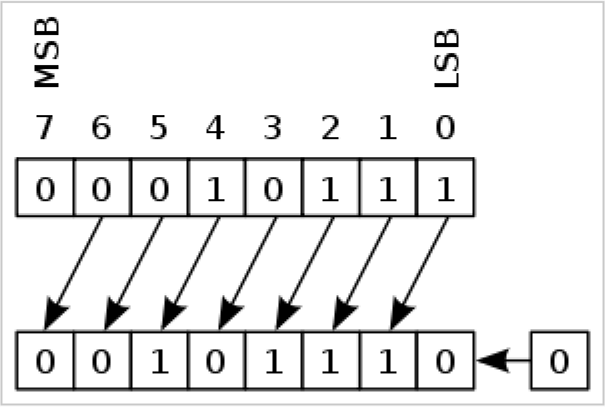
Lecture # 2C: Fundamentals
(Steganography)

Steganography

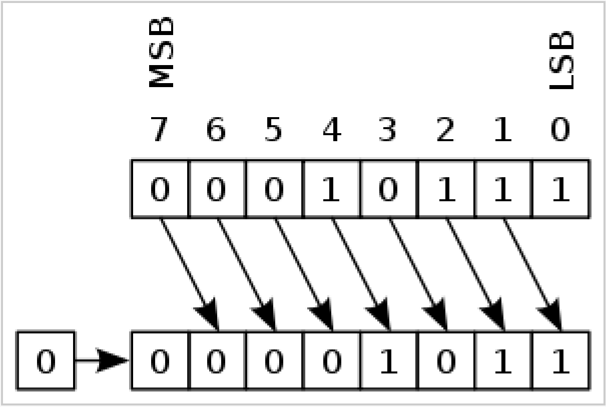
The science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message

Steganography

- Before Moving On ….
- Recall – Logical Shift Operators



Logical left shift one bit



Logical right shift one bit

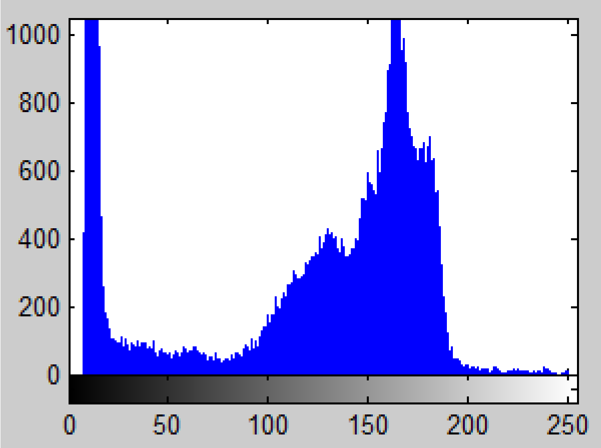
Steganography

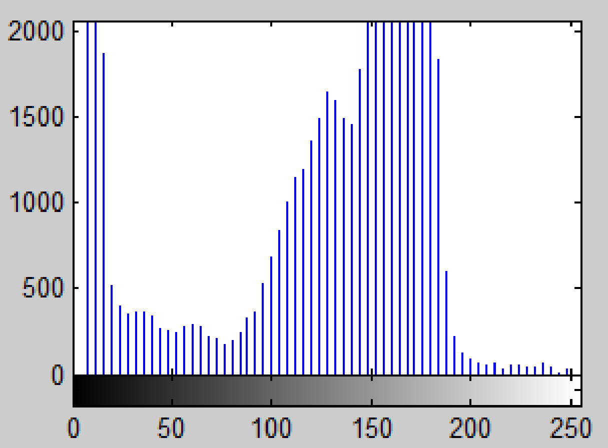
8-bit Image

Two least significant bits are 0

6-bit Image

If an image is quantized, say from 8 bits to 6 bits and redisplayed it can be all but impossible to tell the difference between the two.





