



# 使用權重技巧的高品質偽裝學演算法

A Very High Quality Steganographic Algorithm Using  
a Novel Weighted Modulus Technique

王宗銘

[cmwang@cs.nchu.edu.tw](mailto:cmwang@cs.nchu.edu.tw)

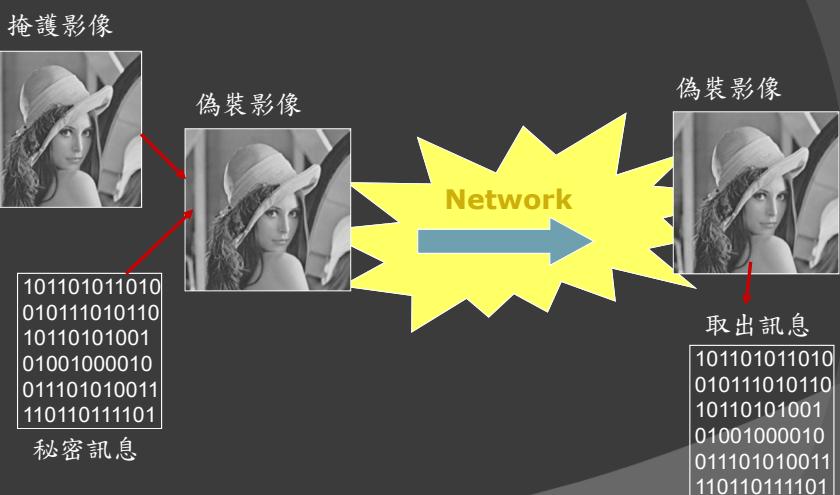
投影片原創作者  
研究生黎世琦

國立中興大學 資訊網路與多媒體所 圖學多媒體暨虛擬實境實驗室  
National Chung Hsing University - Computer Science and Engineering - GMVR LAB

## 大綱

- ◎ 簡介
- ◎ 相關文獻研究
- ◎ 使用權重模數方法進行嵌入與擷取
- ◎ 使用權重技巧的高品質偽裝學演算法
- ◎ 實驗結果與分析
- ◎ 結論與未來工作

# 簡介

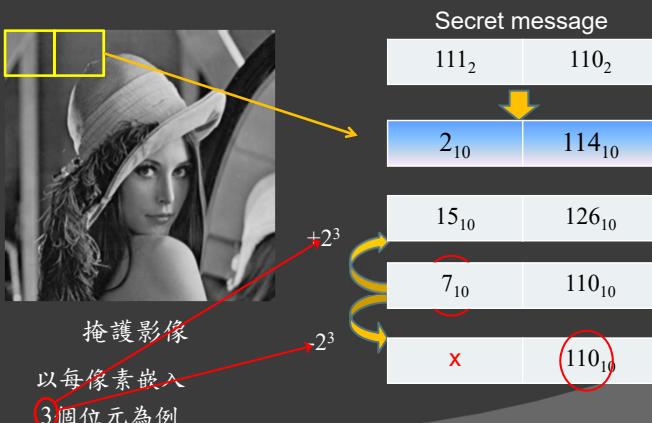


## 相關文獻

OPAP	1 bpp	2 bpp	3 bpp	4 bpp
MSE	0.5	1.5	5.5	21.5

- [Chan2004]

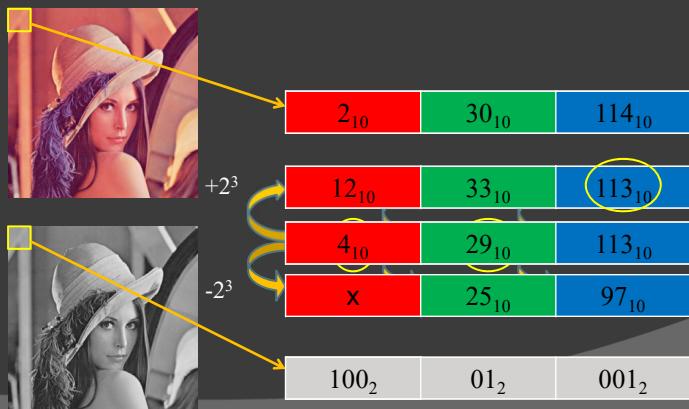
C. K. Chan and L. M. Cheng, "Hiding Data in Images by Simple LSB Substitution," *Pattern Recognition*, Vol. 37, No. 3, pp. 469-474, 2004.



## 相關文獻研究(2)

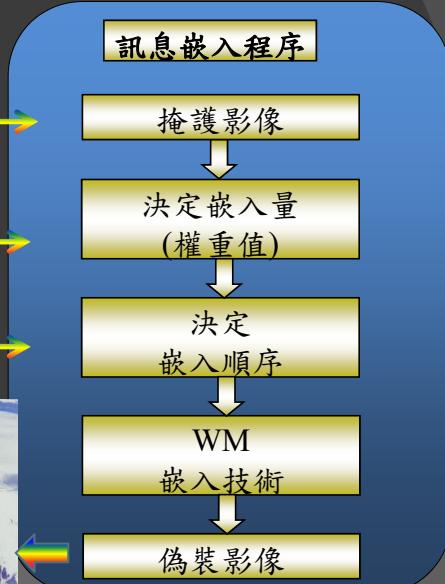
- [Yu2007]

Y. H. Yu, C. C. Chang, and I. C. Lin, "A New Steganographic Method for Color and Grayscale Image Hiding," Computer Vision and Image Understanding, Vol. 107, pp. 183-194, 2007.



