Efficient Steganographic Embedding by Exploiting Modification Direction (EMD)

Source: IEEE communications letters, Vol. 10, No. 11, pp. 781-783,Nov. 2006.

Authors: Xinpeng Zhang and Shuozhong Wang

Chung-Ming Wang
cmwang@cs.nchu.edu.tw

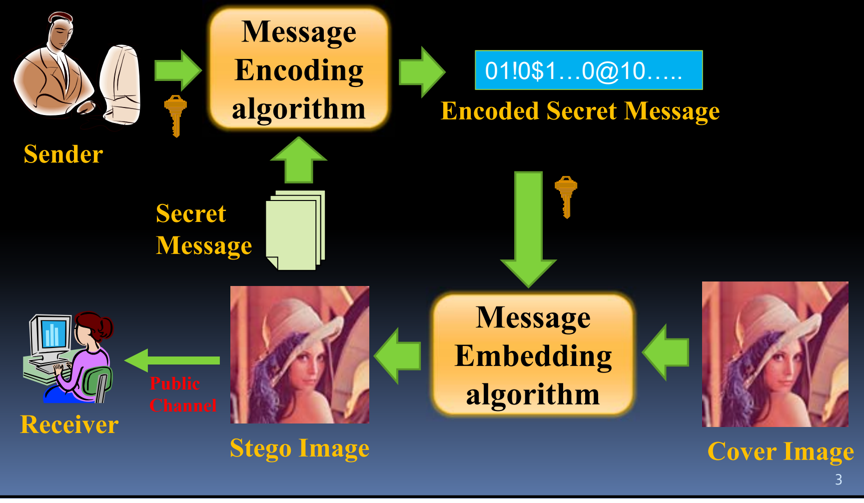
Department of Computer Science and Engineering,
National Chung Hsing University, Taiwan

Outline

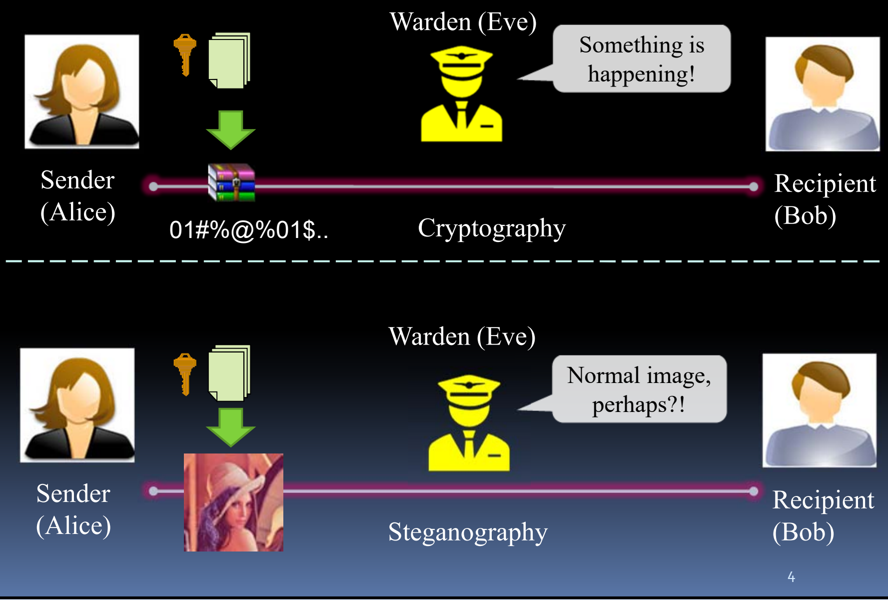
1. Introduction
2. Exploiting Modification Direction(EMD)
3. Experimental Results

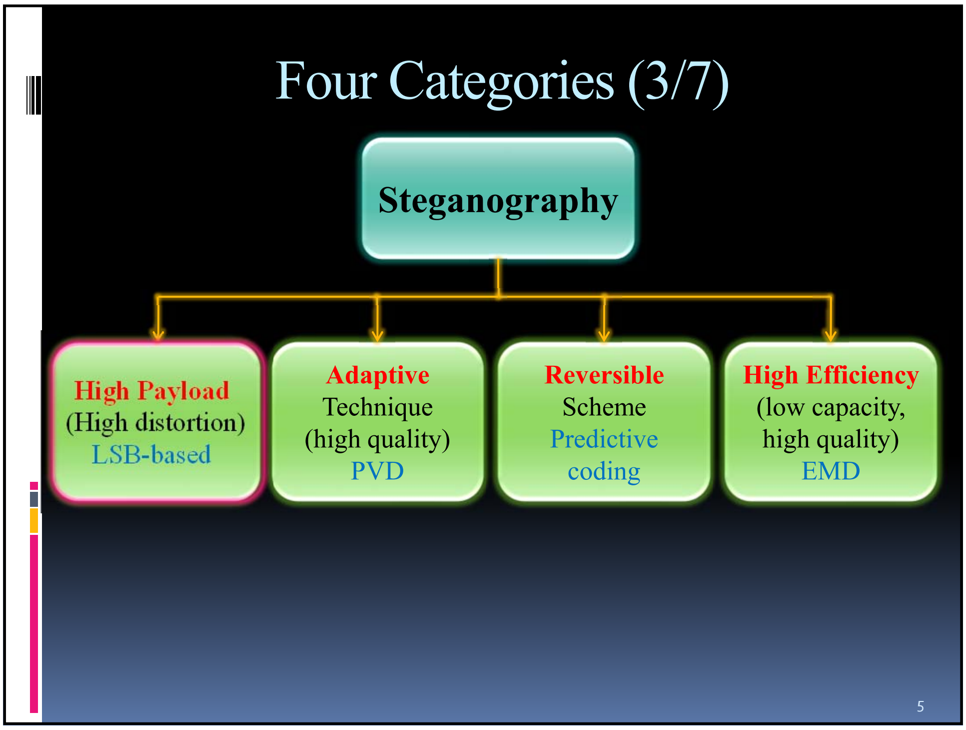
Steganography (1/7)

