國立交通大學

資訊學院　資訊學程

碩士論文

逐張顯像式的多張影像分享

Sharing using image-by-image decoding

研 究 生：葉姿敏

指導教授：林志青 博士

中 華 民 國 一百零一 年 七 月

逐張顯像式的多張影像分享

Sharing using image-by-image decoding

研 究 生：葉姿敏　　Student：Tzu-Min Yeh

指導教授：林志青 博士　　Advisor：Dr. Ja-Chen Lin

國 立 交 通 大 學

資訊學院 資訊學程

碩 士 論 文

A Thesis

Submitted to College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Computer Science

July 2012

Hsinchu, Taiwan, Republic of China

中 華 民 國 一百零一 年七月

逐張顯像式的多張影像分享

學生：葉姿敏　　指導教授：林志青 博士

國立交通大學　資訊學院　資訊學程 碩士班

摘　　要

傳統影像分享由機密影像產生 n 張分存來保護機密影像: 任意 k 份分存可還原機密影像，但少於 k 份分存則無法還原。機密影像分享常使用在單一的影像上面，但實際生活上一個專案團隊經常會同時處理多張機密影像。員工集體開會時，如果我們可以使用逐張顯像式解碼，則開會的時候，如果有些人遲到了，已經到達現場的部分員工可以先行解碼一些較不重要的影像。經過解碼運算，已到場的員工即開始討論那些較不重要的影像。然而若是更多員工到達後，則可以進一步將更重要的影像解碼，接下來才開始討論這些較為重要的影像。本論文提出三種多張影像分享方式來得到逐張顯像式解碼:　1)基本型; 2)加權型;3)解碼需董事會到場型。

Sharing using image-by-image decoding

student：Tzu-Min Yeh　　Advisors：Dr. Ja-Chen Lin

Degree Program of Computer Science

National Chiao Tung University

ABSTRACT

This thesis proposes three forms of sharing for multiple images. Traditional *(k,n)* secret image sharing protects a secret image by splitting the secret image into *n* noise-like shadows. Any *k* of the *n* shadows can reconstruct the secret image, but fewer than *k* shadows cannot. However, for multiple images, there are several forms (choices) to decode the images of an image set, as can be seen in this thesis. An application of this research is that, in a meeting, if some people are late, once the number of attendants has reached a threshold, some less important images can be decoded and discussed first. When more and more people attend the meeting, then the more important images can be decoded and discussed. This thesis proposes three forms to unveil the images: 1) a basic form in which the decoding is according to the security level of each image; 2) a sharing scheme whose image-by-image decoding scheme uses weights; and 3) a sharing scheme whose image-by-image decoding needs the attendance of the essential shares.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Professor Ja-Chen Lin, for his invaluable assistance and technical advice during my graduate study. I would like to thank Dr. Kun-Yuan Chao, Dr. Lee Shu-Teng Chen, Dr. Sian-Jheng Lin, and Mr. Suiang-Shyan Lee for their discussion and suggestions. I would like to thank all members of Computer Vision Laboratory for their help. Finally, I wish to express my deep appreciation to my family for their encouragement and support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Traditionally, (*k*, *n*) secret image sharing can protect a secret image by splitting the secret image into *n* noise-like shadows. Any *k* of *n* shadows can reconstruct the secret image, but fewer than *k* shadows cannot. The concept of secret sharing was introduced independently in 1979 by Shamir [1] and Blakley [2]. Thien and Lin [4] proposed a secret image sharing method based on the (k, n) threshold scheme. Each generated shadow image is small in their method, which is advantageous in the transmission and hiding of shadow images.

For multiple images, there are several ways to decode the images. Tsai et al. [11] proposed a method to share multiple secrets in digital images. Each pair of cover images shares a different secret, and different secret images can be held by several participants. Feng [12] developed multiple images sharing methods based on polynomial approach. This thesis proposes three kinds of sharing and decoding for multiple images in an image set.