Steganography

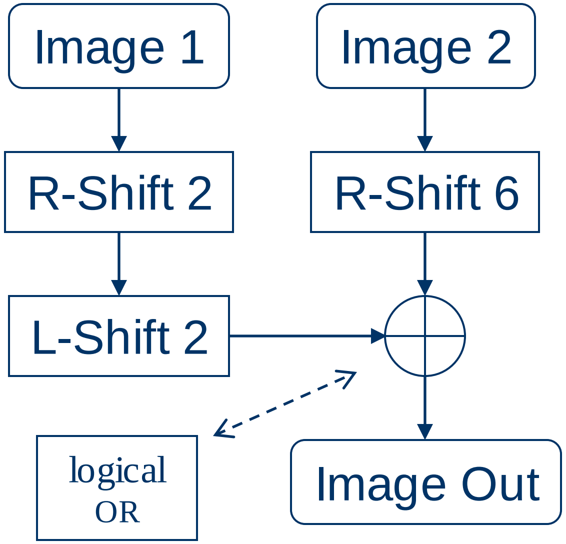
If the 6-bit version is displayed as an 8-bit image then the 8-bit pixels all have zeros in the lower 2 bits:

b	b	b	b	b	b	0	0

b = 0 or 1 always 0

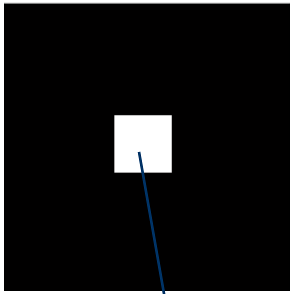
This introduces the possibility of encoding other information in the low-order bits.

That other information could be a message, perhaps encrypted, or even another image.



X-Shift *n* = logical left or right shift by *n* bits.

Steganography



182

1	0	1	1	0	1	1	0

R-Shift 2

0	0	1	0	1	1	0	1

L-Shift 2

180

1	0	1	1	0	1	0	0

220

1	1	0	1	1	1	0	0

R-Shift 6

03

0	0	0	0	0	0	1	1

$\oplus$

183

1	0	1	1	0	1	1	1

How to get the second image

183

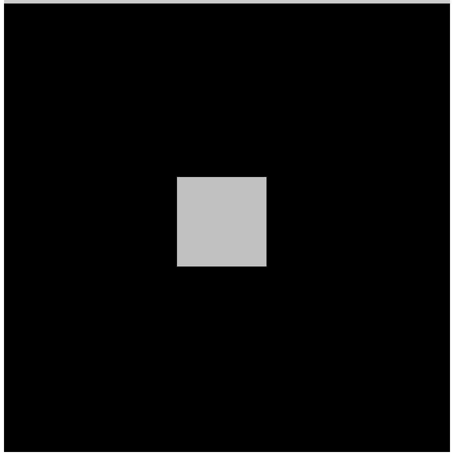
1	0	1	1	0	1	1	1

← L-Shift 6

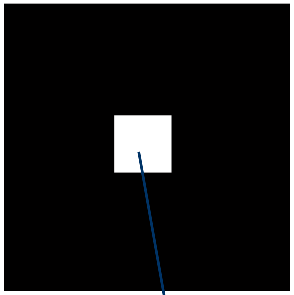
192

1	1	0	0	0	0	0	0

Extracted Second Image



Steganography



If we have only 4 colors (2-bits) and we put them in the lower order bits

182

1	0	1	1	0	1	1	0

R-Shift 2

0	0	1	0	1	1	0	1

L-Shift 2

180

1	0	1	1	0	1	0	0

03

0	0	0	0	0	0	1	1

~~R-Shift 6~~

~~03~~

~~0~~	~~0~~	~~0~~	~~0~~	~~0~~	~~0~~	~~1~~	~~1~~

No need to shift right

$\oplus$

183

1	0	1	1	0	1	1	1

What would you do to get back original data?

