Lecture slides of the course

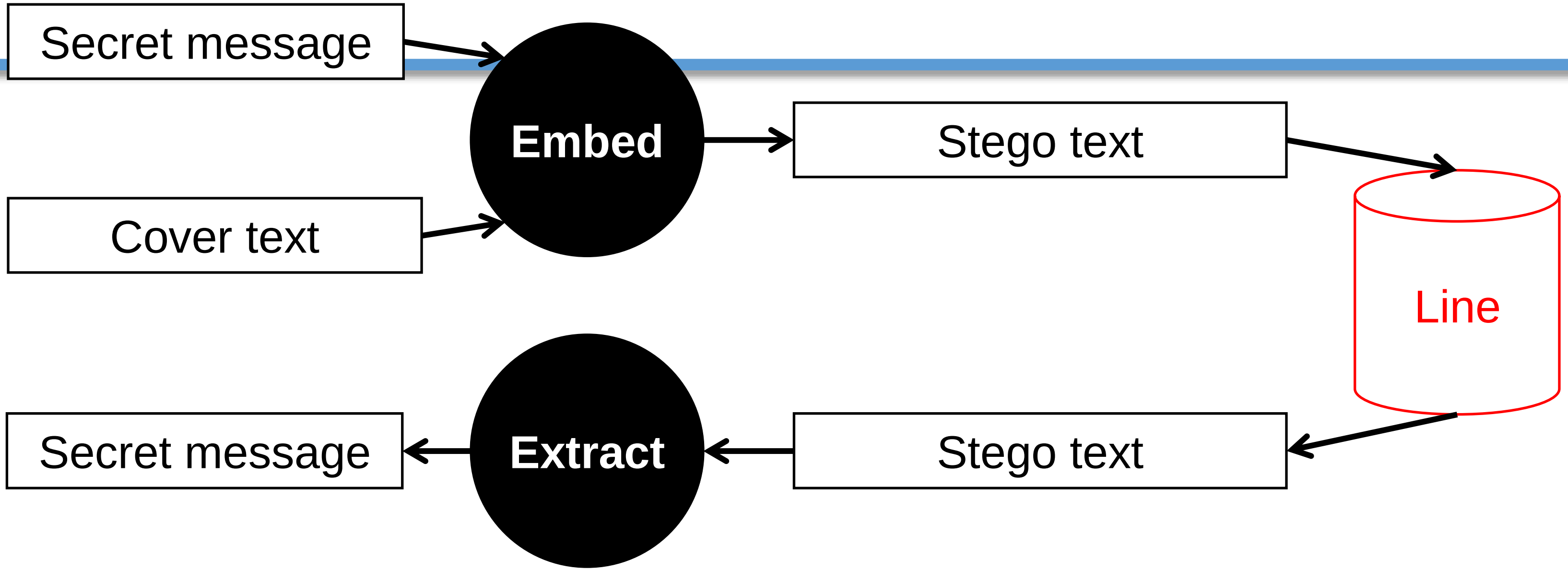
Information hiding & secret sharing

Image Steganography (P1)

Phạm Trọng Nghĩa

ptnghia@fit.hcmus.edu.vn

Last lecture



Secret message

Embed

Stego text

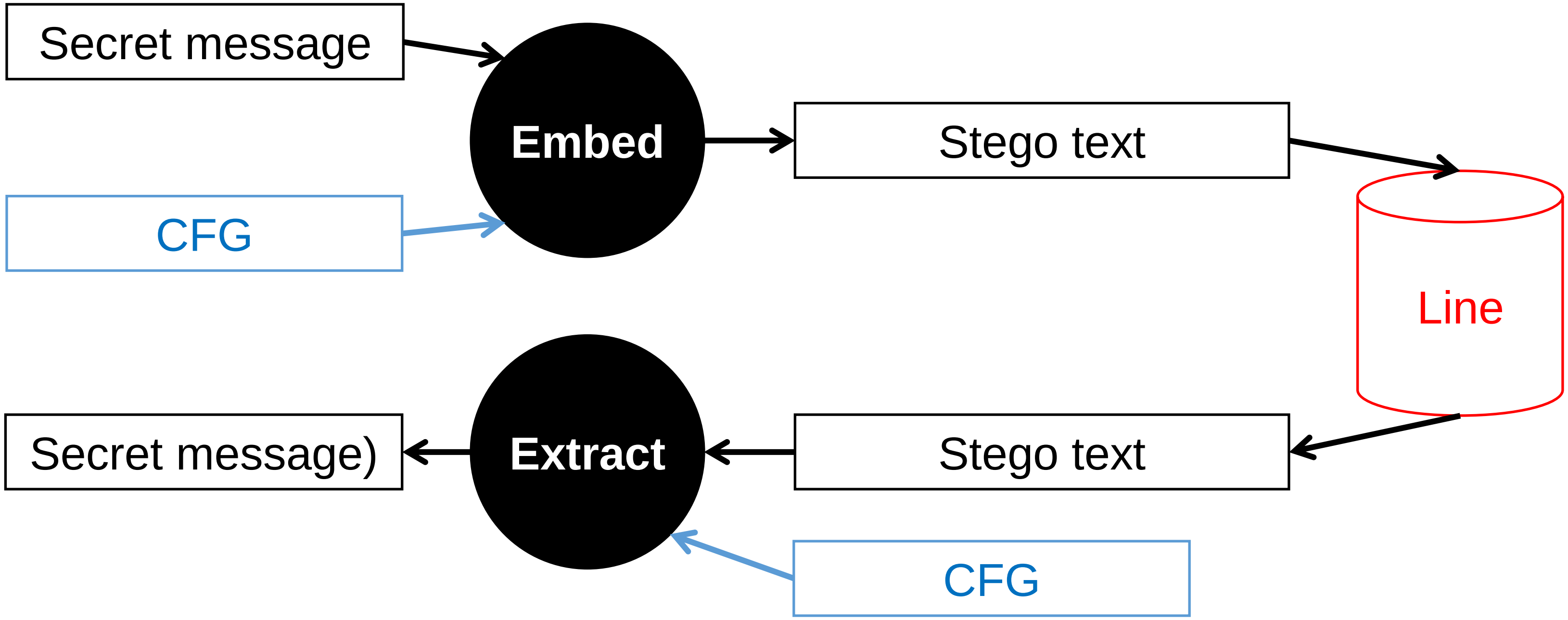
Cover text

Line

Secret message

Extract

Stego text



Secret message

Embed

Stego text

CFG

Line

Secret message)