1.2 Thesis Overview

The thesis proposes three kinds of sharing for image-by-image decoding. The framework of the thesis is shown in Figure 1.1, and a brief overview of each proposed method is described in the following subsections.

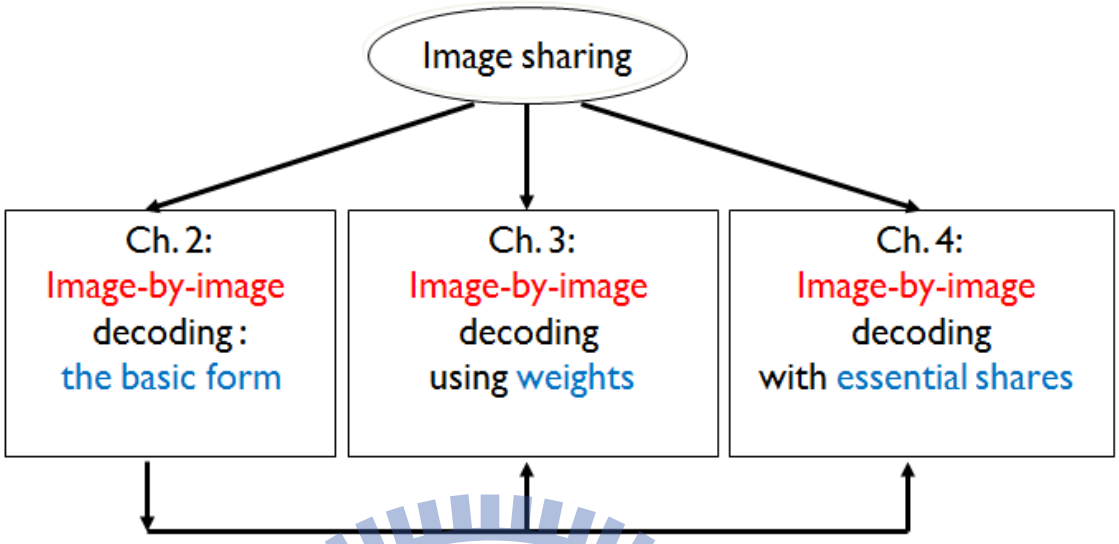


Figure 1.1. Framework of thesis.

1.2.1 Image-by-image decoding: the basic form

Chapter 2 proposes a sharing method whose decoding is image-by-image. The decoding is according to the different security levels of images. When the numbers of received shadows reach certain pre-specified thresholds, the secret images using these thresholds are unveiled.

1.2.2 Image-by-image decoding using weights

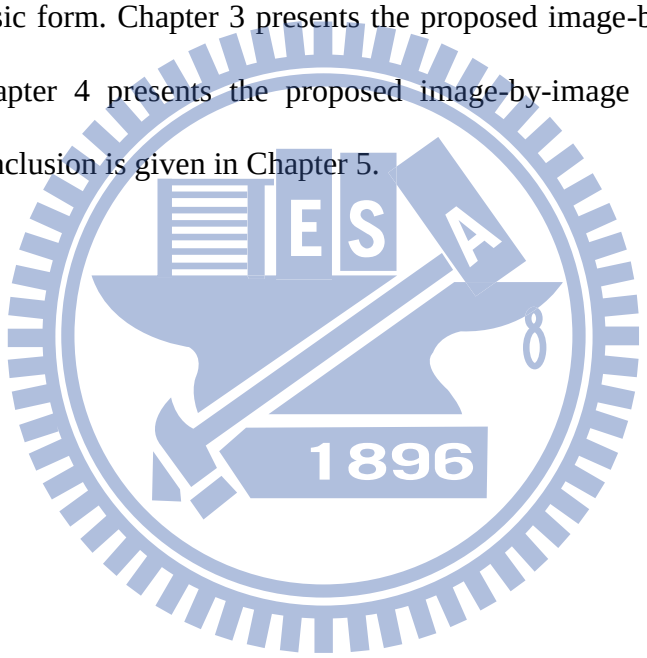
Chapter 3 proposes a sharing method whose decoding is also image-by-image. However, the decoding is according to the total of the received weights. Images of higher security level are unveiled when the total sum of received weights reach a larger threshold.