Steganography

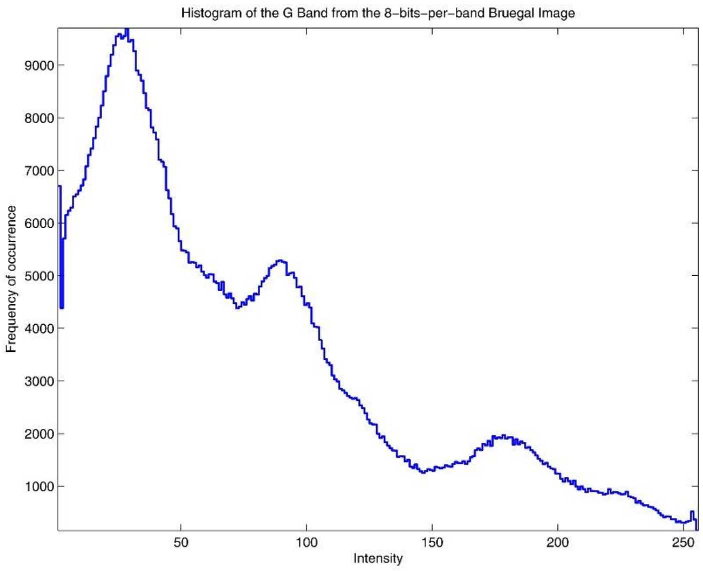
8-bit-per-band, 3-band, "original" image