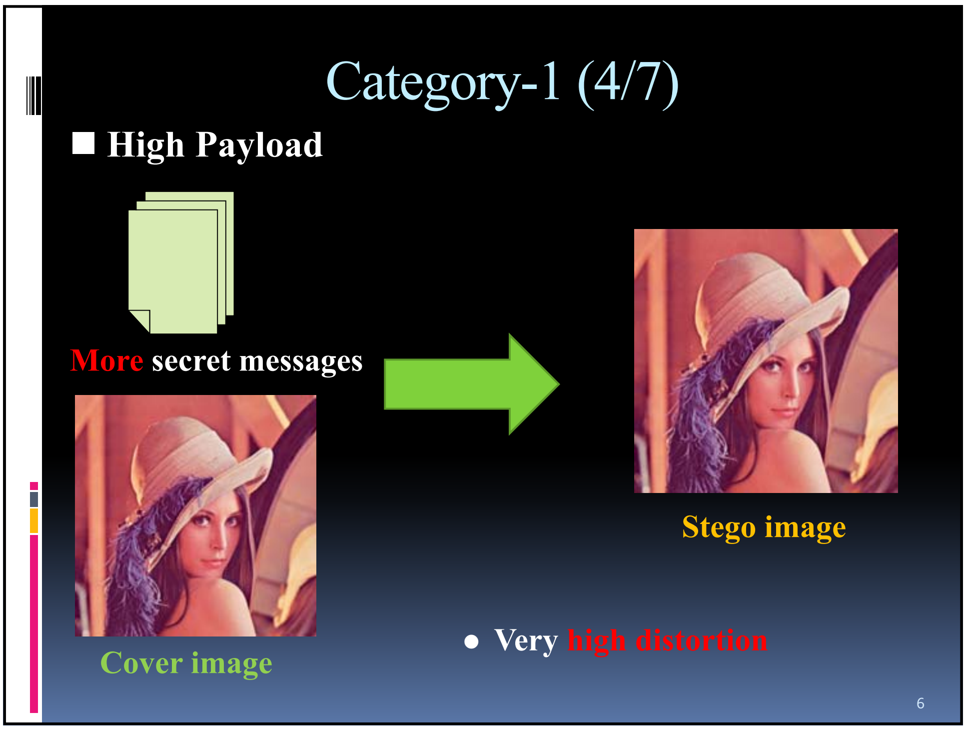
Steganography: (偽裝學、藏密學):
conceals secret information within seemingly innocuous carriers to achieve covert communication between the sender and the receiver



Steganography vs Cryptography




Four Categories (3/7)
Steganography
High Payload
(High distortion)
LSB-based
Adaptive
Technique
(high quality)
PVD
Reversible
Scheme
Predictive
coding
High Efficiency
(low capacity,
high quality)
EMD
5


Category-1 (4/7)
High Payload
More secret messages
Stego image
Cover image
Very high distortion
6

Four Categories

Steganography

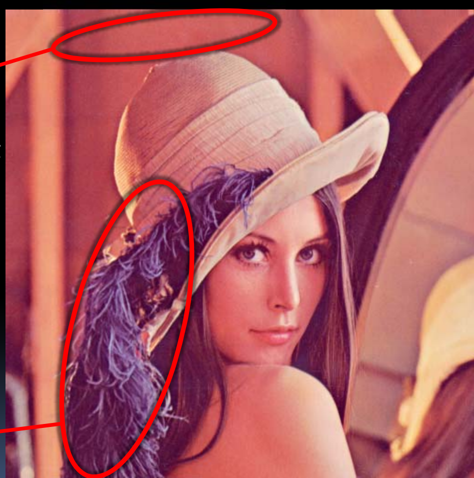
- **High Payload** (High distortion) LSB-based
- **Adaptive** Technique (high quality) PVD
- **Reversible** Scheme Predictive coding
- **High Efficiency** (low capacity, high quality) EMD