Extract

Stego text

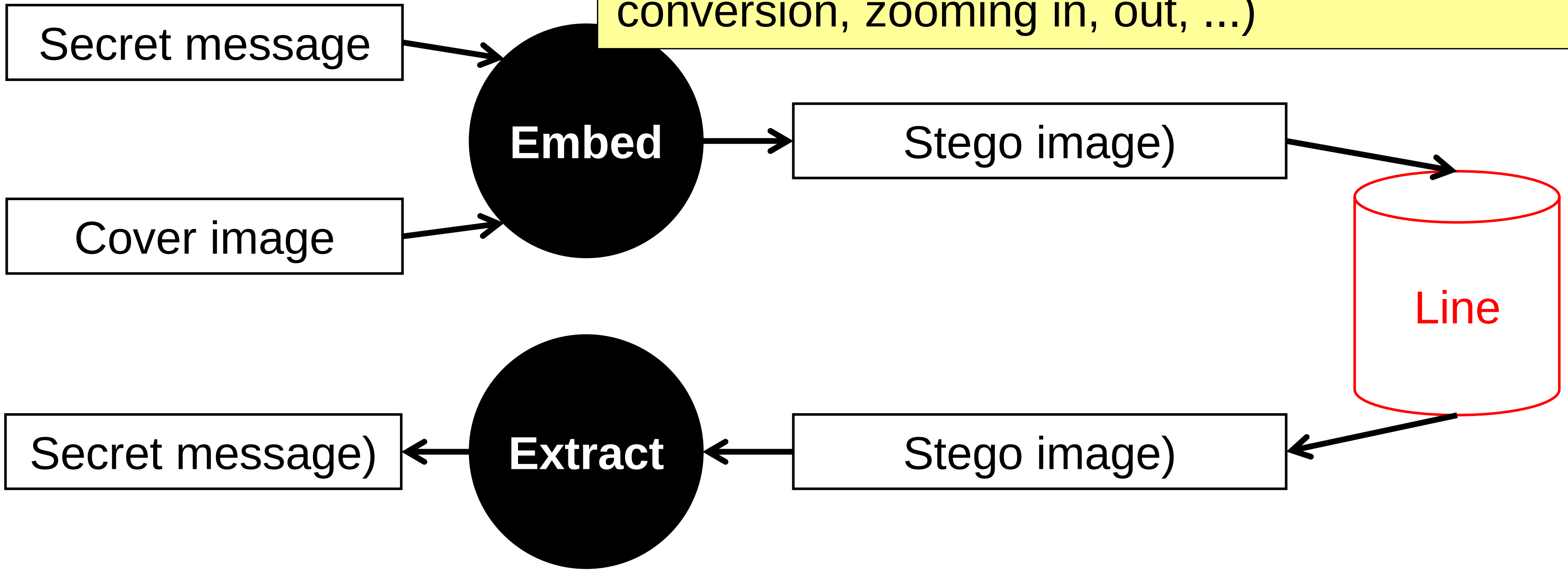
CFG

This lecture: hide secret information on images

- The problem of hiding secret information on photos
- LSB (Least Significant Bit) method – a simple method to hide secret information on images

Stagenography on Image

Desire:
Invisibility: it must be difficult for a third person to know the existence of secret information in stego image
In addition, there are other desires: **capacity**, robustness (secret information is not lost when there are editing operations on images: format conversion, zooming in, out, ...)



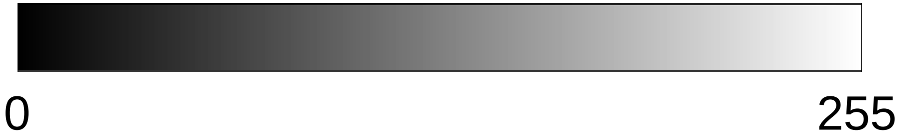
Stagenography on Image

Q: Where should the secret be embedded in the image?

A: First, need to understand about photos...

Simple image: grayscale

A pixel is represented by an 8-bit integer, ranging from 0-255 .



Where should we embed secret information on grayscale image?

- If we change the **LSB (Least Significant Bit)** of a pixel value, how much will this value change?
  - From -1 to 1
- With this change, will the human eye detect it?
  - Hard to detect ☺