5

## 權重模數方法嵌入程序



6

## 權重模數方法擷取程序



effective  
weight



010101010001010



7

## 演算法嵌入程序-實例嵌入說明(1)

$p_1'$	$p_2'$	$p_3'$	立元訊息 秘密訊息 $d = 6$
11	15	23	1, 2, 3)

$$R \left( \begin{array}{c} 10 \cdot 15 \cdot 23 \\ 1 \cdot 2 \cdot 3 \end{array} \right) = \left[ \sum_{i=1}^3 (p_i \times w_i) \right] \bmod 2^n$$

嵌入3位元訊息

$$\begin{aligned} R &= (10 \times 1 + 15 \times 2 + 23 \times 3) \bmod 2^3 \\ &= 109 \bmod 8 = 5 \end{aligned}$$

$$s = (d - R) \bmod 8 = (6 - 5) \bmod 8 = 1$$

查找權重和餘數與差值對照表得知

將  $p_1 + 1 \rightarrow (p_1', p_2', p_3') = (11, 15, 23)$   
即可完成嵌入動作

8

此處MSE為一個嵌入單位整體之變動量

$$\begin{aligned} \text{MSE} &= (p_1' - p_1)^2 + (p_2' - p_2)^2 + (p_3' - p_3)^2 \\ &= (11-10)^2 + (15-15)^2 + (23-23)^2 \\ &= 1^2 + 0^2 + 0^2 = 1 \end{aligned}$$

$p_1'$	$p_2'$	$p_3'$
11	15	23

$$R' \left( \begin{array}{ccc} p_1', p_2', p_3' \\ w_1, w_2, w_3 \end{array} \right) = \left[ \sum_{i=1}^3 (p_i' \times w_i) \right] \bmod 2^n$$

$$\begin{aligned} R' &= (11 \times 1 + 15 \times 2 + 23 \times 3) \bmod 2^3 \\ &= 110 \bmod 8 = \underline{6} \end{aligned}$$

6即為嵌入之秘密訊息，取出成功!

9

## 演算法嵌入程序-實例嵌入說明(2)

$p_1'$	$p_2'$	$p_3'$	立元訊息	秘密訊息	$d = 6$
11	16	23	1, 3, 5		

$$R \left( \begin{array}{ccc} 10 & 15 & 23 \\ 1 & 3 & 5 \end{array} \right) = \left[ \sum_{i=1}^3 (p_i \times w_i) \right] \bmod 2^n$$

↑ 嵌入3位元訊息

$$\begin{aligned} R &= (10 \times 1 + 15 \times 3 + 23 \times 5) \bmod 2^3 \\ &= 170 \bmod 8 = 2 \end{aligned}$$

$$s = (d - R) \bmod 8 = (6 - 2) \bmod 8 = 4$$

查找權重和餘數與差值對照表得知

將  $p_1 + 1, p_2 + 1 \rightarrow (p_1', p_2', p_3') = (\underline{11}, \underline{16}, \underline{23})$   
即可完成嵌入動作

10

## 演算法擷取程序-實例擷取說明(2)

$$\begin{aligned} \text{MSE} &= (p_1' - p_1)^2 + (p_2' - p_2)^2 + (p_3' - p_3)^2 \\ &= (11-10)^2 + (16-15)^2 + (23-23)^2 \\ &= 1^2 + 1^2 + 0^2 = 2 \end{aligned}$$

$p_1'$	$p_2'$	$p_3'$
11	16	23

$$R' \left[ \begin{array}{c} p_1', p_2', p_3' \\ w_1, w_2, w_3 \end{array} \right] = \left[ \sum_{i=1} (p_i' \times w_i) \right] \bmod 2^n$$

→

$$\begin{aligned} R' &= (11 \times 1 + 16 \times 3 + 23 \times 5) \bmod 2^3 \\ &= 174 \bmod 8 = \underline{6} \end{aligned}$$

6即為嵌入之秘密訊息，取出成功!

11

並非只權重組合  
(1, 2, 3), (2, 1, 3)

MSE =

權重值必須慎重選取！

一八八九

$$\begin{aligned} \text{MSE} &= (p_1' - p_1)^2 + (p_2' - p_2)^2 + (p_3' - p_3)^2 \\ &= (9-10)^2 + (10-15)^2 + (23-23)^2 \\ &= (-1)^2 + 1^2 + 0^2 = 2 \end{aligned}$$