7

Category-2

- **Adaptive technique**

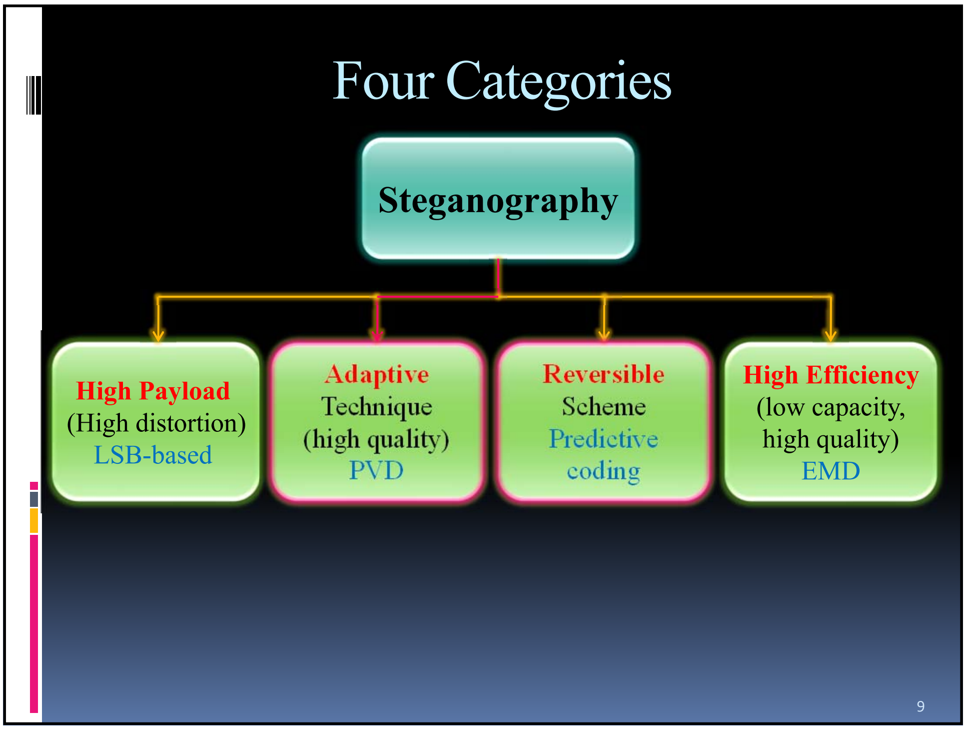
Cover image

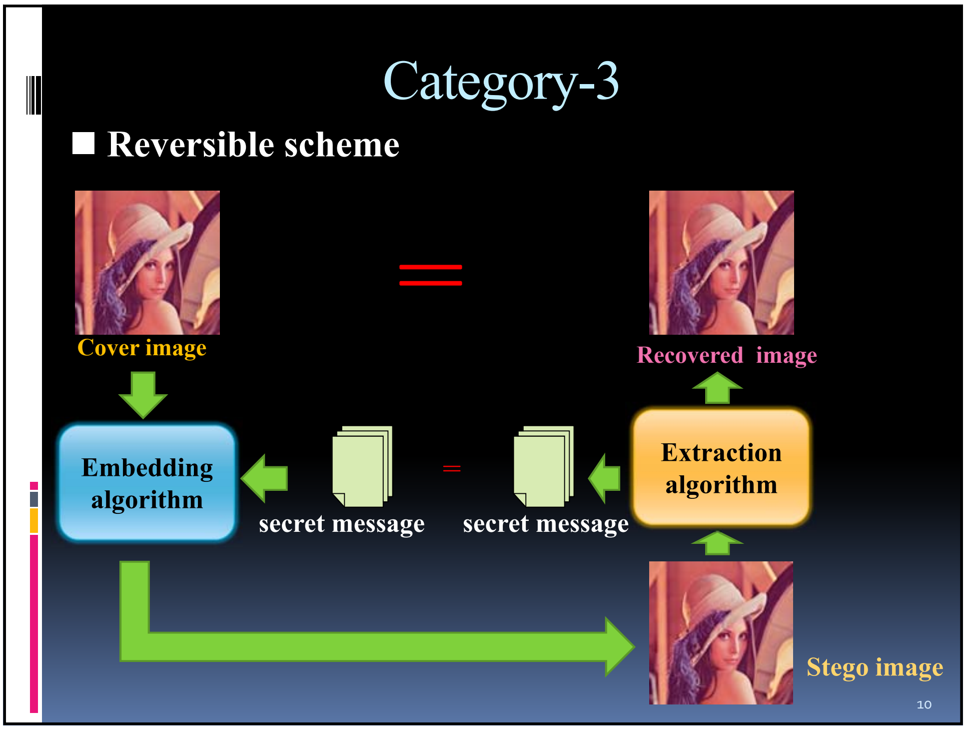


Smooth Area :
Embedding less secret message or does not embedding anything

Complex Area:
Embedding more secret message

8


Four Categories
Steganography
High Payload
(High distortion)
LSB-based
Adaptive
Technique
(high quality)
PVD
Reversible
Scheme
Predictive
coding
High Efficiency
(low capacity,
high quality)
EMD
9


Category-3
Reversible scheme
Cover image
Recovered image
Embedding algorithm
secret message
secret message
Extraction algorithm
Stego image
10

Four Categories

Steganography

- **High Payload** (High distortion) LSB-based
- **Adaptive** Technique (high quality) PVD
- **Reversible** Scheme Predictive coding
- **High Efficiency** (low capacity, high quality) EMD

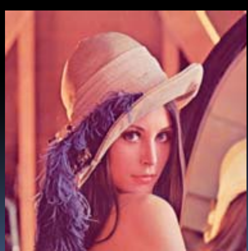
The EMD method provides high *embedding efficiency* that is better than previous techniques

11

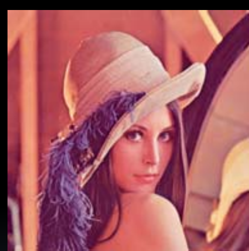
Category-4

■ **High Embedding Efficiency: conveying messages per unit of distortion**

secret message



Cover image



Stego image

- **Very high quality**

12

MSE

MSE (Mean Square Error) : distortion per pixel

Used to measure the distortion due to the message embedding

9	10	10	10	10	10	12	12	Cover image
9	10	10	9	10	11	12	12	Stego image

$$\text{MSE} = ((0)^2+(0)^2+(0)^2+(1)^2+(0)^2+(-1)^2+(0)^2+(0)^2)\ /\ 8$$
$$= (\ 2\ )\ /\ 8$$
$$= 0.25$$

13

PSNR

PSNR (Peak signal-to-noise ratio):

Used to represent the image quality after message embedding

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{\text{MSE}}$$

MSE = 0.25

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{0.25} = 54.1514\,\text{dB}$$

14

Visual Comparison-1 (LSB-k)

Cover

Stego k=1

Stego k=2

Secret Messages

Visual Comparison-2

Cover

Stego k=3

Stego k=4

Visual Comparison-3

Cover

Stego k=5

Stego k=6

Visual Comparison-4

Cover

Stego k=4

Stego k=7

Visual Comparison-5

Cover, Stego k=1, Stego k=2, Stego k=3

Stego k=4, Stego k=5, Stego k=6, Stego k=7

PSNR Requirement

PSNR (Peak signal-to-noise ratio): the higher, the better

Hard to perceive

30 dB

Easy to perceive

30 dB

0 dB

Introduction to MSE and PSNR(3/5)

PSNR (Peak signal-to-noise ratio)

Cover image : Lena

LSB-4
PSNR = 31.7891

ar

Easy to perceive

LSB-6
PSNR = 19.8864

0 dB

0 dB

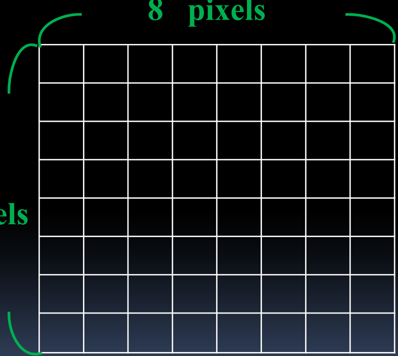
Embedding Rate

Embedding rate (R):
an average amount of secret bits concealed in each pixel

bits per pixel (bpp)

8 pixels

8 pixels



Stego image

: pixel

pixels: 64 pixels

Capacity: 32 bits secret data

Embedding rate
R = 32 / 64
= 0.5 bpp (bits per pixel)