Steganography

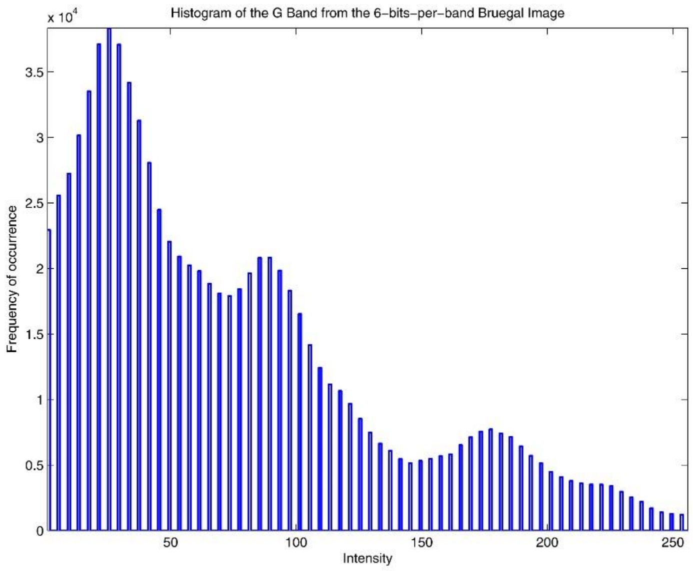
6-bit-per-band, 3-band, quantized image

Steganography

green-band histogram of 8-bit image



green-band histogram of 6-bit image

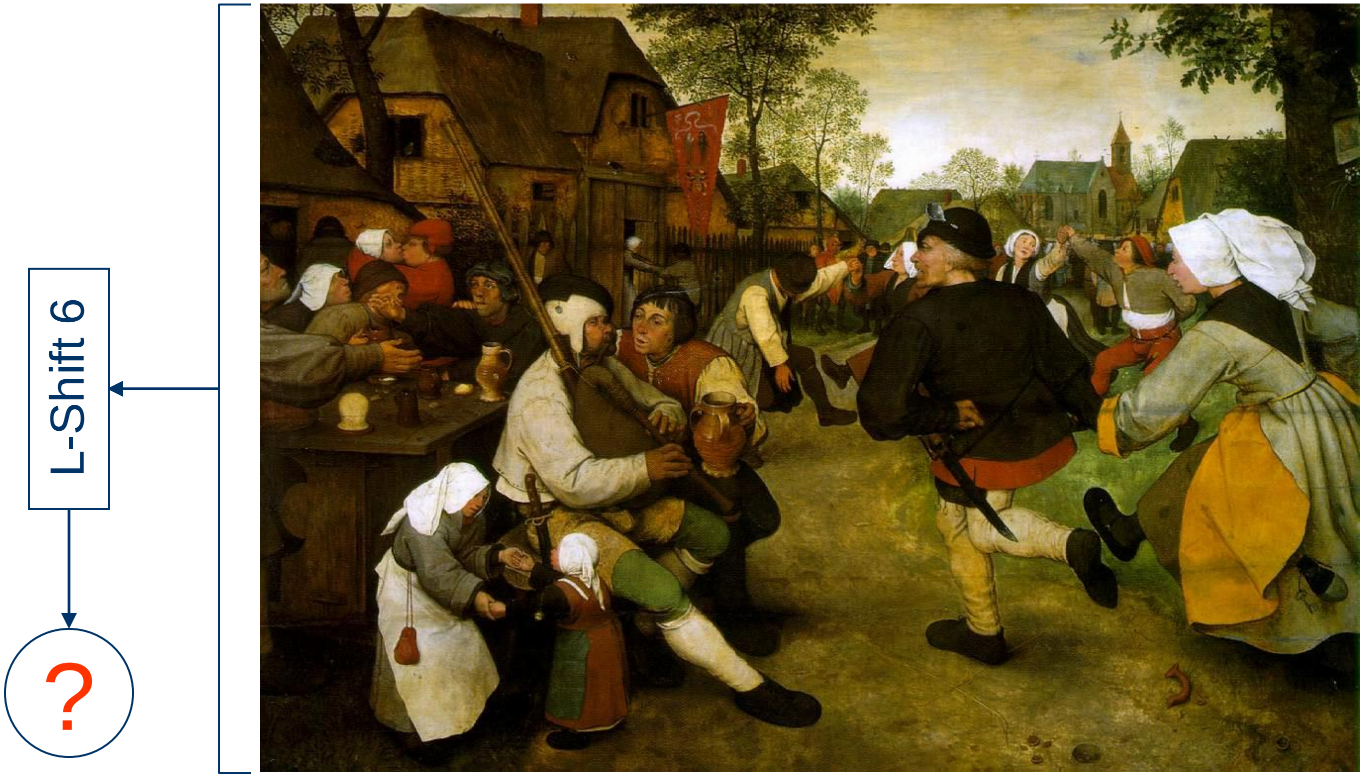


The histograms of the two versions indicate which is which. If the 6-bit version is displayed as an 8-bit image it has only pixels with values 0, 4, 8, … , 252.