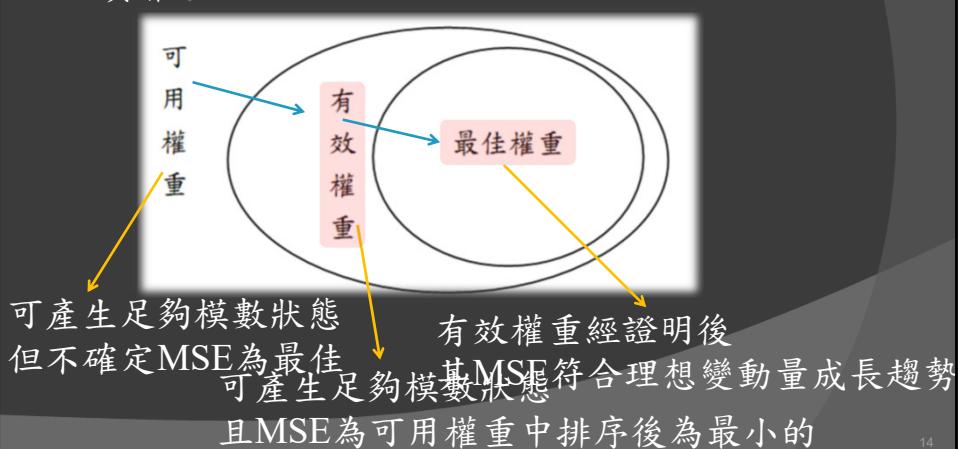
12

# 使用權重技巧的高品質偽 裝學演算法

13

## 決定嵌入權重值-權重分類

- ◎ 由嵌入時造成的變動量多寡可區分不同性質權重：



14

3像素嵌入3bits訊

$w_1$	$w_2$	$w_3$	MSE
1	2	3	8
1	2	5	8
1	3	6	8
1	3	10	8
1	5	6	8
1	5	10	8
2	3	7	8
2	5	9	8
3	6	7	8
3	6	9	8
3	7	10	8
3	9	10	8
5	6	7	8
5	6	9	8
5	7	10	8
5	9	10	8
.	.	.	.

1	2	4	9
1	3	4	9
1	4	5	9
1	4	6	9
1	4	10	9
2	3	4	9
2	4	5	9
2	4	7	9
2	4	9	9
3	4	6	9
3	4	7	9
3	4	9	9
3	4	10	9
.	.	.	.

1	2	6	10
1	2	10	10
1	3	3	10
1	3	5	10
1	3	7	10
1	3	8	10
1	3	9	10
1	5	5	10
1	5	7	10
1	5	8	10
1	5	9	10
1	6	6	10
1	6	10	10
2	2	3	10
2	2	5	10
2	2	7	10
2	2	9	10
2	3	6	10
.	.	.	.

## 決定嵌入權重值

- ◎ 目標1：找到可用權重值
- ◎ 必要條件：
  - ✓ 滿足嵌入狀態數(達到需求嵌入量)

## 決定嵌入權重值-滿足嵌入狀態數(1)

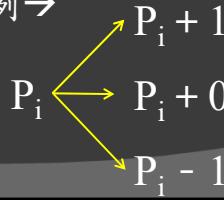
- ◎ 定義n為欲嵌入訊息位元數

則可決定像素值的變動範圍為c  $P_1 \quad P_2 \quad P_3$

$$\Rightarrow (2^{\lceil c+1 \rceil}) \times (2^{\lceil c+1 \rceil})^n \times (2^{\lceil c+1 \rceil}) \geq 2^n$$

Ex : n=4  $\rightarrow$  c=1  
3<sup>3</sup>=27  $\geq$  2<sup>4</sup>=16

以c=1為例  $\rightarrow$



17

## 決定嵌入權重值-滿足嵌入狀態數(2)

- ◎ 權重值選取範圍

- 可能無法找到有效權重或最佳權重
- 甚至無法求得可用權重

$$\Rightarrow w_r = \lceil 2^n / c \rceil \quad \text{Ex : } n=4 \cdot c=1 \\ \rightarrow w_r = \text{ceil}(2^4 / 1) = 16$$

- ◎ 在最差(大)的變動量下可以達到模數的最大值，增加其造成更多狀態數的機會

18

## 決定嵌入權重值

for  $w_1 = 1 \sim w_r$   
for  $w_2 = 1 \sim w_r$        $w_r = 16$       “ $2^4 \times 3^3 \times 16^3$ ”  
for  $w_3 = 1 \sim w_r$

for  $c_1 = -c \sim c$   
for  $c_2 = -c \sim c$        $c = -1 \sim 1$   
for  $c_3 = -c \sim c$

檢查  $(w_1 \times c_1 + w_2 \times c_2 + w_3 \times c_3)$  是否有  $2^n$  種不同狀態

→ 得到可用權重

19

## 決定嵌入權重值

◎ 目標2：對MSE排序後求出有效權重

→ 使用者決定之嵌入量n後

經由  $(2\lceil c+1 \rceil)^3 \geq 2^n$  &  $w_r = \lceil 2^n/c \rceil$  得到  
可用權重。

再對MSE作排序即可得到有效權重

◎ 目標3：驗證其變動量是否為最佳權重

20

## 理想的變動量增長趨勢

1. 三個像素值中沒有改變任何一個  
 $1^2+0^2+1^2=2 \leftarrow$

$$\rightarrow C_0^3 = 1\text{種狀態 } \boxed{0 \ 0 \ 0} \rightarrow 0^2+0^2+0^2=0$$

2. 三個像素中只有一個改變1(+1 or -1)

$$\rightarrow C_1^3 \times 2 = 6\text{種狀態 } \boxed{0 \ 0 \ 1} \rightarrow 0^2+0^2+1^2=1$$