1.2.3 Image-by-image decoding with essential shares

Chapter 4 proposes a sharing method whose decoding is still image-by-image. However, a necessary condition for decoding is that all essential shares must be collected. Images of higher security level are unveiled when the number of received shadows reach certain pre-specified thresholds, as long as all essential shares are collected .

1.3 Thesis Organization

In the rest of this thesis, Chapter 2 presents the proposed image-by-image decoding using the basic form. Chapter 3 presents the proposed image-by-image decoding using weights. Chapter 4 presents the proposed image-by-image decoding with essential shares. A conclusion is given in Chapter 5.



Chapter 2

Image-by-image decoding: the basic form

This chapter proposes a sharing method whose decoding is image-by-image. Section 2.1 presents the proposed method. Section 2.2 presents the experimental results. Finally, Section 2.3 summarizes this chapter.

2.1. The Proposed Method

2.1.1 Encoding

Input: n important image $\{I_1, I_2, \ldots I_n\}$; n cover images; and T positive integer parameters $t_1 < t_2 < \ldots < t_T$.

Output: The n JPEG stego codes.

Step 1: Rearrange the n images $\{I_1, I_2 \ldots I_n\}$ so that I_1 has minimum security level; I_2 has higher security level than $I_1 \ldots$; and I_n has maximum security level. Then use JPEG to compress each image.

Step 2: According to security level of the n images, divide n images into T groups. For each j=1, 2...T, the security level of j^{th} group are lower than $j+1^{th}$ group.

Step 3: For each image of the j^{th} group, use (t_j, n) Secret image sharing to split the image into n shares. For each $i=1, \ldots, n$; collect the i^{th} share of each image and call the collection the i^{th} shadow. Denote the i^{th} share of j^{th} group by $(c_i)_j$.

Step 4: Do Step3 for all groups (j=1, 2…, T).

Step 5: Use JPEG data hiding method [8] to hide the n shadows in the n JPEG code of the n cover images, respectively.

2.1.2 Decoding

When any t_1 of the n JPEG stego codes are received, the t_1 shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of first group. When any t_2 of the n JPEG stego codes are received, the t_2 shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of first group and second group. When any t_T of the n JPEG stego codes are received, the t_T shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of all groups.

2.2 Experiments

2.2.1 Experimental Results

First Experiment

Input：Four 180×180 important images {Boat, Jet, Lake, Houses} which are shown in Figure 2.1. Four 512×512 Cover images {Barbara, Baboon, Lena, Peppers} which are shown in Figure 2.2. Two positive integer parameters: {2, 4}.

Output: The four JPEG stego codes whose decompressed images are shown in Figure 2.4.

Step 1: According to secret levels of the four images, divide four images into two groups.

Step 2: Use (2,4) Secret image sharing to split images of lower secret level group into *c*1~*c*4.

Step 3: Use (4,4) Secret image sharing to split images of higher secret level group into *c*1~*c*4.

Step 4: The first shadow is formed by binding together the shares $c1$of groups 1 and 2, and the second shadow is formed by binding together the shares $c2$ of groups 1 and 2. The third and fourth shadows are formed likewise by binding together the shares $c3$ and $c4$, respectively. Figure 2.3 shows the four shadows, each shadow has size (15,345/4) + (16,263 /4) + (22,264/2) + (16,948/2) = 27,507 bytes.

Step 5: Use JPEG data hiding method [8] to hide the 4 shadows in the 4 JPEG code of the 4 cover images, respectively. Number of hidden bits and hiding ratio of four cover images are shown in Table 2.1.