Steganography

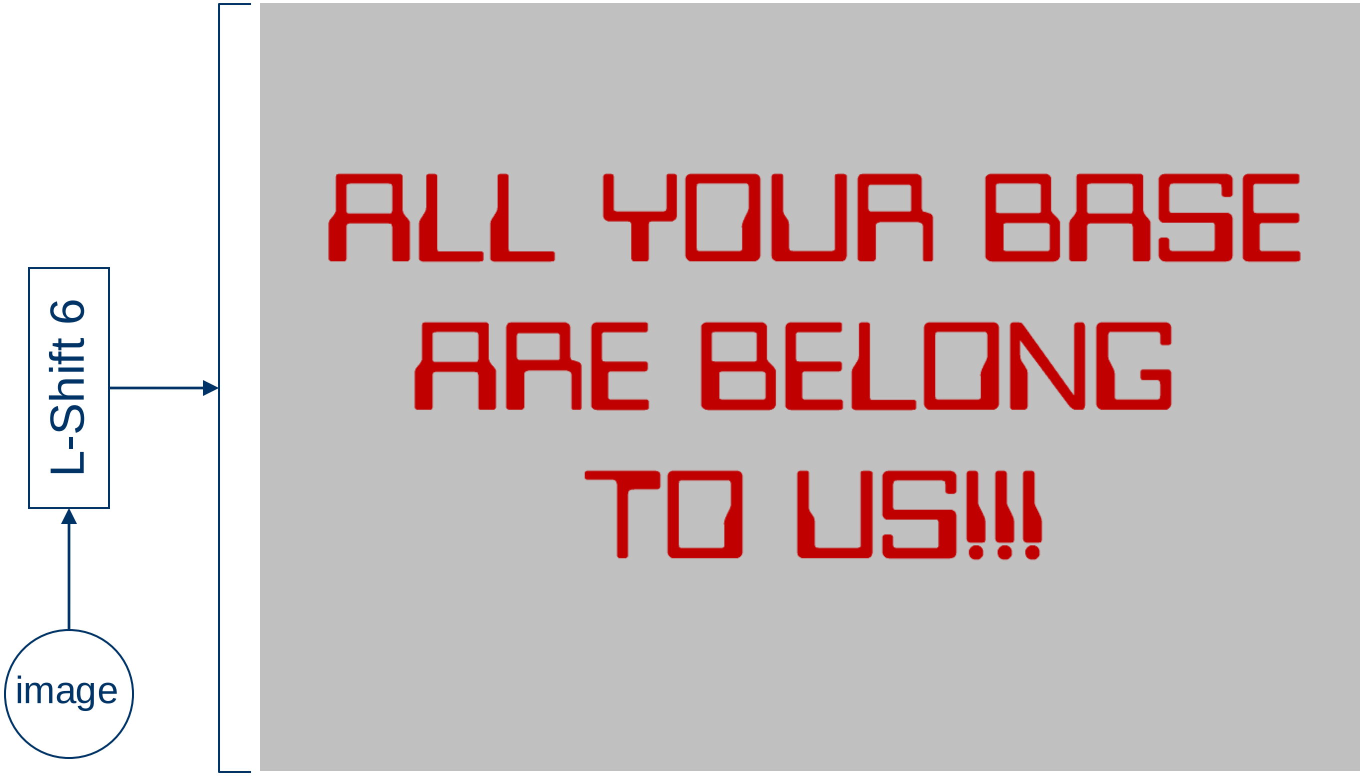
The second image is invisible because the value of each pixel is between 0 and 3. For any given pixel, its value is added to the to the collocated pixel in the first image that has a value from the set {0, 4, 8, … , 252}. The 2nd image is noise on the 1st.

Steganography



To recover the second image (which is 2 bits per pixel per band) simply left shift the combined image by 6 bits.

Steganography

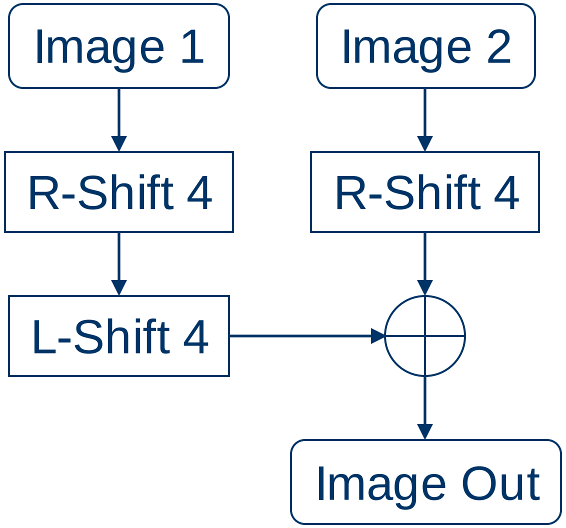


To recover the second image (which is 2 bits per pixel per band) simply left shift the combined image by 6 bits.

Steganography

This is so effective that two 4-bit-per-pixel images can be superimposed with only the image in the high-order bits visible. Both images contain the same amount of information but the image in the low-order bits is effectively invisible

Images 1 and 2 each have 4-bits per pixel when combined.