Embedding efficiency

Embedding efficiency (E): balancing the embed. rate and MSE

$$\text{Embedding effciency} = \frac{\text{Embedding rate}}{MSE}$$

MSE = 0.25

Embedding rate = 0.5 bpp

Embedding efficiency E = 0.5 / 0.25 = 2

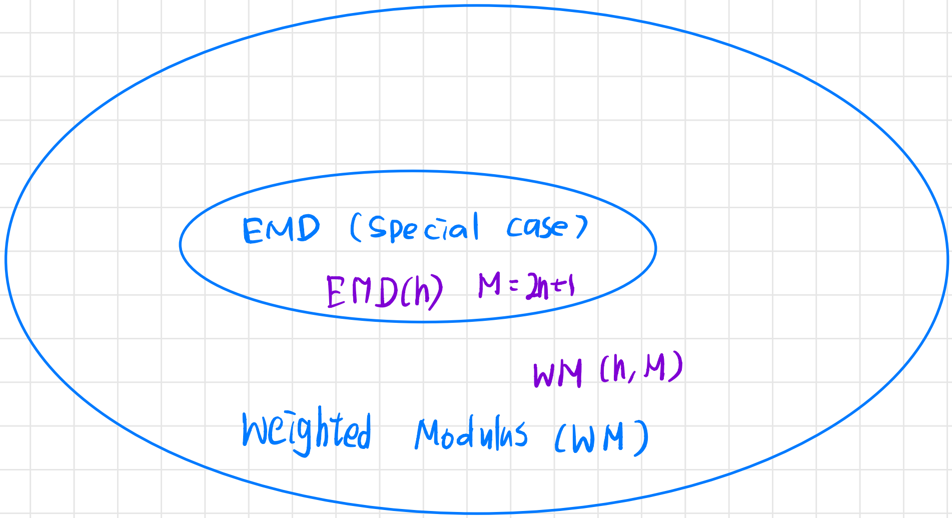
Embedding efficiency: the higher, the better

23

Exploiting Modification Direction (EMD)

- First assumption: secret message is a digit in a notational system rather than binary bits
- We will later describe an approach if secret messages are binary bits
- Input parameter n: number of pixels that are grouped together
- EMD conceals a (2n+1)-ary digit into n pixels
- e.g. n=3, we conceal a 7-ary digit into 3 pixels
- e.g. n=152, we conceal a 305-ary digit into 152 pixels $152 \times 2 + 1 = 304 + 1 = 305$
- Secret digit must be in an odd notational system

24



WM (n, M) when M = 2n+1

$\Rightarrow$ WM degenerates to EMD

退化

- EMD provides minimal distortion: one of three cases for pixel alternation
  - all n pixels do not need to change
  - only one out of n pixels will increase by 1 (+1), positive
  - only one out of n pixels will decrease by 1 (-1), negative

Digit Embedding and Extraction

- Embedding--Basic idea: given the secret digit d,
  - use the consecutive weight <u>**w**</u>=(1, 2, 3, …, n) to determine the extraction function, f
  - determine which pixel to be altered, s
  - determine the direction of the alternation (positive or negative)? addition (+) or subtraction (-)
- Extraction – Basic idea: given n, simply using the extraction function f to derive the secret digit d

EMD: Features

- Conveys a $(2n+1)$-ary secret digit in a cover pixel-group of n pixels
- Embedding rate (ER): $\log_2(2n+1)/n$ bit per pixel (bpp) 每个 pixel 藏多少 bits
- n=2, ER= $\log_2(5)/2$=1.16096 bpp (max.)
- ER decreases as n increases↑ bpp decreases ↓
- At most, only one pixel is increased or decreased by 1 (±1)

Some Statistics

- MSE in n pixels: 2/(2n+1)
- PSNR = $10*\log_{10}\{255*255/[2/(2n+1)]\}$

n	2n+1	R	MSE	PSNR	E
2	5	1.1610	0.4000	52.11	2.902
15	31	0.3303	0.0645	60.03	5.119
81	163	0.0907	0.0123	67.24	7.394
123	247	0.0646	0.0081	69.05	7.981
157	315	0.0529	0.0063	70.10	8.326

n=2

00
10
01

2n+1=5 ary

	1	2		mod5
0	0	0		0
1	1	0		1
2	0	1		1
3	0	-1	-2	1
4	-1	0	-1	1

$\frac{}{4}$

MSE: $(4/2)$ / 5种状态 $= \frac{4}{10} = \frac{2}{5} \leftarrow \frac{2}{2n+1}$

n= 2bits

$g_{2n+1} - s = g_{2n+1} - s - 1$

EMD 3-Step Procedure

- 1. Determine the extraction function f

  $f = [\sum_{i=1}^{n}(g_i \times i)] \bmod (2n+1)$
  - f = [**G** dot **w**] mod (2n+1) /* expressed as dot operation
  - **G**= [$g_1, g_2, \ldots, g_n$] and **w**=[1, 2, …, n]
- 2. If $d = f$, $G' = G$ /* no need to change any pixel

  else $s = (d - f) \bmod (2n+1)$ /* alter s-th/ 2n+1-s pixel
- 3. If $s \le n$, $g_s' = g_s + 1$ /* positive direction

  else $g'_{2n+1-s} = g_{2n+1-s} - 1$ /* negative direction

29

WM

n个可动n个

不能用 2n+1-S

索引值找

要用table找

n=3 EMD Weight

1 2 3

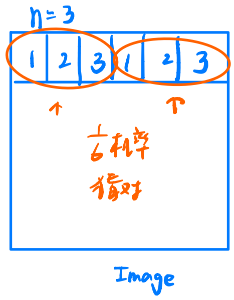
WM 3! = 6种 123 132 213 231 312 321

EMD Example-1

- n=4 (four pixels in a group) and secret digit $d=4_9$
- ***G***= (137, 139, 141, 140)
- f=(137×1+139×2+141×3+140×4) % (2*4+1)=1398 % 9 = 3
- s=(d-f) % (2n+1) = (4-3) mod 9 = 1
- Since $s \le n$ $(1 \le 4)$, $g_1' = g_1 + 1$ /* positive direction
- G'=(137+1, 139, 141, 140)
- Extraction: given G' and n,
- f=(138×1+139×2+141×3+140×4) % (2*4+1)

  = 1399 % 9=4_9

30



n! = 3!