When any two JPEG stego codes are received, we can reconstruct all images in the lower secret level group, and when all four JPEG stego codes are received, we can reconstruct all images.

Figure 2.1. Four 180×180 important images {Boat, Jet, Lake, Houses}.

Figure 2.2. Four 512×512 Cover images {Barbara, Baboon, Lena, Peppers} utilized to cover the four important images.

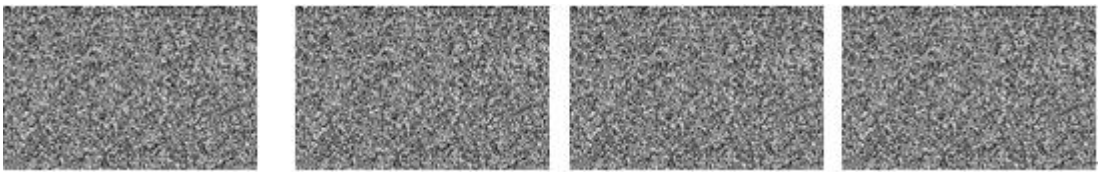


Figure 2.3. Four shadows obtained by binding together the shares *c*1~ *c*4, respectively, of groups 1 and 2.

Figure 2.4. The decompressed images of the four JPEG stego codes. (a) The 37.75-dB stego-image Barbara, (b) The 37.5-dB stego- image Baboon, (c) The 37.63-dB stego-image Lena, (d) The 37.79-dB stego- image Peppers.

Cover Image	**Jpeg file size (bits) of cover image**	**Number of hidden bits**	**Hiding ratio**
Barbara	792,008	220,056	27.78%
Baboon	926,144	220,056	23.76%
Lena	742,872	220,056	29.62%
Peppers	762,680	220,056	28.85%

Table 2.1.Number of hidden bits and hiding ratio of four cover images

Second Experiment

Input：Six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly} which are shown in Figure 2.5. Six 512×512 cover images {Barbara, Baboon, Lena, Peppers, Fish, Couple} which are shown in Figure 2.6. Three positive integer parameters: 4, 5, and 6.

Output: The six JPEG stego codes whose decompressed images are shown in Fig. 2.8.

Step 1: According to secret level of six images, divide the six images into three groups.

Step 2: Use (4,6) secret image sharing to split images of lower secret level group into $c1$~$c6$.

Step 3: Use (5,6) secret image sharing to split images of moderate secret level group into $c1$~$c6$, and use (6,6) secret image sharing to split images of highest level group into $c1$~$c6$.

Step 4: The first shadow is formed by binding together the shares $c1$ of groups 1, 2 and 3. The second shadow is formed by binding together the shares $c2$ of groups 1, 2 and 3. Repeat this for $c3$-$c6$. Figure 2.7 shows the six shadows. Each shadow has size (15,345/6) + (16,263/6) + (23,127/6) + (22,264/5) + (16,948/5) + (17,007/4) = 21,213 bytes.

Step 5: Use JPEG data hiding method [8] to hide the 6 shadows in the 6 JPEG code of the 6 cover images, respectively. Number of hidden bits and hiding ratio of six cover images are shown in Table 2.2.

When any four JPEG stego codes are received, we can reconstruct all images in lower secret level group. When any five JPEG stego codes are received, we can reconstruct images in higher secret level group, and any six JPEG stego codes are

received, we can reconstruct images in highest secret level group.

Figure 2.5. The six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly} used in Experiment 2.