3. 三個像素值中只有兩個改變1(+1 or -1)

$$\rightarrow C_2^3 \times 2 \times 2 = 12\text{種狀態 } \boxed{0 \ 1 \ 0}$$

1	0	1
1	0	-1
-1	0	1
-1	0	-1
1	1	0
1	-1	0
0	1	0
-1	-1	0
:	:	:
:	:	:

21

## 理想的變動量增長趨勢

沒有改變像素

1種

0	0	0
---	---	---

只改變1個像素

6種

0	0	1
0	0	-1
0	1	0
0	-1	0
1	0	0
-1	0	0

只改變2個像素

12種

1	0	1
1	0	-1
-1	0	1
-1	0	-1
1	1	0
1	-1	0
0	1	0
-1	-1	0
:	:	:
:	:	:

6	6	6
:	:	:
7	7	7
:	:	:
8	8	8
:	:	:

22

## 決定嵌入權重值-最佳變動量

嵌入位元數 (每嵌入單位= 三個像素)	所需狀態數
3 bits	$2^3=8$ 種
4 bits	$2^4=16$ 種
5 bits	$2^5=32$ 種
:	:
:	:
12 bits	$2^{12}=4096$ 種

0	0	0
0	0	1
0	0	-1
0	1	0
0	-1	0
:	:	:
:	:	:
8	8	8

23

## 可用權重分析-最佳權重

►以3個像素藏入4位元秘密訊息為例 (1, 2, 6)

MSE	個數	最佳化使 用個數	最佳化 MSE計算	實際使用 個數	實際MSE 計算
<i>MSE&lt;最佳化使用個數</i>					
0	1	1	0 <i>0×1</i>	1	0
1	6	6	6 <i>1×6</i>	6	6
2	12	9	18 <i>2×9</i>	9	18
使用個數 MSE總和				24	24

24

## 可用權重分析-有效權重(非最佳權重)

➤以3個像素藏入6位元秘密訊息為例 (2, 8, 15)

MSE	個數	最佳化使 用個數	最佳化 MSE計算	實際使用 個數	實際MSE 計算
0	1	1	0	1	0
1	6	6	6	6	6
2	12	12	24	12	24
3	8	8	24	8	24
4	6	6	24	6	24
5	24	24	120	21	105
6	24	7	42	10	60
使用個數			240	243	
MSE總和					

25

## 決定嵌入權重值-有效權重與最佳權重

n bits in 3 pixels	權重值組合 (有效權重)	組數	MSE	最佳化MSE
n=3	(1, 2, 3)	18	8	8
n=4	(1, 2, 6)	10	24	24
n=5	(1, 4, 10)	4	77	74
n=6	(2, 8, 15)	4	243	240
n=7	(1, 10, 25)	4	771	758
n=8	(3, 21, 40)	3	2452	2372
n=9	(29, 42, 49)	23	7750	7575
n=10	(6, 67, 87)	3	24749	24018
n=11	(59, 135, 293)	23	80526	76270
n=12	(195, 322, 459)	13	251752	242157

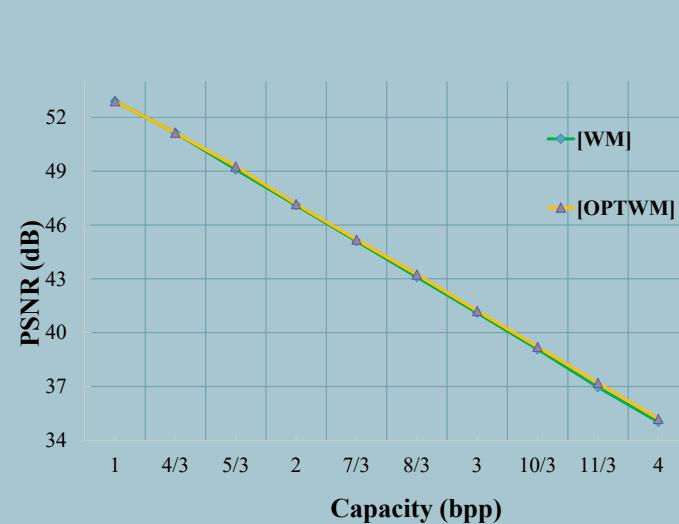
26

## OPTWM v.s. WM 理論數據分析

Capacity (bpp)	OPTWM		WM	
	PSNR(dB)	單位像素MSE	PSNR(dB)	單位像素MSE
1	52.9020	$8/2^3*(1/3)$	52.9020	$8/2^3*(1/3)$
4/3	51.1411	$24/2^4*(1/3)$	51.1411	$24/2^4*(1/3)$
5/3	49.2612	$74/2^5*(1/3)$	49.0886	$77/2^5*(1/3)$
2	47.1617	$240/2^6*(1/3)$	47.1078	$243/2^6*(1/3)$
7/3	45.1774	$758/2^7*(1/3)$	45.1036	$771/2^7*(1/3)$
8/3	43.2333	$2372/2^8*(1/3)$	43.0892	$2452/2^8*(1/3)$
3	41.2009	$7575/2^9*(1/3)$	41.0978	$7750/2^9*(1/3)$
10/3	39.1996	$24018/2^{10}*(1/3)$	39.0694	$24749/2^{10}*(1/3)$
11/3	37.1918	$76270/2^{11}*(1/3)$	36.9560	$80526/2^{11}*(1/3)$
4	35.1846	$242157/2^{12}*(1/3)$	35.0159	$251752/2^{12}*(1/3)$