WM

$(\frac{1}{6})^{\frac{M \times N}{3}}$ ↔ EMD 1

猜对机率 固定 必猜中

d=0

G=(0,0,0,1)

f= 4%9=4

d= (0-4)%9= -4+9=5

5>4

$g'_4 = g_4 - 1$

d=6

G=(0,0,0,1)

f= 4%9=4

d=(6-4)%9=2

$s=1 \quad g_1'=g_1+1$ $\qquad$ $s=5 \quad g'_{9-5}=g'_4=g_4-1$

$s=2 \quad g_2'=g_2+1$ $\qquad$ $s=6 \quad g'_{9-6}=g'_3=g_3-1$

$s=3 \quad g_3'=g_3+1$ $\qquad$ $s=7 \quad g'_{9-7}=g'_2=g_2-1$

$s=4 \quad g_4'=g_4+1$ $\qquad$ $s=8 \quad g'_{9-8}=g'_1=g_1-1$

now 餘2
want 餘7) +5
或

$5-9=-4$

$88-81=7$

$(83+5)\%9=7$

$(83-4)\%9=7$

79
7
72

EMD Example-2

- n=4 (four pixels in a group) and secret digit d=~~3~~ 7$_9$
- $\underline{\boldsymbol{G}}$= (14, 13, 9, 4)
- f=(14×1+13×2+9×3+4×4) % (2*4+1)=2 $\quad$ 83 $\quad$ 9
- s=(d-f) % (2n+1) = (~~3~~ 7-2) mod 9 = ~~1~~ 5
- Since $s \le n$ $(1 \le 4)$, $g_1'=g_1+1$ /* positive direction
- G'=(14+1, 13, 9, 4)=(15, 13, 9, 4)
- Extraction: given G' and n,
- f=(15×1+13×2+9×3+4×4) % (2*4+1)
  = 84 % 9=3$_9$

$79\%9=7$
$78\%9=6$
$77\%9=5$
$76\%9=4$

31

$g'_{9-5} = g_4 - 1$ (4)

ex. $5>4$ (5) $\quad g'_{2\times4+1-5} = g_{2\times4+1-5} - 1$

Pixel Saturation

- If $s \le n$, $g_s'=g_s+1$ /* positive direction
  else $\quad g'_{2n+1-s}=g_{2n+1-s}-1$ /* negative direction
- Increase or decrease may not be allowed if the pixel is saturated (<0 or >255)
- Solution:
- 1. change the saturated pixel by 1
- 2. embed the secret digit again
- 3. repeat 1 and 2 until there is no saturated pixel

32

$s=5 \qquad g_4-1$

$d = 6, n = 4$

$G = (0,0,0,1)$

$f = 4 \% 9 = 4$

$s = (6-4) \% 9 = 2$

$s \leq n \ (2 \leq 4)$

$g_2' = g_2 + 1$

$G' = (0,1,0,1)$

$d = 6, n = 4$

$G = (1,0,0,0)$

$f = 1 \% 9 = 1$

$s = (6-1) \% 9 = 5$

$s > n \ (5 > 4)$

$g'_{9-5} = g_{9-5} - 1$

$g'_4 = g_4 - 1$

$G' = (1,0,0,\cancel{-1})$

$G = (1,0,0,1)$ $1+4=5$

$f = 5 \% 9 = 5$

$s = (6-5) \% 9 = 1$

$s \leq n \ (1 \leq 4)$

$g'_1 = g_1 + 1$

$G \ (2,0,0,1)$

Pixel Saturation: Example 1/3

- n=4 (four pixels in a group) and secret digit $d=0_9$
- $\underline{\boldsymbol{G}}$= (255, 255, 255, 254)

  f=(255 ×1+255×2+255×3+254×4) % 9

  =2546 % 9 =8
- s=(0-8) % 9 = (-8 % 9) = 1
- Since $s \leq n$ $(1 \leq 4)$, $g_1' = g_1 + 1$ /* positive direction
- G'=(255+1, 255, 255, 254)=(256, 255, 255, 254)
- Saturation pixel: in 1st pixel

33

G' (0-1, 0, 0, 1) = (-1, 0, 0, 1)

G (1, 0, 0, 1)

Pixel Saturation: Example 2/3

- 1. change the saturated pixel by 1

  $\boldsymbol{G}$= (254, 255, 255, 254)
- 2. embed the secret digit again

  f=(254 ×1+ 255 ×2+255×3+254×4) % 9

  = 2545 % 9 =7
- s=(0-7) % 9 = (-7 % 9) = 2
- Since $s \leq n$ $(2 \leq 4)$, $g_2' = g_2 + 1$ /* positive direction
- G'=(254, 256, 255, 254)
- Saturated pixel in 2^{nd} pixel

34

Pixel Saturation: Example 3/3

- 1. change the saturated pixel by 1

  G= (254, 254, 255, 254)

- 2. embed the secret digit again

  f=(254×1+254×2+255×3+254×4) % (2*4+1)=2543 % 9 =5

- s=(0-5) % 9 = (-5 % 9) = 4
- Since $s \le n$ $(4 \le 4)$, $g_4' = g_4 + 1$ 原(255,255,255,254)
- G'=(254, 254, 255, 254+1)=(254, 254, 255, 255) 变动量 3.
- Extraction: given G' and n,
- f=(254 ×1+254×2+255×3+255) % 9

  = 2547 % 9=0_9

35

n=4 最多3次移动

Implementatoin

- Implement EMD message embedding algorithm
- Input:

  1. n>=2 but n<=MxN (ignore pixels not in a group)

  2. seed = 100 to produce a random sequence as (2n+1)-ary secret digits which are in the range of 0~2n

  3. a color image, say cover-kodak-1
- Output:

  1. a stego image, say stego-kodak-1

  2. a secret digit file, say secret-digit.txt, separate by space

中興大學資工系 GMVR 王宗銘 36

Suggestion for the Development

- You are suggested to use a function to conceal the secret messages
- This function can be re-used later in the next assignment
- e.g.
- EMD_Mess_Conceal(n, K, d, pixel-arrary[])
- n: number of pixels in a group
- K: this integer parameter will be used later
- d: secret digit to be concealed
- pixel-array[]: pixel array representation