→ Embed the secret bit in the LSB bit of the pixel value

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

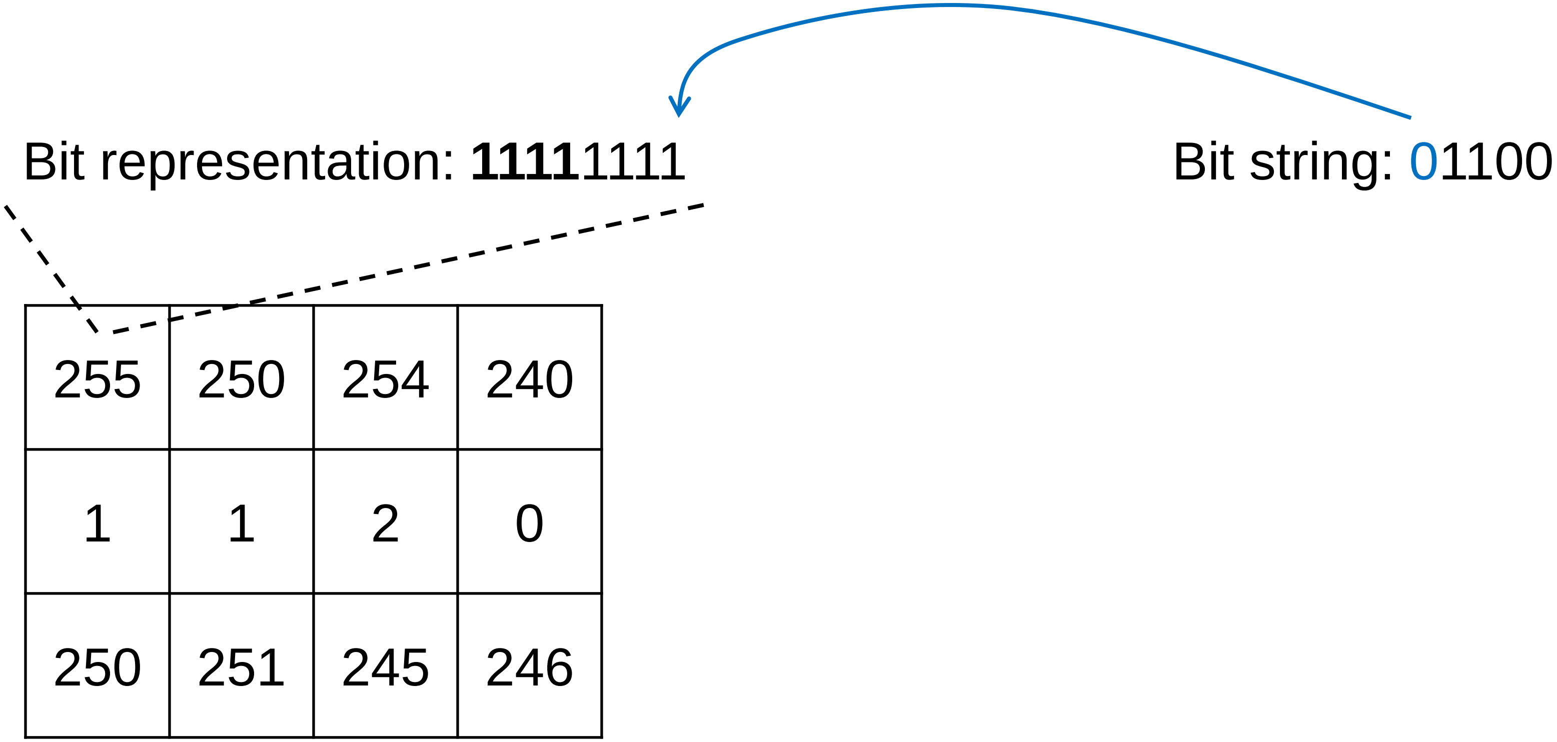
Bit representation: **1111**1111

Bit string: 01100

255	250	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below



Bit representation: **1111**1111

Bit string: 01100

255	250	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

Bit representation: **1111**1111

Bit string: 01100

255	250	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

Bit representation: **1111**1110

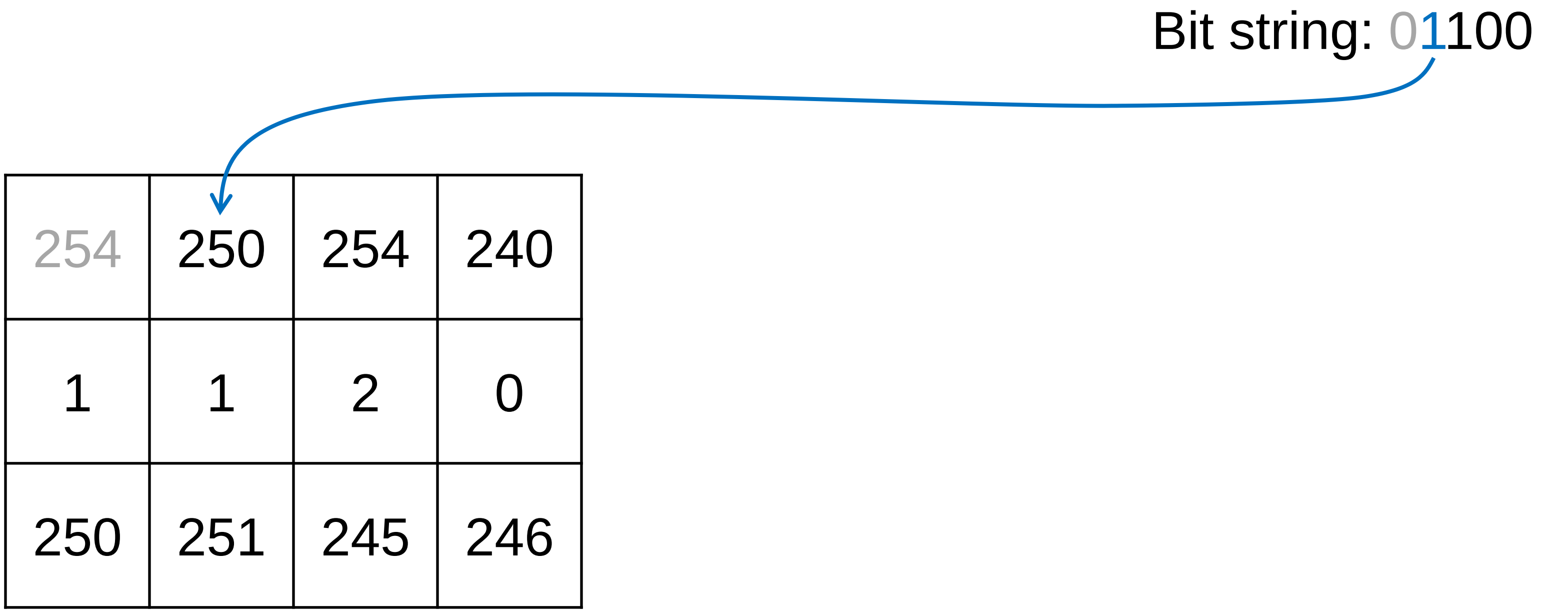