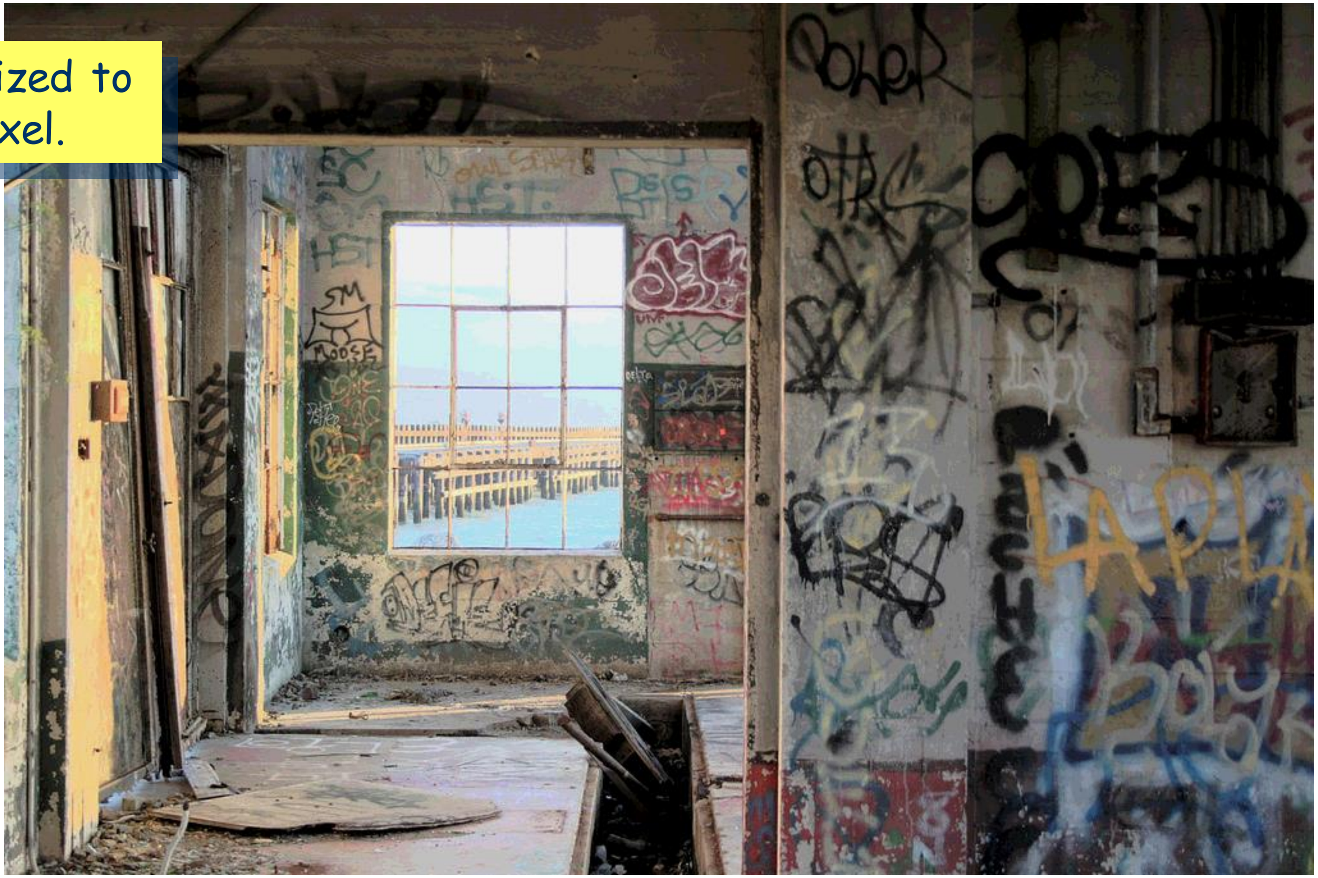


Steganography

Original Image