37

Current Limitation

- Previous EMD approach assumes that the secret messages are represented by (2n+1)-ary digits
- Secret messages are normally being compressed and encrypted
  - Save space and increase the security
- It is reasonable to consider secret messages are a binary bitstream with an equal probability of the bits "0" and "1"
- We need two extra parameters K and L to cope with this situation

38

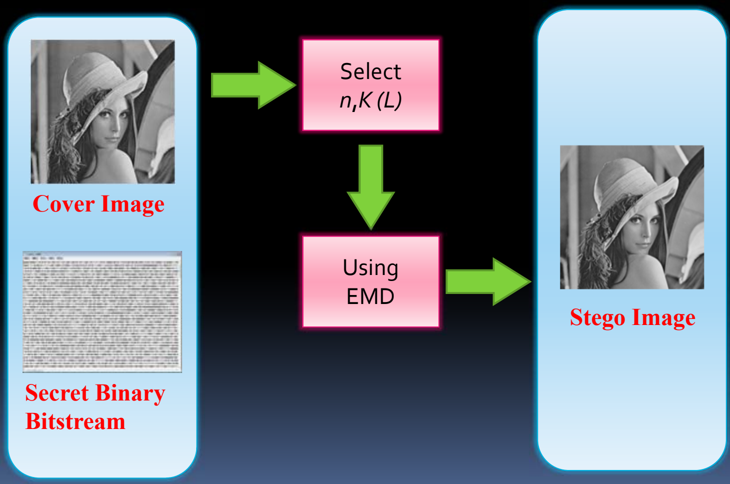
Exploiting Modification Direction

- We assume the following two conditions:
- 1. The secret message to be concealed is a binary bitstream
- 2. The binary bitstream contains bits "0" or "1" which have equal probability of appearance
- EMD(n, K) has two input parameters
- n: number of cover pixels in a group
- K: number of (2n+1)-ary digits to carry L-bits secret bits
- $L = \lfloor K \times log_2(2n+1) \rfloor$

39

Exploiting Modification Direction(1/17)

- **EMD Embedding**



40

Exploiting Modification Direction(2/17)

- **Related variables**

n: a group of cover pixels, e.g. $n = 4$
K: the digits in a $(2n+1)$-ary notational system, e.g. K=2

$$\boxed{L = \lfloor K \cdot \log_2(2n+1) \rfloor}$$ e.g. $$\boxed{L = \lfloor \boxed{2} \cdot \log_2(2 \cdot \boxed{4} + 1) \rfloor = 6}$$

Cover Image

9	10	10	10

Group 1

10	10	12	12

Group 2

110101 101001 …00110001000…10_2

6 bits

Exploiting Modification Direction(3/17)

- **Related variables**

n: a group of cover pixels,
K: the digits in a $(2n+1)$-ary notational system,
L: the length of secret message to be subdivided into a message fragment

EX: $n = 4$, $K = 2$

Secret message : $(110101101001110100..)_2$

$$\boxed{L = \lfloor 2 \cdot \log_2(2 \cdot 4 + 1) \rfloor = 6}$$

Secret message : (110101_2 101001_2 110100_2 ..)

$\frac{53}{7^0} \% 7 = 4$

$\frac{53}{7^1} \% 7 = 0$

$\frac{53}{7^2} \% 7 = 1$

$n=3, \; K=5, \; L = \lfloor 5 \cdot \log_2 7 \rfloor = 14$

$53_{10} = 104_7$

$7^2 + 4 = 49 + 4 = 53$

$53 \% 7 = 4$

$53/7 \% 7 = 0 \quad \lfloor \frac{53}{7} \rfloor \% 7$

$\lfloor \frac{53}{7 \times 7} \rfloor \% 7 = 1$

Exploiting Modification Direction(4/17)

- **EMD Embedding**

I. Select n , K

EX: $n = 4$, $K = 2$, $\boxed{L = \lfloor 2 \cdot \log_2(2 \cdot 4 + 1) \rfloor = 6}$

II. Reading secret message

- convert a secret message into a sequence of (2n+1) digits

EX: Secret message : ($\boxed{110101}$ 101001 110100 .. $)_2$

53_{10}

$K = 2$ digits

53_{10} → $\boxed{58}_9$ → $2 \cdot 4 + 1 = 9$

for j in range(k):

dec

$\lfloor 2^j \rfloor \% 9$

Group

9	10	10	10

Group

10	10	12	12

43

$53 \% 9 = 8$

$53/9 \% 9 = 5$

5

$8 + 5 \times 9 = 53$

45

$\frac{L}{K} = \frac{6}{2} = 3$

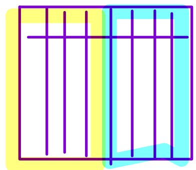
$\frac{H \times V}{n} = \frac{8 \times 8}{4} = 16$

$16 \times 3 = 48$

32
16
4
1
53

8

8



Exploiting Modification Direction(5/17)

- **Why " $2n+1$ " ?**

EX: $n = 4$

Cover pixel :

9	10	10	10

→ **Group**

8	9	9	9

+1			
	+1		
		+1	
			+1
			-1
		-1	
	-1		
-1			
0	0	0	0

$2 \cdot 4 + 1 = 9$ possible ways of modification

Range of the secret data : 0 ~ 8

44

$\frac{H \times V}{n} \quad \frac{8 \times 8}{4} = 16$

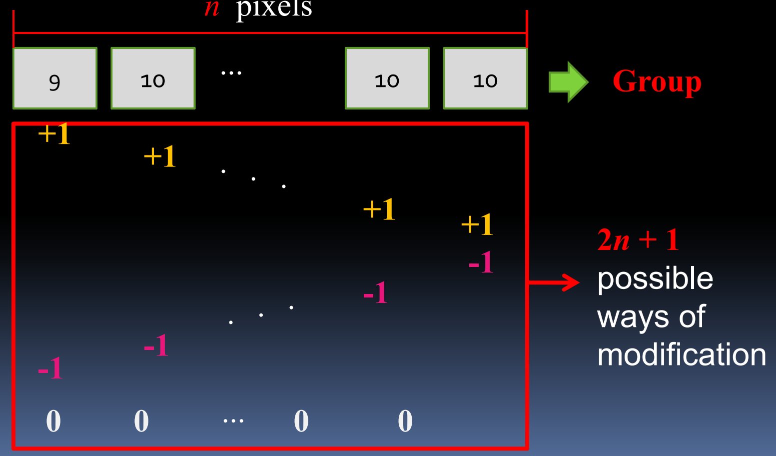
$\frac{16}{k} = \frac{16}{2} = 8$

$8 \times L = 48$

6

Exploiting Modification Direction(6/17)