Bit string: 01100

254			
1	1	2	0
250	251	245	246

$$\text{Value after embedding} = \left\lfloor \frac{\text{Original value}}{2} \right\rfloor \times 2 + \text{Embedded bit}$$

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

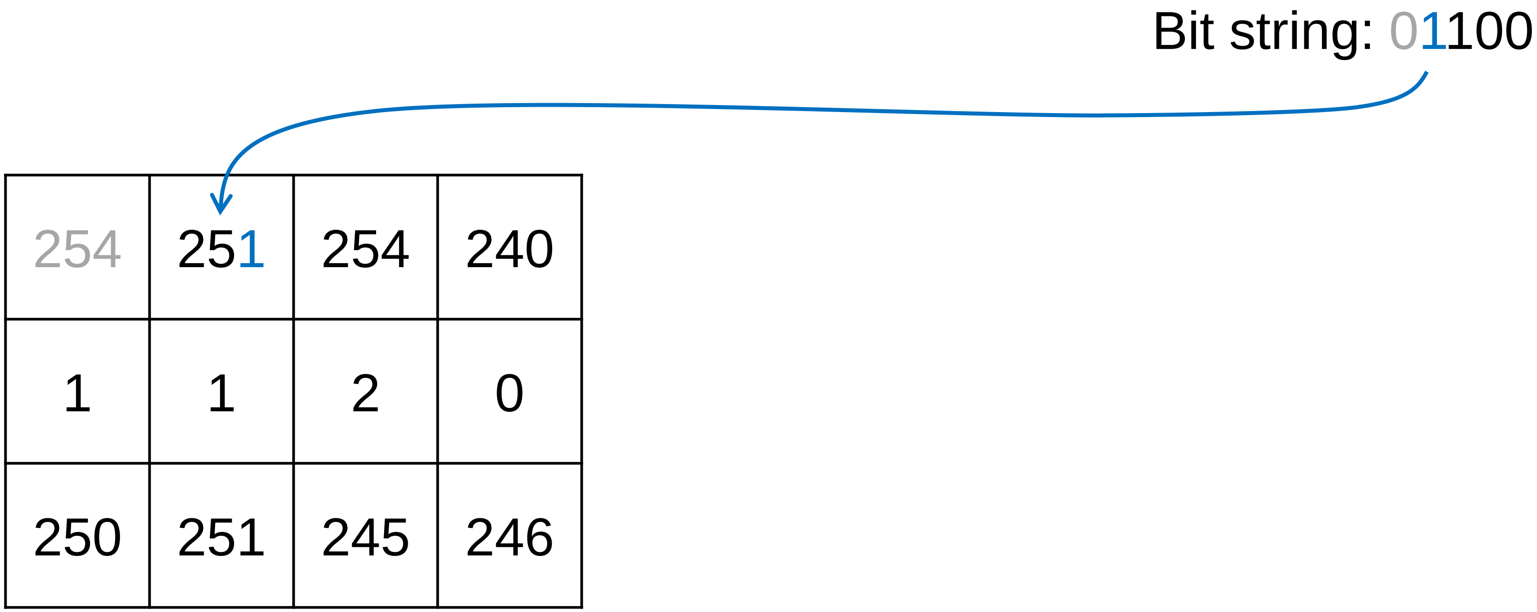


Bit string: 01100

254	250	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below



Bit string: 01100

254	251	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

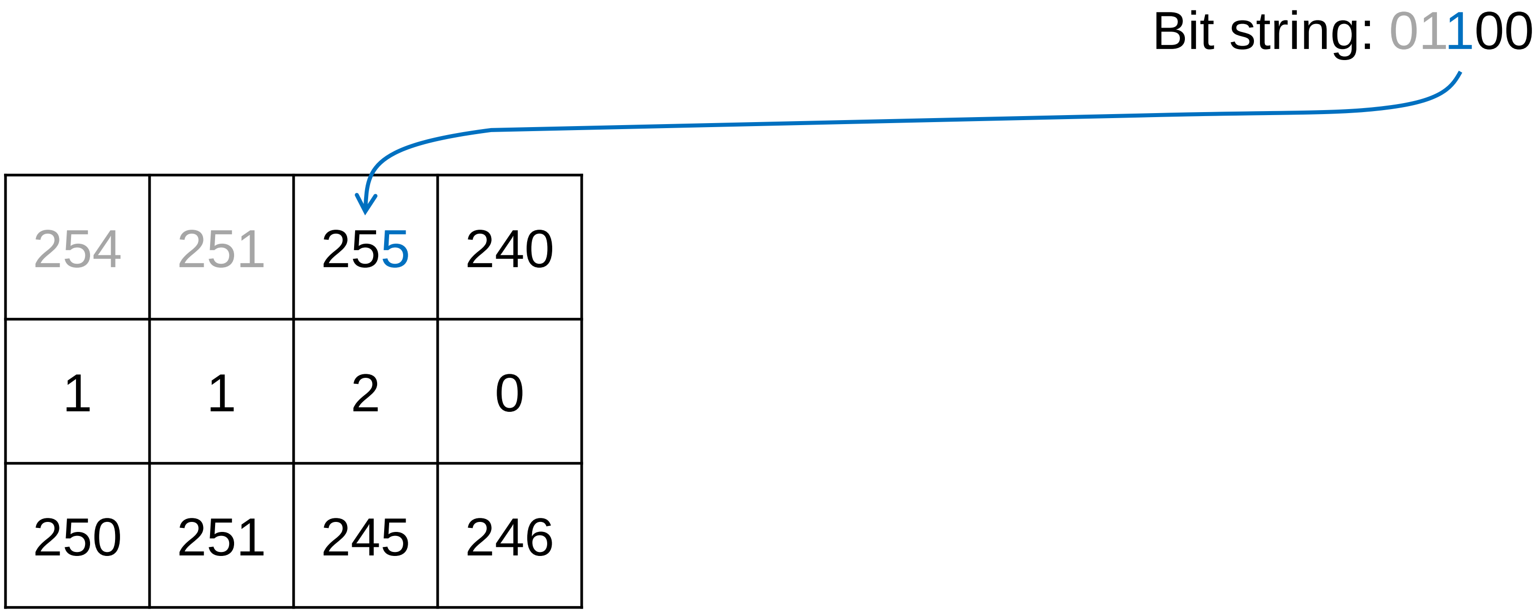
Example 1: embed secret bit string 01100 in the image below