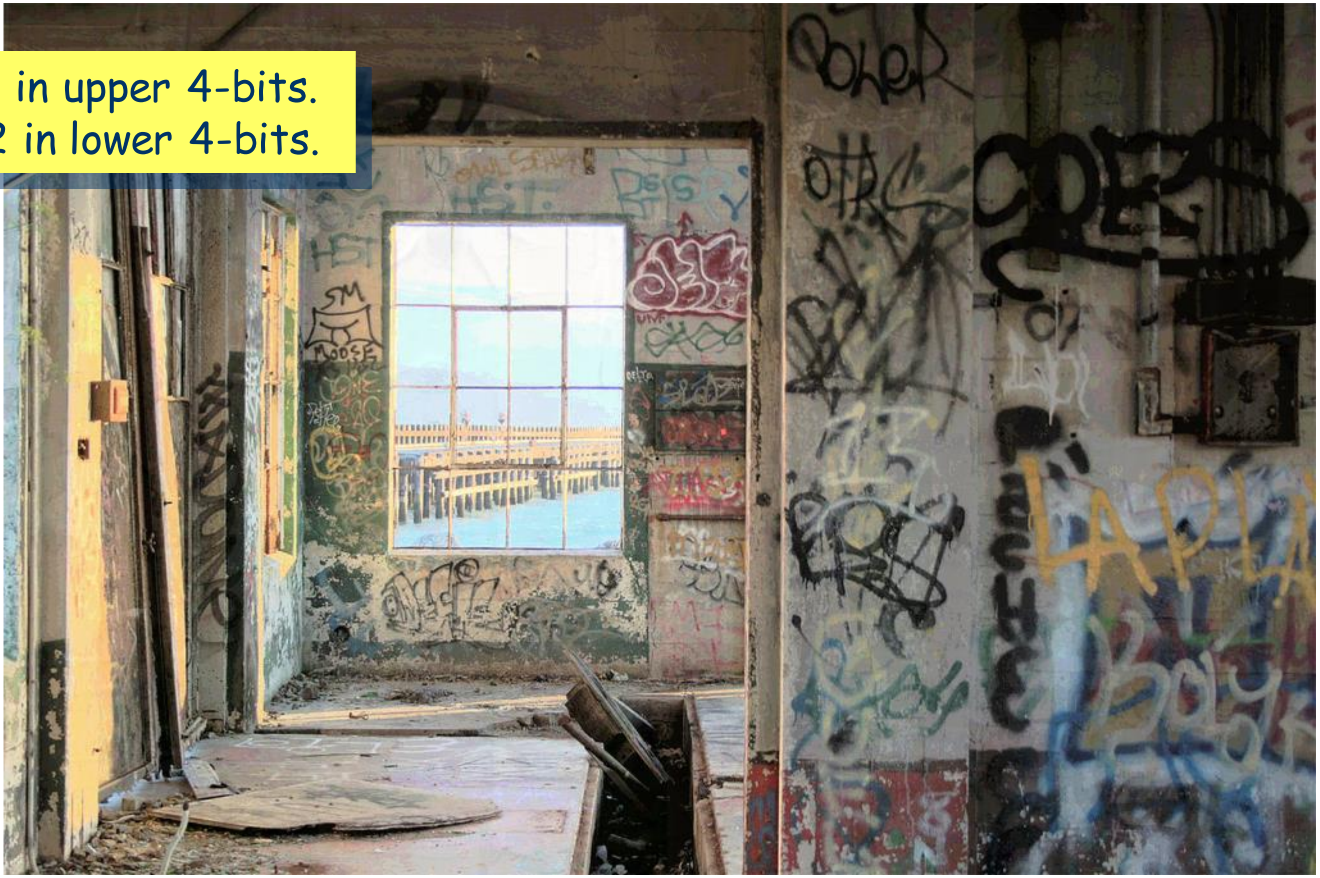
Steganography

Image quantized to 4-bits per pixel.