27

## OPTWM v.s. WM - 圖表



28

## 應用於彩色影像

- ◎ 彩色影像上以一個像素的三個色彩頻道為一個嵌入單位
  - 除了降低變動量外
  - 考量人眼視覺的特性，對權重值組合做適當的對應

29

$w_1=2$	$w_2=1$	$w_3=6$
0	0	0
0	1	0
1	0	0
1	1	0
-1	0	1
0	-1	1
0	0	1
0	1	1
-1	0	-1
0	-1	-1
0	0	-1
0	1	-1
1	0	-1
-1	-1	0
-1	0	0
0	-1	0

$MSE \text{ of } w_1 \rightarrow (1^2+1^2+(-1)^2+(-1)^2+1^2+(-1)^2+(-1)^2) \div 16 = 0.4375$

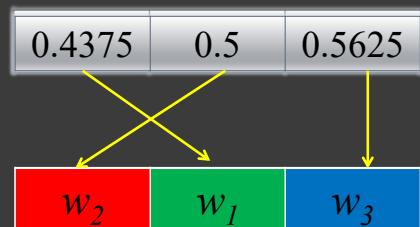
$MSE \text{ of } w_2 \rightarrow (1^2+1^2+(-1)^2+1^2+(-1)^2+1^2+(-1)^2+1^2) \div 16 = 0.5$

16種狀態

$MSE \text{ of } w_3 \rightarrow (1^2+1^2+1^2+1^2+(-1)^2+(-1)^2+(-1)^2+(-1)^2+(1)^2) \div 16 = 0.5625$

30

## 應用於彩色影像-權重值對應



31

## 實驗結果與分析

- ◎ 灰階影像實驗結果與分析
- ◎ 彩色影像實驗結果與分析

32

## 灰階影像實驗數據-嵌入隨機產生訊息(1)

在掩護影像中嵌入1000次隨機產生的秘密訊息

Cover Image	n (bits for 3 pixels)	Our Algorithm (WM)	K (bpp)	[Chan2004]
		PSNR (dB)		PSNR (dB)
Lena	3	52.9023	1	51.1410
	6	47.1078	2	46.3699
	9	41.0979	3	40.7271
	12	35.0163	4	34.8062
Baboon	3	52.9032	1	51.1414
	6	47.1084	2	46.3691
	9	41.0965	3	40.7253
	12	35.0168	4	34.8021

33

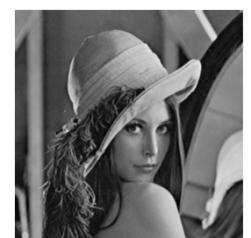
## 灰階影像實驗數據-嵌入隨機產生訊息(2)

在掩護影像中嵌入1000次隨機產生的秘密訊息

Cover Image	n (bits for 3 pixels)	Our Algorithm (WM)	K (bpp)	[Chan2004]
		PSNR (dB)		PSNR (dB)
Jet	3	52.9031	1	51.1405
	6	47.1079	2	46.3700
	9	41.0974	3	40.7273
	12	35.0159	4	34.8065
Scene	3	52.9023	1	51.1410
	6	47.1078	2	46.3702
	9	41.0973	3	40.7270
	12	35.0157	4	34.8060

34

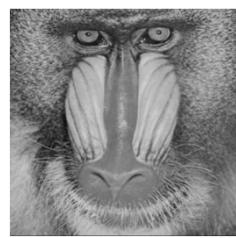
## 灰階掩護影像



Lena (512 × 512)



Jet (512 × 512)



Baboon (512 × 512)



Scene(512 × 512) 35

3像素嵌3位元 3像素嵌12位元

以Lena分別嵌入 1bpp ~ 4bpp (1)

秘密訊息：由亂數產生01字串



1bpp (52.9023 dB)



2bpp (47.1078 dB)

WM : 52.9020 dB  
OPTWM : 52.9020 dB

WM : 47.1078 dB  
OPTWM : 47.1617 dB

36

## 以Lena分別嵌入 1bpp ~ 4bpp (2)



3bpp (41.0979 dB)



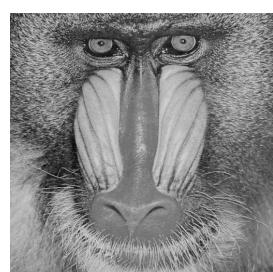
4bpp (35.0163 dB)