Bit string: 01100

254	251	254	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

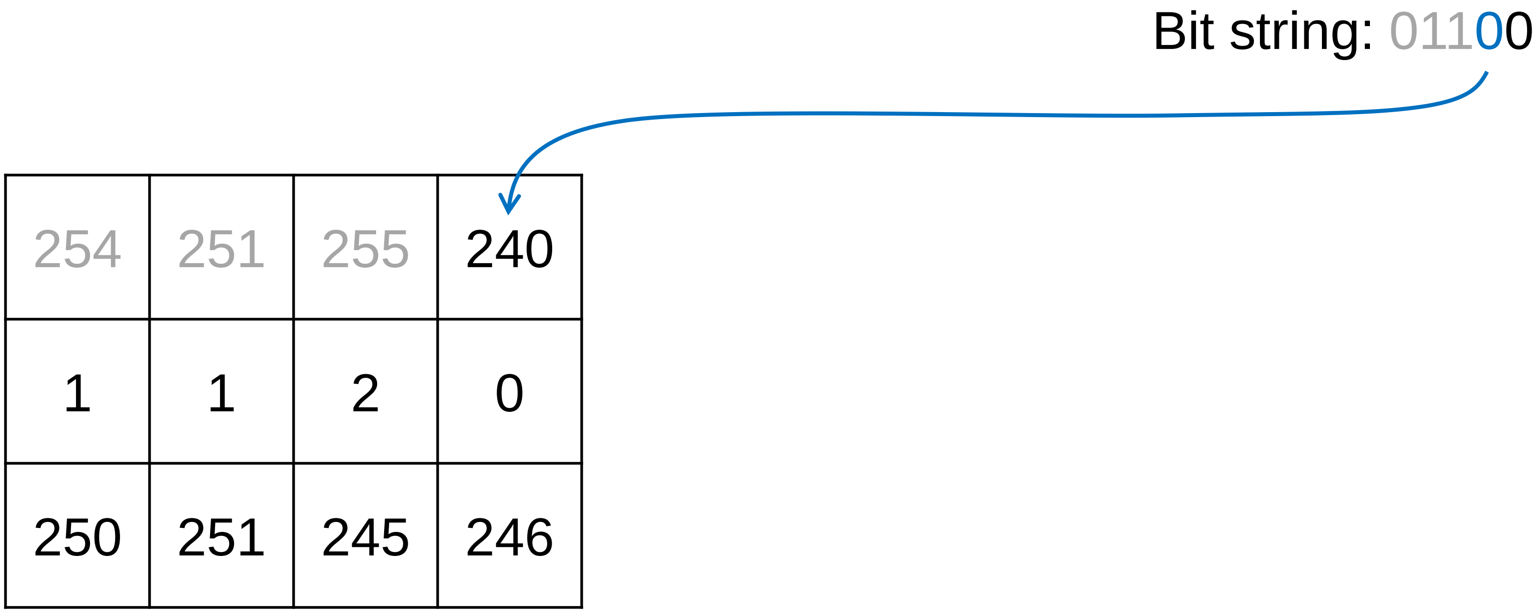


Bit string: 01100

254	251	255	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

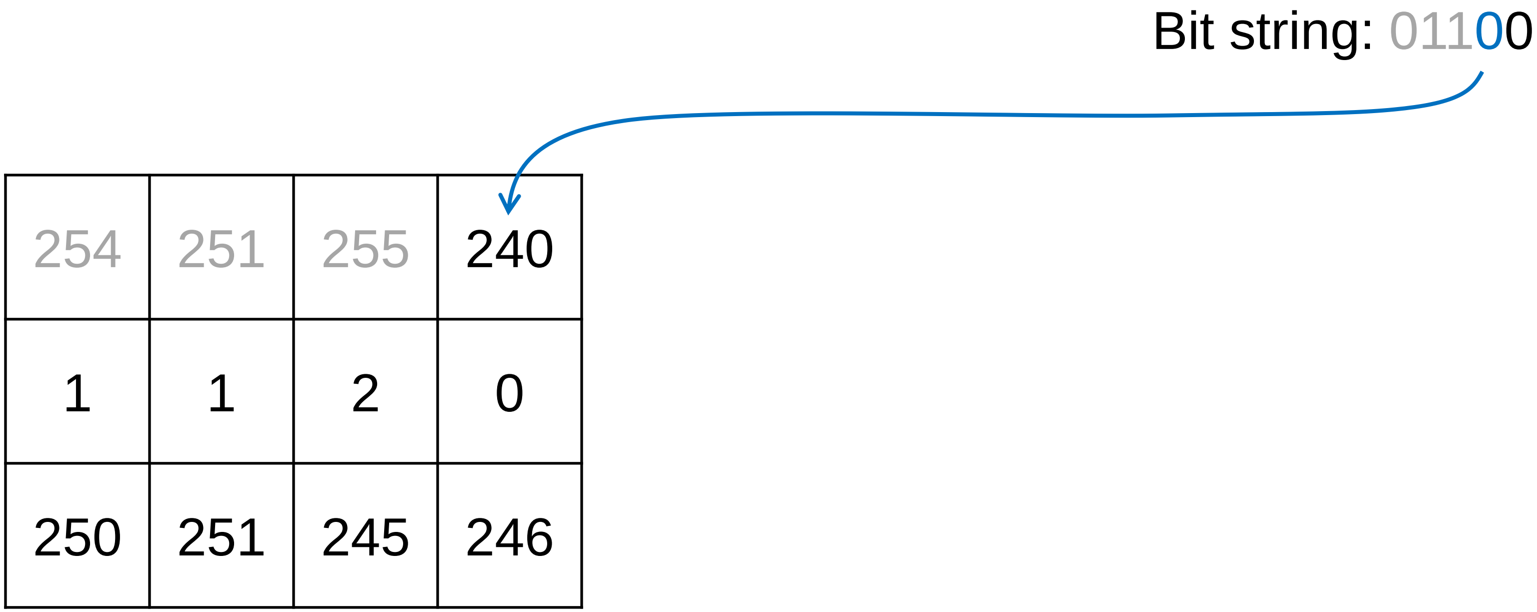


Bit string: 01100

254	251	255	240
1	1	2	0
250	251	245	246

Embedding secret bits on grayscale image using LSB method

Example 1: embed secret bit string 01100 in the image below

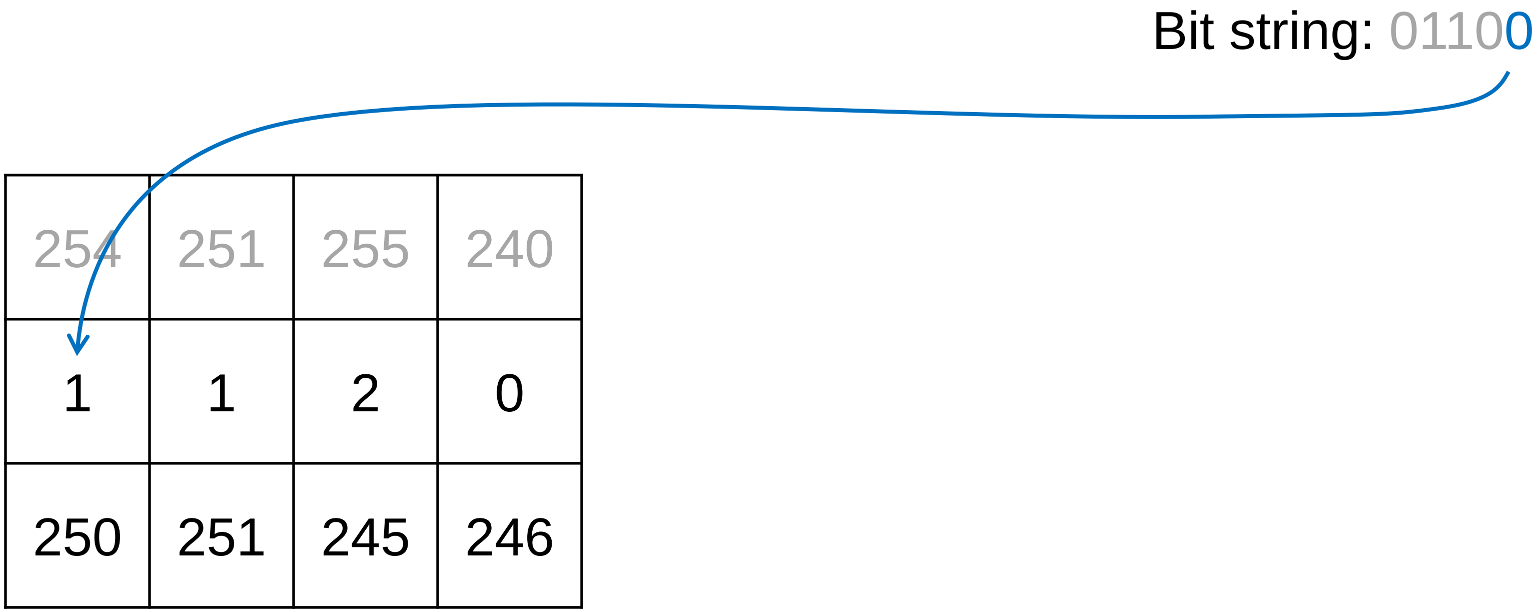


Bit string: 01100

254	251	255	240
1	1	2	0
250	251	245	246

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

Example 1: embed secret bit string 01100 in the image below