- **Why "2*n*+1" ?**
  each group of *n* pixels :



45

EMD 3-Step Procedure

- 1. Determine the extraction function f

  $$f = [\sum_{i=1}^{n}(g_i \times i)] \bmod (2n+1)$$

  - f = [<u>**G**</u> dot <u>**w**</u>] mod (2n+1) /* expressed as dot operation
  - <u>**G**</u>= [g_1, g_2, …, g_n] and <u>**w**</u>=[1, 2, …, n]
- 2. If $d = f$, G' =G /* no need to change any pixel

  else $s = (d - f) \bmod (2n+1)$ /* alter s-th/ 2n+1-s pixel
- 3. If $s \le n$, $g_s' = g_s + 1$ /* positive direction

  else $g'_{2n+1-s} = g_{2n+1-s} - 1$ /* negative direction

46

Exploiting Modification Direction(7/17)

9	10	10	10

n=4
$d=5_9$

1. Determine the extraction function f

$f = (9 \times 1 + 10 \times 2 + 10 \times 3 + 10 \times 4)\ mod\ 9 = 99\ mod\ 9 = 0$

2. If $d = f$, $G' = G$ /* no need to change any pixel
   else $s = (d - f) \bmod (2n + 1)$ /* alter s-th/ 2n+1-s pixel

   Since $d=5_9 \neq f=0 \rightarrow$ s=(5-0) mod 9 = 5

3. If $s \leq n$, $g'_s = g_s + 1$ /* positive direction
   else $g'_{2n+1-s} = g_{2n+1-s} - 1$ /* negative direction

   $s=5 \geq n=4 \rightarrow g'_{2n+1-s} = g_{2n+1-s} - 1 \rightarrow g'_4 = g_4 - 1$

   $2 \times 4 + 1 - 5 = 9 - 5 = 4$

9	10	10	10-1	→	9	10	10	9

Exploiting Modification Direction(9/17)

f = 0 *d* = 5	4 pixels					
	$g_1 = 9$	$g_2 = 10$	$g_3 = 10$	$g_4 = 10-1$	**Sum**	***f = sum mod 9***
Weight	**1**	**2**	**3**	**4**	99	0
$d = 1_{(9)}$	**+1**				100	1
$d = 2_{(9)}$		**+1**			101	2
$d = 3_{(9)}$			**+1**		102	3
$d = 4_{(9)}$				**+1**	103	4
$d = 5_{(9)}$				**-1**	95	5
$d = 6_{(9)}$			**-1**		96	6
$d = 7_{(9)}$		**-1**			97	7
$d = 8_{(9)}$	**-1**				98	8

Exploiting Modification Direction(10/17)

- **EMD Embedding**
  - Extraction function *f*:

$$f(g_1, g_2, \dots, g_n) = \left[\sum_{i=1}^{n}(g_i \cdot i)\right] \bmod (2n+1)$$

9-ary notational system : $(\ 5\ \boxed{8}\ )_9$, $n = 4$

Group

10	10	12	12

$d = 8$,

$$f(g_1, g_2, g_3, g_4) = (10 \times 1 + 10 \times 2 + 12 \times 3 + 12 \times 4) \bmod 9$$
$$= 114 \bmod 9 = 6$$

9	10	10	9	10	10	12	12
9	9	10	10	10	11	12	13
8	7	6	5	15	14	13	12
15	12	14	29	24	25	31	22

Cover image

8 ≠ 6 only **one pixel require to be +1 or -1**.

49

Exploiting Modification Direction(11/17)

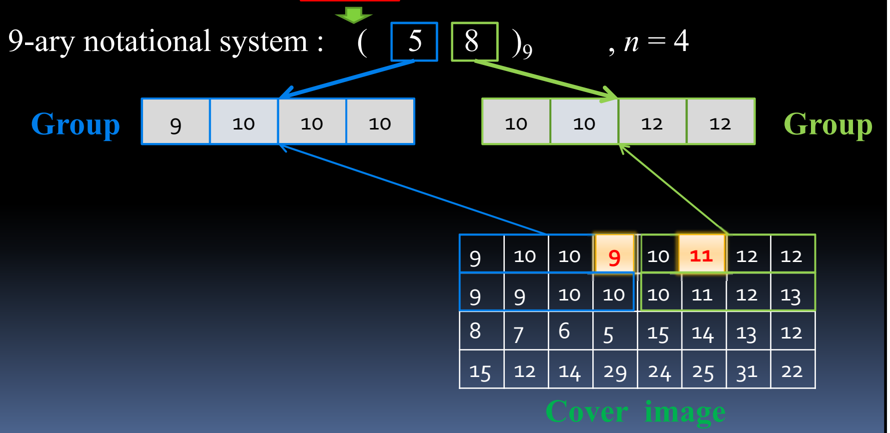
f = 6 *d* = 8	4 pixels					
	$g_1 = 10$	$g_2 = 10+1$	$g_3 = 12$	$g_4 = 12$	**Sum**	*f = sum mod 9*
Weight	**1**	**2**	**3**	**4**	114	6
$d = 7_{(9)}$	+1				115	7
$d = 8_{(9)}$		+1			116	8
$d = 0_{(9)}$			+1		117	0
$d = 1_{(9)}$				+1	118	1
$d = 2_{(9)}$				-1	110	2
$d = 3_{(9)}$			-1		111	3
$d = 4_{(9)}$		-1			112	4
$d = 5_{(9)}$	-1				113	5