WM : 41.0978 dB  
OPTWM : 41.2009 dB

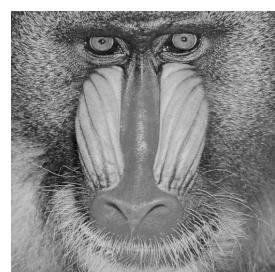
WM : 35.0159 dB  
OPTWM : 35.1846 dB

37

## 以Baboon分別嵌入 1bpp ~ 4bpp (1)



1bpp (52.9032 dB)



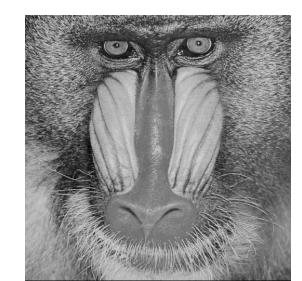
2bpp (47.1084 dB)

WM : 52.9020 dB  
OPTWM : 52.9020 dB

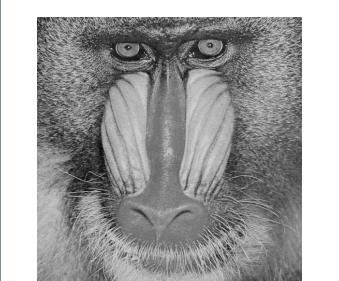
WM : 47.1078 dB  
OPTWM : 47.1617 dB

38

## 以Baboon分別嵌入 1bpp ~ 4bpp (2)



3bpp (41.0965 dB)



4bpp (35.0168 dB)

WM : 41.0978 dB  
OPTWM : 41.2009 dB

WM : 35.0159 dB  
OPTWM : 35.1846 dB

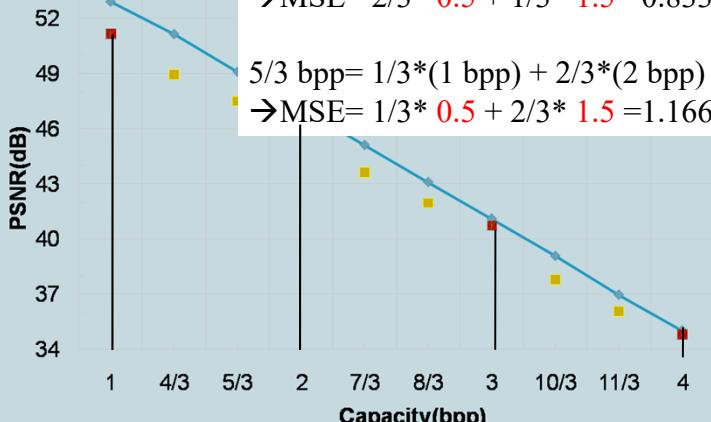
39

## 灰階影像

OPAP	1 bpp	2 bpp	3 bpp	4 bpp
MSE	0.5	1.5	5.5	21.5

$$4/3 \text{ bpp} = 2/3 * (1 \text{ bpp}) + 1/3 * (2 \text{ bpp}) \\ \rightarrow \text{MSE} = 2/3 * 0.5 + 1/3 * 1.5 = 0.8333$$

$$5/3 \text{ bpp} = 1/3 * (1 \text{ bpp}) + 2/3 * (2 \text{ bpp}) \\ \rightarrow \text{MSE} = 1/3 * 0.5 + 2/3 * 1.5 = 1.1666$$



40

## 彩色影像實驗數據

-相同嵌入量(8 bpp)的PSNR比較

提出方法與[Yu2007]各嵌入100次隨機產生之秘密訊息，  
相同嵌入量下的PSNR (dB)比較

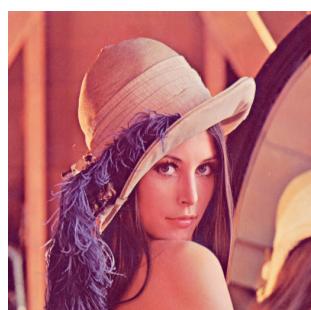
Host Image (解析度)	[WM]	[Yu2007]
Airplane(512x512)	43.090	41.914
Baboon(512x512)	43.087	41.946
House(512x512)	43.088	41.932
Lena(512x512)	43.091	41.943
Peppers(512x512)	43.090	41.682
Sailboat(512x512)	43.088	41.938