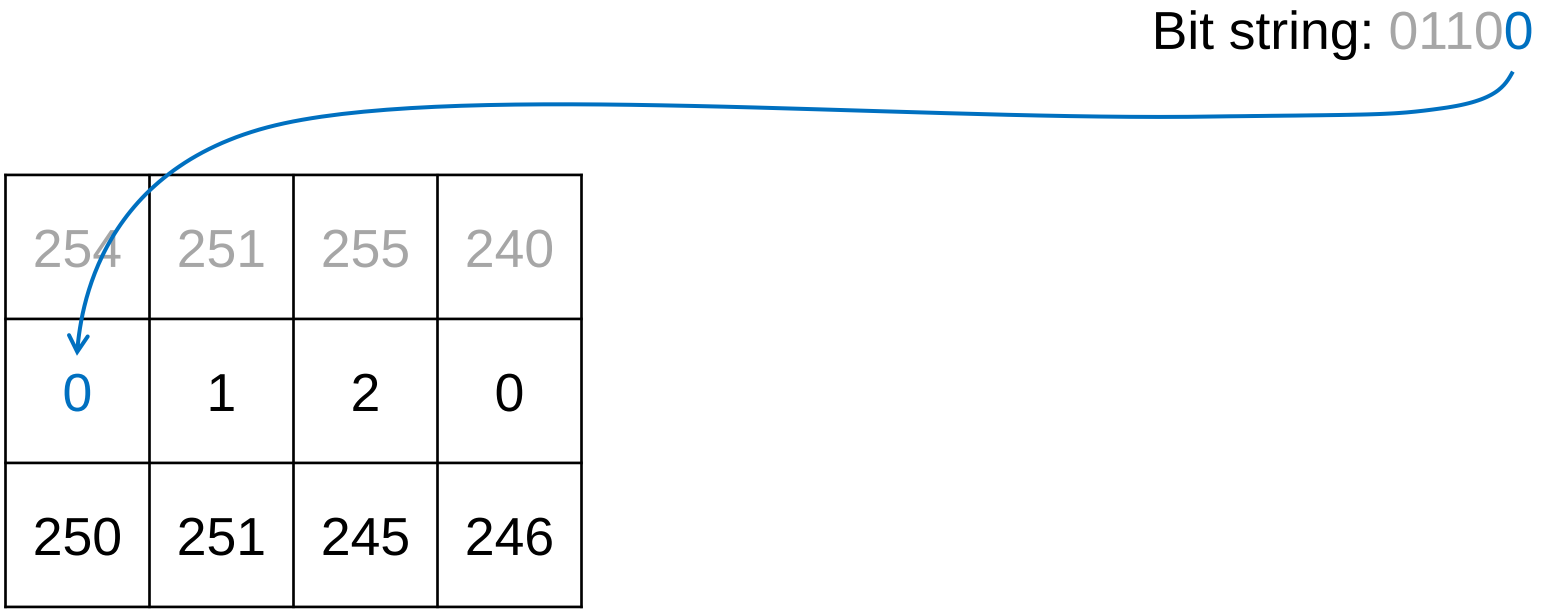


Bit string: 01100

254	251	255	240
1	1	2	0
250	251	245	246

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

Example 1: embed secret bit string 01100 in the image below



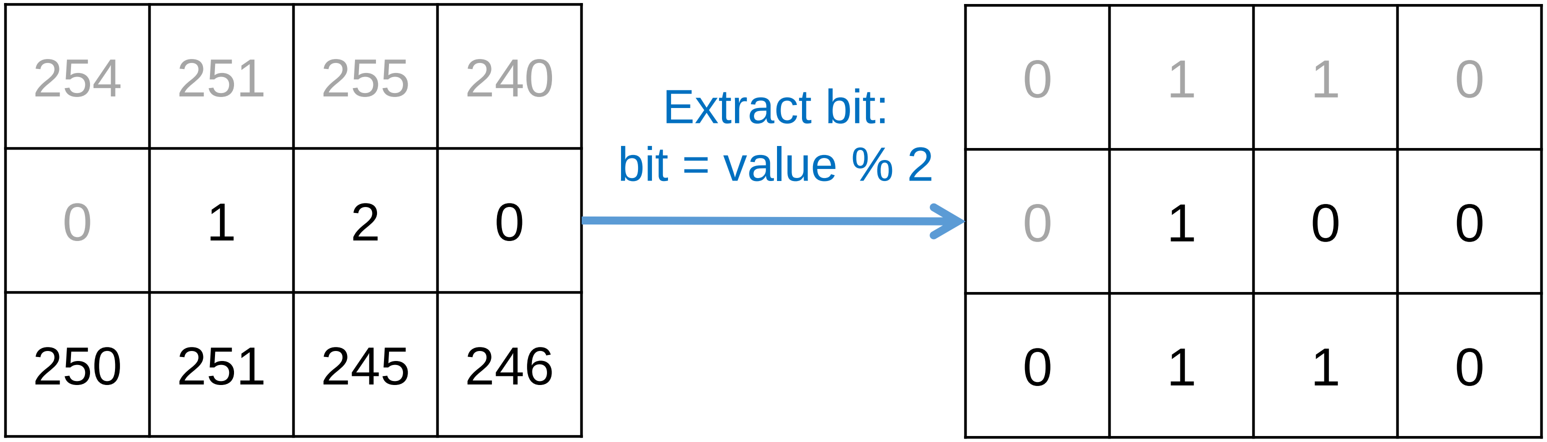
Bit string: 01100

254	251	255	240
0	1	2	0
250	251	245	246

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

Example 1: embed secret bit string 01100 in the image below

Try extracting from the embedded image...



254	251	255	240
0	1	2	0
250	251	245	246

Extract bit:
bit = value % 2

0	1	1	0
0	1	0	0
0	1	1	0

How to get the embedded bit string???

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

How to get the embedded bit string???

254	251	255	240
0	1	2	0
250	251	245	246

Extract bit:
bit = value % 2

0	1	1	0
0	1	0	0
0	1	1	0

One way is when embedding a 100... in the cryptographic bit string and embedding over the en
When extracting, we will get a bit string ending in 100... and could easily cut this tail off

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

Example 2: embed the secret bit string 0110001011100010 (16 bits) into the image below

255	250	254	240
1	1	2	0
250	251	245	246

The image has 12 pixel, each can store 1 bit, but need to embed up to 16 bits
 How can still embed?