Figure 2.6. The six 512×512 Cover images {Barbara, Baboon, Lena, Peppers, Fish, Couple} which are utilized to cover the six important images

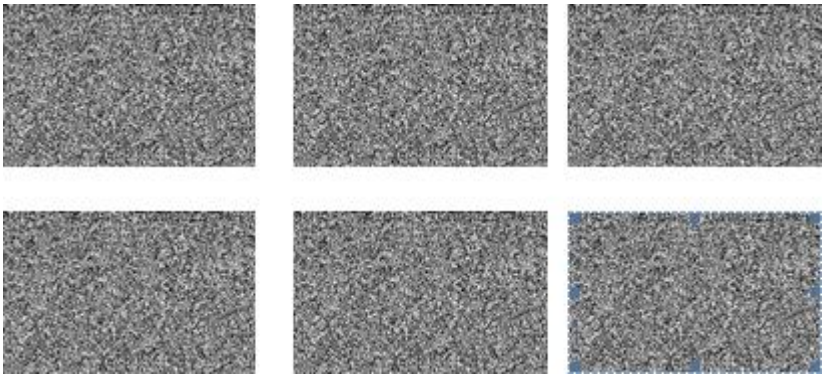


Figure 2.7. The six shadows obtained by binding together the *c*1~ *c*6 shares, respectively, of groups 1-3.

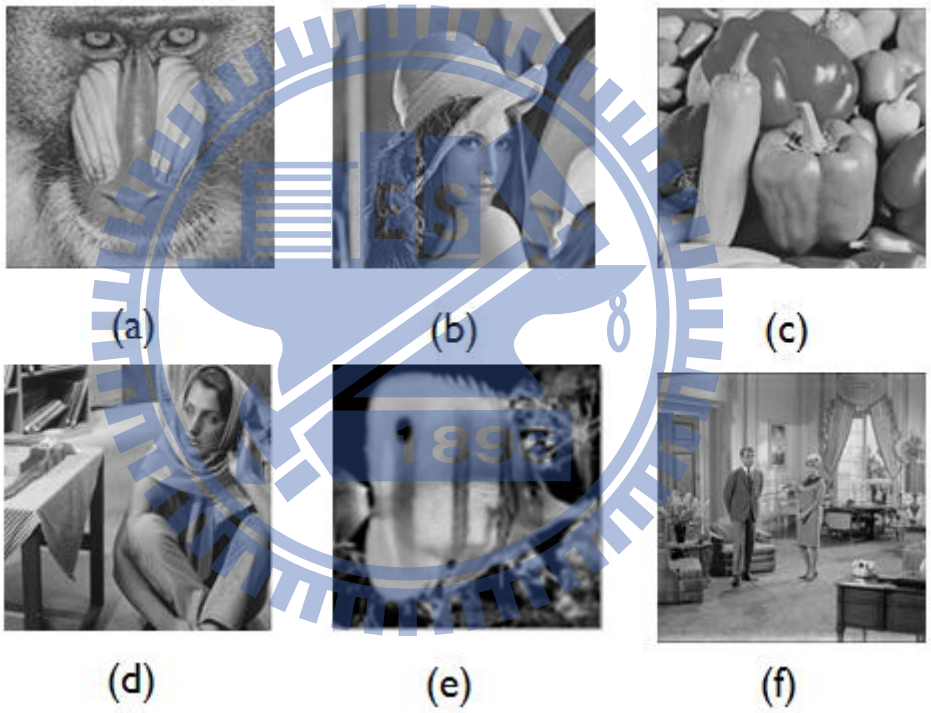


Figure 2.8. The six JPEG stego codes. (a) The 36.8-dB stego- image Baboon, (b) The 36.82-dB stego- image Lena, (c) The 37.09-dB stego-image Peppers, (d) The 36.81-dB stego-image Barbara, (e) The 37.58-dB stego-image Fish, (f) The 36.85-dB stego-image Couple.