約大於1.1dB

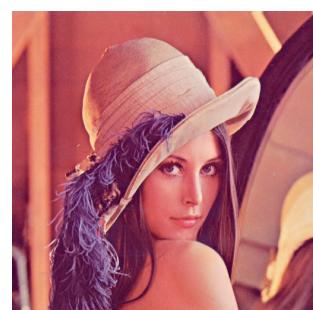
41

## 彩色影像嵌入實驗結果-Lena

嵌入8bits 隨機產生的秘密訊息



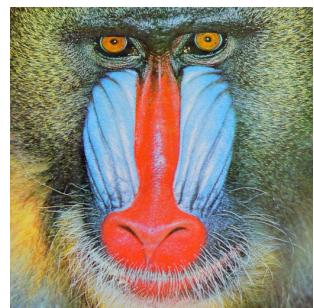
掩護影像



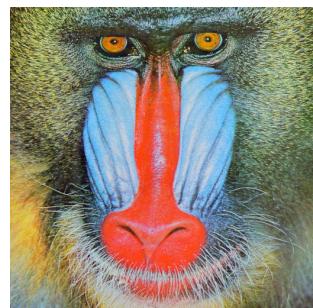
偽裝影像

42

## 彩色影像嵌入實驗結果-Baboon



掩護影像



偽裝影像

43

## 相似PSNR下的藏量比較

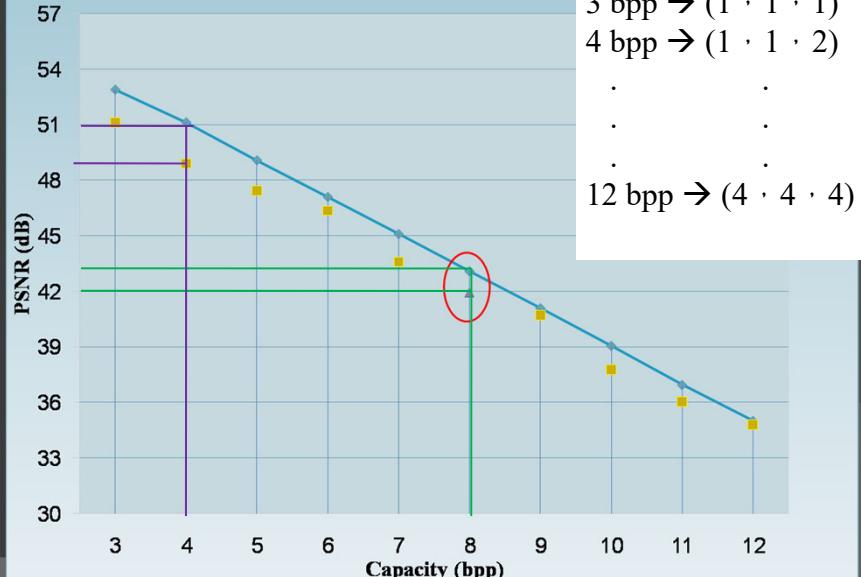
提出方法與IVu2007名嵌入100%隨機產生之秘密訊息  
8 bpp → PSNR : 43.0892 dB  
9 bpp → PSNR : 41.0978 dB

$$\begin{aligned} & 43.0892 \times + 41.0978(1-x) = 41.932 * 1.005 \\ \rightarrow & x=0.52 \\ \rightarrow & 8 * 0.52 + 9 * 0.48 = 8.46 (\text{bpp}) \end{aligned}$$

House(512×512)	42.093	41.921
Lena(512×512)	42.116	41.925
Peppers(512×512)	41.783	41.661
Sailboat(512×512)	42.091	41.923

44

## 彩色影像實驗分析



結論與未來工作

46

## 結論

使用權重模數方法會有以下特點

1. 極佳的視覺品質 ( $1\text{bpp} \rightarrow 52.9\text{dB}$ ,  $4\text{bpp} \rightarrow 35.02\text{dB}$ )
2. 優於現存文獻結果 ( $1\text{bpp} \rightarrow 51.14\text{dB}$ ,  $4\text{bpp} \rightarrow 34.8\text{dB}$ )
3. 可延伸至彩色影像 (嵌入單位三個一組、變動量對應)
4. 預知各種嵌入量下的變動量
5. 滿足更精細的嵌入量 (非整數位元嵌入量)
6. 高安全性
7. 盲擷取

47

## 未來工作

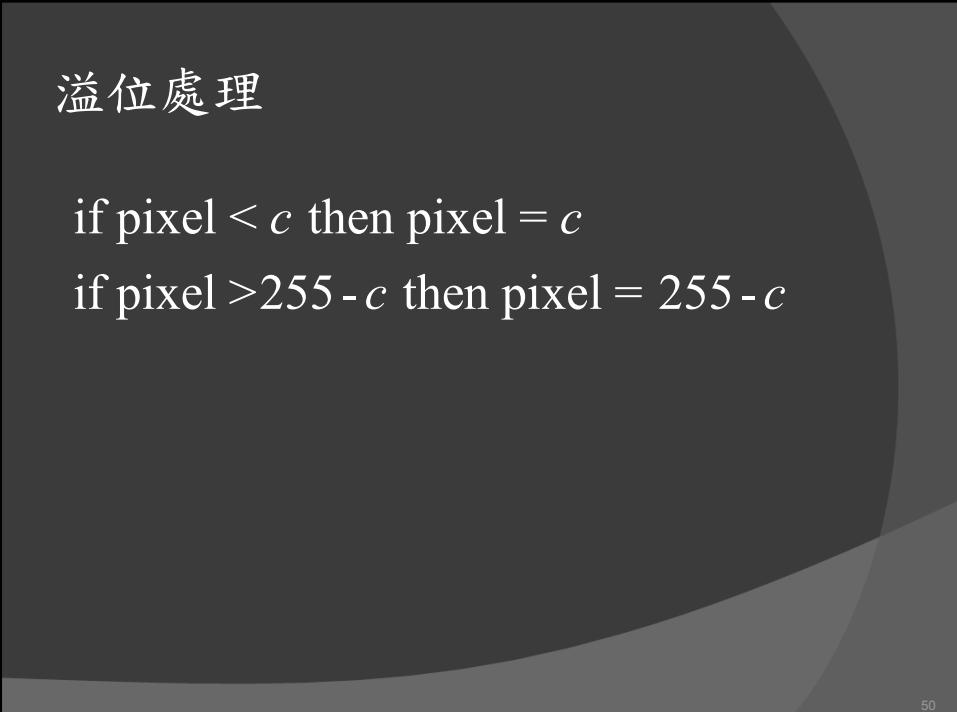
- ◎ 所有嵌入率下都找到最佳權重
- ◎ 以不同的像素個數做為一組嵌入單位
- ◎ 藉由理論分析來找出最佳權重
- ◎ 擴展演算法到訊息認證或是影像浮水印等領域