Nhúng bit mật trên ảnh xám bằng phương pháp LSB

Example 2: embed the secret bit string 0110001011100010 (16 bits) into the image below

255	250	254	240
1	1	2	0
250	251	245	246

Embed ***k bits*** in ***k LSB bits*** of each pixel
For example, ***k* = *2*** then we will be able to em
12×2= 24 bits

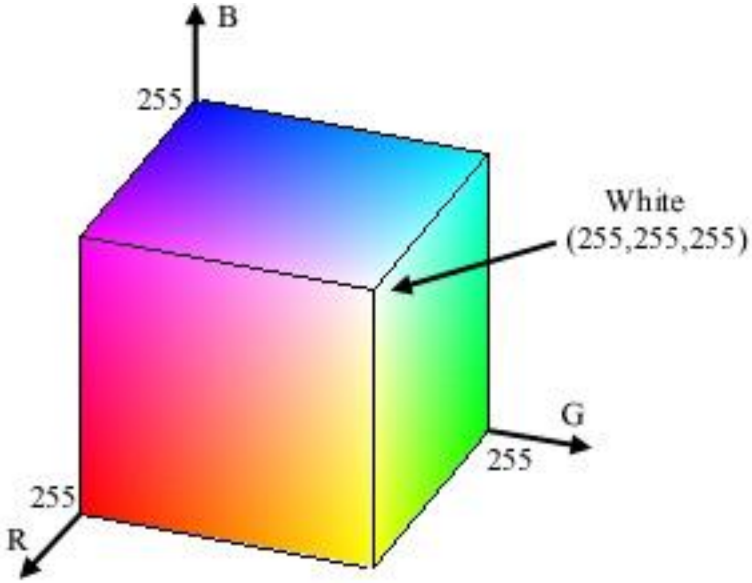
Increase k: *capacity*? *invisibility*?

Demo

RGB color Image

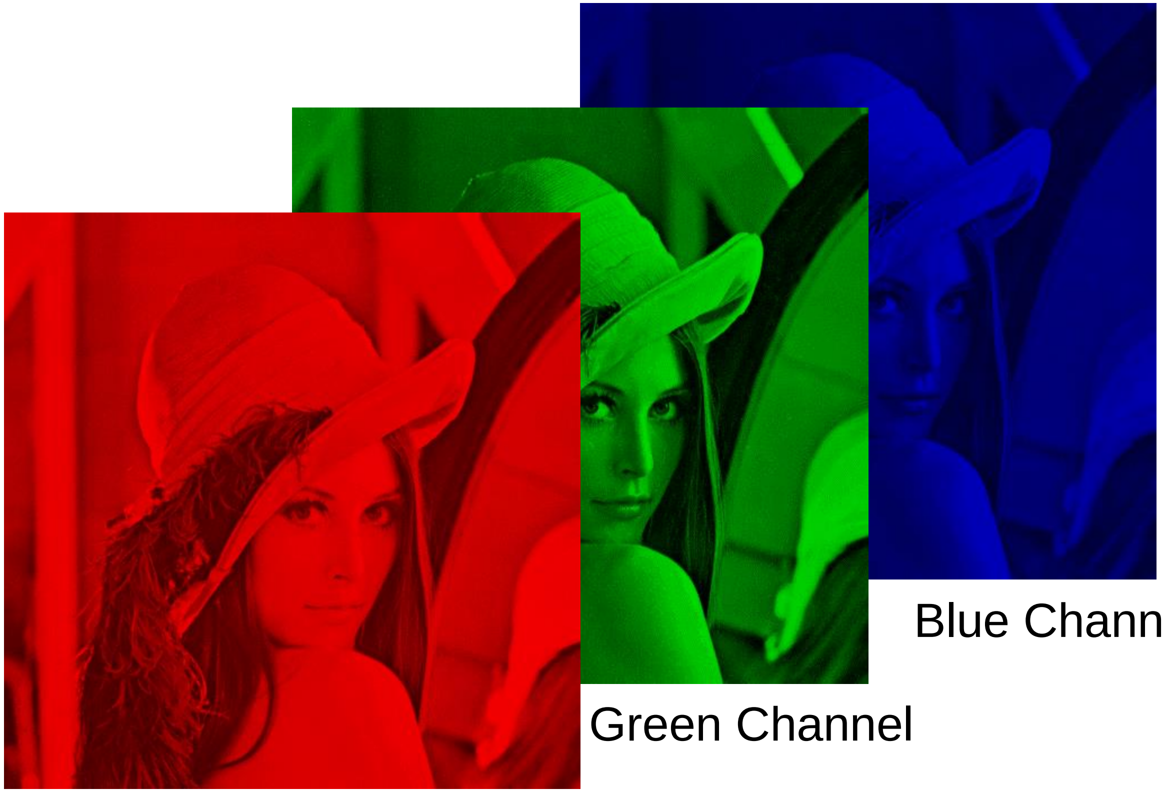
Each pixel is represented by a set of 3 integers (r, g, b), each integer has a value from 0-255 (8 bits), where :

- r indicate the intensity of red
- g indicate the intensity of green
- b indicate the intensity of blue



Nguồn ảnh RGB cube: http://radio.feld.cvut.cz/matlab/toolbox/images/color4.ht

Dùng phương pháp LSB cho ảnh màu RGB như thế nào?



Blue Channel

Green Channel

Red Channel

Embed LSB on each color channel like grayscale image

Analyze on the LSB method

- Invisibility?
  - ☺
- Capacity?
  - ☺
- Robustness?
  - ☹