Exploiting Modification Direction(12/17)

■ **EMD Embedding**

$n = 4$, $K = 2$ $L = 6$

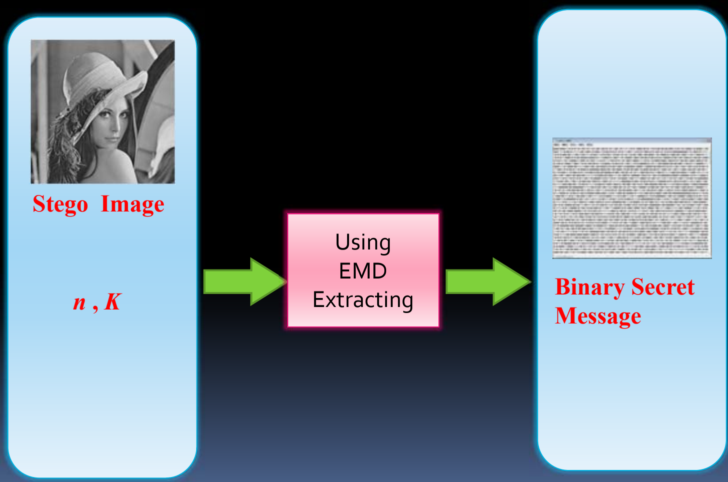
Secret message : $(\boxed{110101}101001110100..)_2$



51

Exploiting Modification Direction(13/17)

■ **EMD Extracting**



52

Exploiting Modification Direction(14/17)

- **EMD Extracting**

I. *n* , *K*

EX: $n = 4$, $K = 2$, $\boxed{L = \lfloor 2 \cdot \log_2(2 \cdot 4 + 1) \rfloor = 6}$

9 → 2·4 + 1 = 9

Stego pixels:

9	10	10	9	10	11	12	12

Group Group

9	10	10	9	10	11	12	12
9	9	10	10	10	11	12	13
8	7	6	5	15	14	13	12
15	12	14	29	24	25	31	22

Stego image

Exploiting Modification Direction(15/17)

- **EMD Extracting**

II. Data extracting

$\boxed{5}$ 9 → 2·4 + 1 = 9

Group

9	10	10	9	10	11	12	12

❖ Extraction function *f*:

$$\boxed{f(g_1, g_2, \ldots, g_n) = \left[\sum_{i=1}^{n}(g_i \cdot i)\right] \text{mod } (2n+1)}$$

$$f(g_1, g_2, g_3, g_4) = (9 \times 1 + 10 \times 2 + 10 \times 3 + 9 \times 4) \bmod 9$$
$$= 95 \bmod 9 = 5$$

$\frac{3844}{2}$

Exploiting Modification Direction(16/17)

■ **EMD Extracting**

5 8 9 → $2 \cdot 4 + 1 = 9$

9	10	10	9	10	11	12	12	**Group**

❖ Extraction function *f*:

$$f(g_1, g_2, \ldots, g_n) = \left[\sum_{i=1}^{n} (g_i \cdot i)\right] \bmod (2n+1)$$

$$f(g_1, g_2, g_3, g_4) = (10 \times 1 + 11 \times 2 + 12 \times 3 + 12 \times 4) \bmod 9$$
$$= 116 \bmod 9 = 8$$

55

14131 (7)

↓

3844 (10)

↓

0011110000100 (2)

Exploiting Modification Direction(17/17)

■ **EMD Extracting**

9	10	10	9	10	11	12	12
9	9	10	10	10	11	12	13
8	7	6	5	15	14	13	12
15	12	14	29	24	25	31	22

Stego image

5 8 9

↓ $5 \times 9 + 8$

53_{10}

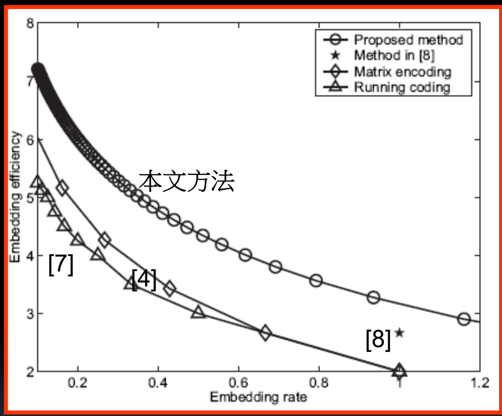
$$\begin{array}{r} 45 \\ 8 \\ \hline 53 \end{array}$$

↓

Secret message : $(110101)_2$

56

Experimental Results(1/2)



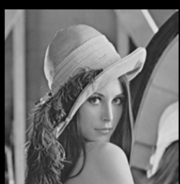
◇: [4] A. Westfeld, “F5: a steganographic algorithm,” in *Proc. 4th Int. Workshop Information Hiding 2001, Lecture Notes in Computer Science*, vol.2137, pp. 289-302.

△: [7] X. Zhang and S. Wang, “Dynamically running coding in digital steganography,” *IEEE Signal Processing Lett.*, vol. 13, no.3, pp. 165-168, Mar. 2006.

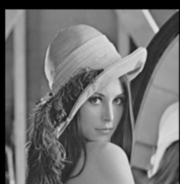
57

Experimental Results(2/2)

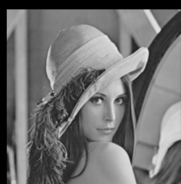
Resolution : 512 × 512



Cover image : Lena



Cp = 0.75 bpp
PSNR = 54.6618 dB



Cp = 0.037 bpp
PSNR = 72.0305 dB

Cover image : Jet

Cp = 0.2 bpp
PSNR = 62.8837 dB

Cp = 0.008bpp
PSNR = 80.0117 dB

C_p = Embedding rate

58

Comments

- High embedding efficiency
- Nonreversible
- Hard to determine (*n*, *K*) for optimal embedding
  - Through analysis to find the optimal (*n*, *K*)
- Not able to predict the embedding results prior to the real embedding process
  - Derive capacity for every possible combination of (*n*, *K*)
  - Satisfy the user demanding
- Time consuming
  - Using double or triplet encoding approach

Thank you for listening!!