Steganography

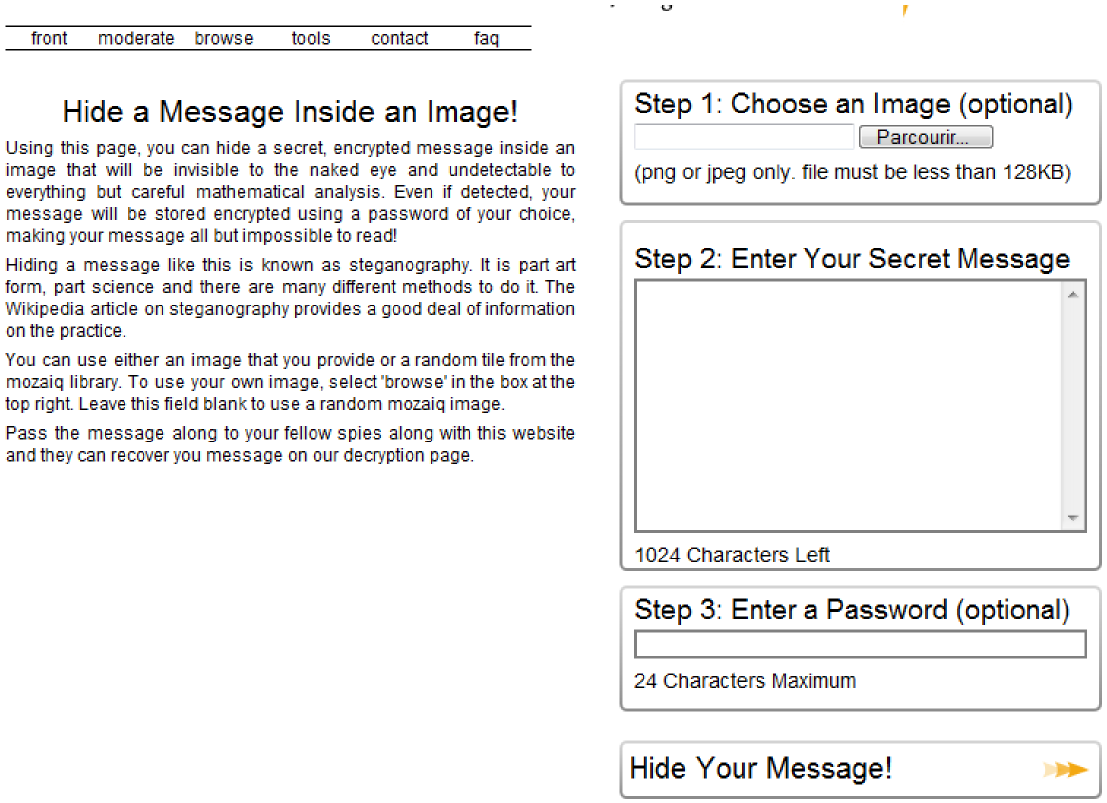
Image 1 in upper 4-bits.
Image 2 in lower 4-bits.



Steganography

Extracted Image

Steganography



Hide a Message Inside an Image!

Using this page, you can hide a secret, encrypted message inside an image that will be invisible to the naked eye and undetectable to everything but careful mathematical analysis. Even if detected, your message will be stored encrypted using a password of your choice, making your message all but impossible to read!

Hiding a message like this is known as steganography. It is part art form, part science and there are many different methods to do it. The Wikipedia article on steganography provides a good deal of information on the practice.

You can use either an image that you provide or a random tile from the mozaiq library. To use your own image, select 'browse' in the box at the top right. Leave this field blank to use a random mozaiq image.

Pass the message along to your fellow spies along with this website and they can recover you message on our decryption page.

Step 1: Choose an Image (optional)

(png or jpeg only. file must be less than 128KB)

Step 2: Enter Your Secret Message

1024 Characters Left

Step 3: Enter a Password (optional)

24 Characters Maximum

Hide Your Message!

http://mozaiq.org/encrypt/

Acknowledgements

- Digital Image Processing”, Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Peters, Richard Alan, II, Lectures on Image Processing, Vanderbilt University, Nashville, TN, April 2008
- Brian Mac Namee, Digitial Image Processing, School of Computing, Dublin Institute of Technology
- Computer Vision & Computer Graphics, Mark Borg

Material in these slides has been taken from the following resources