Cover Image	Jpeg file size (bits) of cover image	Number of hidden bits	Hiding ratio
Barbara	792,008	169,704	21.42%
Baboon	926,144	169,704	18.32%
Lena	742,872	169,704	22.8%
Peppers	762,680	169,704	22.25%
Fish	610,504	169,704	27.79%
Couple	544,000	169,704	31.19%

Table 2.2. Number of hidden bits and hiding ratio of six cover images

2.3 Summary

This chapter proposes a sharing method whose decoding is image-by-image. The decoding is according to the different security levels of images. When the numbers of received shadows reach certain pre-specified thresholds, the secret images using these thresholds are unveiled.

Chapter 3

Image-by-image decoding using weights

This chapter proposes a sharing method whose decoding is also mage-by-image. However, the decoding is according to the total sum of the received weights. Section 3.1 presents the proposed method. Section 3.2 presents the experimental results. Section 3.3 summarizes this chapter.

3.1. Proposed Method

3.1.1 Encoding

Input: n important images $\{I_1, I_2, \ldots I_n\}$; n cover images; T positive integer parameters $t_1 < t_2 < \ldots < t_T$; T positive integer parameters $W_1 \le W_2 \le \ldots \le W_C$ where $W_1 + W_2 + W_3 + \ldots + W_C = n$.

Output: n JPEG stego codes.

Step 1: Rearrange the n images $\{I_1, I_2 \ldots I_n\}$ so that I_1 has minimum security level; I_2 has security level higher than I_1 ; …; and I_n has maximum security level. Then use JPEG to compress each image.

Step 2: According to security level of the n images, divide the n images into T groups. For each j=1, 2...T, the security level of j^{th} group is lower than the $j+1^{th}$ group.

Step 3: For each image of the j^{th} group, use (t_j, n) Secret image sharing to split the image into n shares. For each i=1,…,n; collect the i^{th} share of each image and call the collection the i^{th} shadow. Denote the i^{th} share of j^{th} group by $(c_i)_j$.

Step 4: Do Step 3 for all groups (j=1,2…., T).

Step 5: Divide the n cover images into T cover groups, assign the T weights $\{W_1 <$

$W_2 < \ldots < W_C\}$ to these T cover groups, respectively.

Step 6: Use the JPEG data hiding method [8] to hide the $n \times W_1/n = W_1$ shadows in the cover group whose weight is W_1; and to hide the $n \times W_2/n = W_2$ shadows in the cover group whose weight is W_2; and so on. Finally, use the JPEG data hiding method [8] to hide the $n \times W_C/n = W_C$ shadows in the cover group whose weight is W_C.

3.1.2 Decoding

The decoding is according to the total sum of the received weights. When total sum of the received weights reach t_1, the t_1 shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of first group. When total sum of the received weights reach t_2, the t_2 shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of second group. And so on. Finally, when total sum of the received weights reach $t_{T,}$ the t_T shadows can be extracted from the JPEG stego codes, which are used to reconstruct all images of all groups.

3.2 Experiments

3.2.1 Experimental Results

Input：There are six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly} which are shown in Figure 3.1. There are six 512×512 Cover images {Barbara, Baboon, Lena, Peppers, Fish, Couple} which are shown in Figure 3.2. The input also includes T=3 positive integer parameters $W_1 = 1 \leq W_2 = 2 \leq W_3 = 3$ where $W_1 + W_2 + W_3 = n = 6$.

Output: The six JPEG stego codes whose decompressed images are shown in Figure 3.4.

Step 1: Use JPEG to compress each image.

Step 2: According to the secret level of the n=6 secret images, divide the six secret

images into T=3 groups. The security level of j^{th} group is lower than the j+1th group (true for each j=1, 2, 3).

Step 3: Use (4,6) Secret image sharing to split the code of the secret images in lower secret level group into shares *c*1~*c*6. Use (5,6) Secret image sharing to split the code of the moderate secret level group into shares *c*1~*c*6. *U*se (6,6) Secret image sharing to split images of highest level group into shares c1~c6.