48



*Thank You*

49



## 溢位處理

if pixel <  $c$  then pixel =  $c$

if pixel >  $255 - c$  then pixel =  $255 - c$

50

n=3時，權重和餘數與訊息差值對照表

差值	w <sub>1</sub> =1	w <sub>2</sub> =2	w <sub>3</sub> =3
0	0	0	0
1	+1	0	0
2	0	+1	0
3	0	0	+1
4	-1	0	-1
5	0	0	-1
6	0	-1	0
7	-1	0	0

51

n=3時，權重和餘數與訊息差值對照表

差值	w <sub>1</sub> =1	w <sub>2</sub> =3	w <sub>3</sub> =5
0	0	0	0
1	+1	0	0
2	-1	+1	0
3	0	+1	0
4	+1	+1	0
5	0	0	+1
6	+1	0	+1
7	-1	0	0

52

## 演算法嵌入程序-實例嵌入說明

$p_1$	$p_2$	$p_3$
1	68	213

立元訊息為訊息  $d = 4$

$(1, 2, 3)$

$$\rightarrow (2^{\lceil c \rceil} + 1)^3 \geq 2^n \quad R(p_1, p_2, p_3) = \left[ \sum_{i=1}^3 (p_i \times w_i) \right] \bmod 2^n$$

$\because$  此時  $c=1$ ， $p_1$  是可能產生溢位的像素值

$\therefore$  調整像素，使得  $p_1=1$

$$R = (1 \times 1 + 2 \times 68 + 3 \times 213) \bmod 2^3 \\ = 776 \bmod 8 = 0$$

$$s = d - R = 4 - 0 = 4$$

查找權重和餘數與差值對照表得知

$$\text{將 } p_1 - 1, p_3 - 1 \rightarrow (p_1', p_2', p_3') = (0, 68, 212)$$

即可完成嵌入動作

53

## 演算法擷取程序-實例擷取說明

接續上例，此時偽裝像素為

$p_1$	$p_2$	$p_3$
0	68	212

$$R' (p_1', p_2', p_3') = \left[ \sum_{i=1}^3 (p_i' \times w_i) \right] \bmod 2^n$$



$$R' = (1 \times 0 + 2 \times 68 + 3 \times 212) \bmod 2^3 \\ = 772 \bmod 8 = 4$$

4即為嵌入之秘密訊息，取出成功!

54

## 應用於彩色影像 - 以3 bpp為例

$w_1 = 1$	$w_2 = 2$	$w_3 = 3$
0	0	0
1	0	0
0	1	0
0	0	1
1	0	1
0	0	-1
0	-1	0
-1	0	0
<b>MSE =</b>		
$(1^2 + 1^2 + (-1)^2) \div 8 = 0.375$		
		$(1^2 + (-1)^2) \div 8 = 0.25$
		$(1^2 + 1^2 + (-1)^2) \div 8 = 0.375$

55

$w_1$	$w_2$	$w_3$	MSE				
1	2	6	24	1	2	7	26
1	2	10	24	1	2	9	26
2	3	6	24	1	3	5	26
2	3	10	24	1	3	7	26
2	5	6	24	1	3	9	26
2	5	10	24	1	4	7	26
2	6	7	24	1	4	9	26
2	6	9	24	1	5	7	26
2	7	10	24	1	5	9	26
2	9	10	24	3	4	5	26
				3	5	6	26
				3	5	7	26
				3	5	9	26
				3	5	10	26

56

### 簡介(3)

以3個 pixel 嵌入3 bits並使用權重組合  
(1,2,3)為例

$W_1=1$	$W_2=2$	$W_3=3$	餘數狀態
0	0	0	0
1	0	0	1
0	1	0	2
0	0	1	3
1	0	1	4
0	0	-1	5
0	-1	0	6
-1	0	0	7

57

### 決定嵌入權重值

以3個 pixel 嵌入3 bits並使用權重組合  
(1,3,5)為例

$W_1=1$	$W_2=13$	$W_3=15$	餘數狀態
0	0	0	0
1	0	0	1
-1	1	0	2
0	1	0	3
1	1	0	4
0	0	1	5
1	0	1	6
-1	0	0	7

58

## 決定嵌入權重值

- ◎ 設定像素變動範圍  $C$
- ◎ 設定三個權重值範圍各為  $1 \sim w_r$
- ◎ 檢查  $w_1 \times c_1 + w_2 \times c_2 + w_3 \times c_3$  是否能產生  $2^n$  種不同的狀態

→ 得到可用權重

59