Step 4: The first shadow is created by binding together the shares c1 of groups 1, 2 and 3. The second shadow is created by binding together the shares c2 of groups 1, 2 and 3. Repeat this for c3-c6. Figure 3.3 shows six shadows. Each shadow has size (15,345/6) + (16,263/6) + (23,127/6) + (22,264/5) + (16,948/5) + (17,007/4) = 21,213 bytes.

Step 5: Divide the six cover images into three groups, Then assign weights 1, 2 and 3 to each group, respectively.

Step 6: Use JPEG data hiding method [8] to hide the 6×1/6=1 shadow in the group whose weight is 1, and to hide the 6×2/6=2 shadows in the cover group whose weight is 2, and to hide the 6×3/6=3 shadows in the group whose weight is 3. Number of hidden bits and hiding ratio of six cover images are shown in Table 3.1.

When the total of the received weights reach four, we can reconstruct the images {Butterfly} in lower secret level group. When the total of the received weights reach five, we can also reconstruct the images {Lake, Houses} in moderate secret level group; and when the total of the received weights reach six, we can also reconstruct the images {Boat, Jet, Map} in highest secret level group.

Figure 3.1. The six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly}

Figure 3.2. The six 512×512 Cover images {Baboon, Fish, Barbara, Peppers, Lena, Couple} which are utilized to cover the six important images

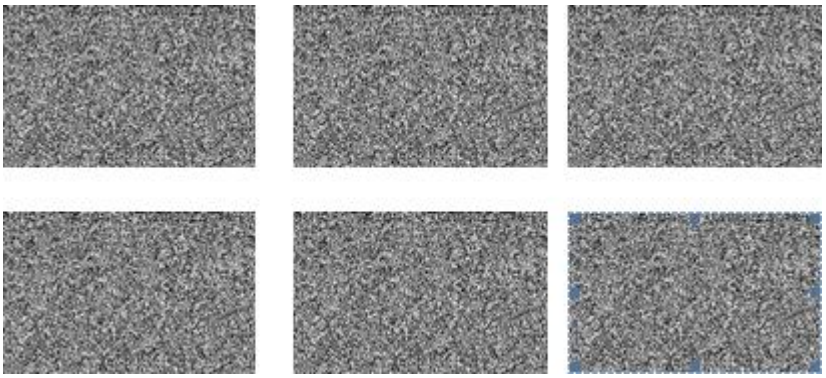


Figure 3.3. Six shadows which bind together, respectively, the *c*1~ c6 shares of group 1, group 2, and group 3.

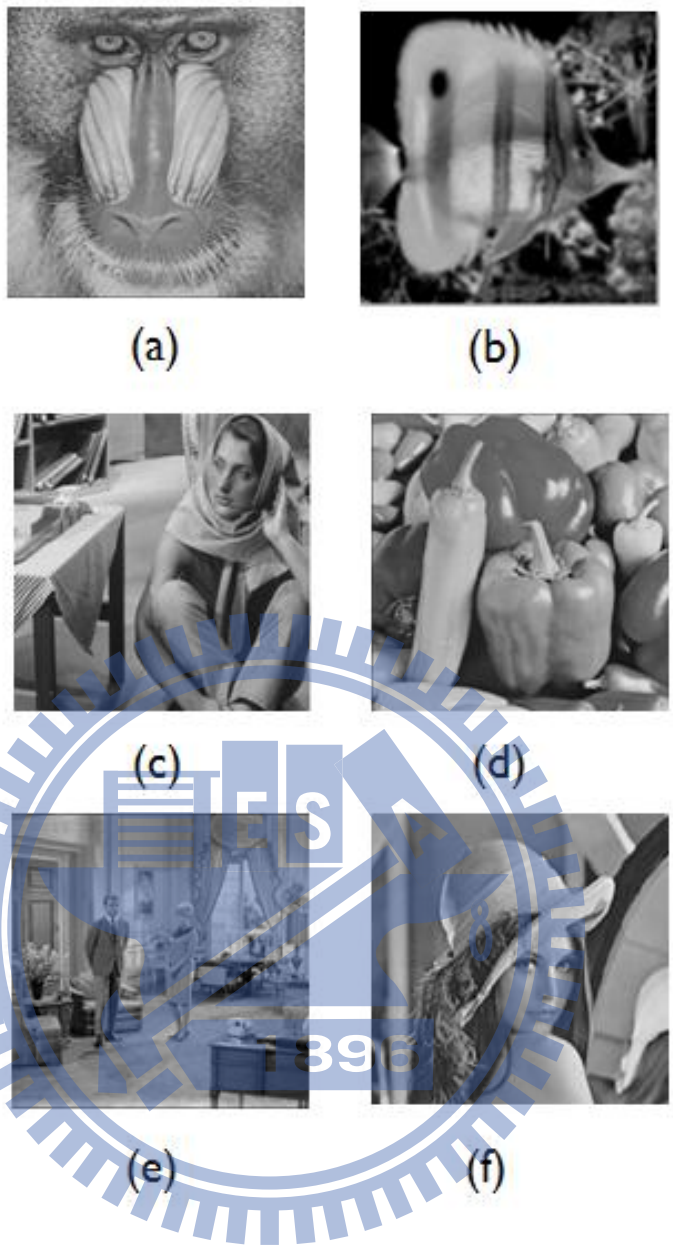


Figure 3.4. Figure 3.4. Decompressed Images of the six JPEG stego codes. Group (a-b) has weight being 1; and it is formed of the 38.11-dB stego image Baboon and the 38.78-dB stego image Fish. Group (c-d) has weight being 2; and it is formed of the 37.98-dB stego image Barbara and the 37.15-dB stego image Peppers. Group (e-f) has weight being 3; and it is formed of the 35.52-dB stego image Couple and the 35.55-dB stego image Lena.

(a)

(b)

(c)

Figure 3.5. When the total of the received weights reach 4, 5, and 6, we can obtain the secret images in (a), (a-b) and (a-c), respectively.

Cover Image	Jpeg file size (bits) of cover image	Number of hidden bits	Hiding ratio
Barbara	792,008	84,848	10.71%
Baboon	926,144	84,848	9.16%
Lena	742,872	169,704	22.84%
Peppers	762,680	169,704	22.25%
Fish	610,504	254,552	41.69%
Couple	544,000	254,552	46.79%

Table 3.1. Number of hidden bits and hiding ratio of six cover images

3.3 Summary

This chapter proposes a sharing method whose image-by-image decoding is according to the total sum of the received weights. Images of higher security level are not unveiled unless the total sum of the received weights reaches a larger threshold. However, images of lower security level can be unveiled when the total sum of the received weights reaches a lower threshold.

Chapter 4

Image-by-Image decoding with essential shares

This chapter proposes a sharing method whose decoding is image-by-image, too. However, a necessary condition for decoding is that all specified essential shares must be collected. Section 4.1 presents the proposed method. Section 4.2 presents the experimental results. Finally, Section 4.3 summarizes this chapter.

4.1. The Proposed Method

4.1.1 Encoding

Input: n important image $\{I_1, I_2, \ldots I_n\}$; n cover images; and T positive integer parameters $t_1 < t_2 < \ldots < t_T$; a positive integer parameters E $(1 \le E \le n)$

Output: n JPEG stego codes.

Step 1: Rearrange the n images $\{I_1, I_2 \ldots I_n\}$ so that I_1 has minimum security level; I_2 has security level higher than I_1 ; …; and I_n has maximum security level. Then use JPEG to compress each image.

Step 2: According to security level of the n images, divide the n images into T groups. For each j=1, 2...T, the security level of j^{th} group is lower than the $j+1^{th}$ group.

Step 3: For each image of the j^{th} group, use (t_j, n) Secret image sharing to split the image into n shares. For each i=1,…, n; collect the i^{th} share of each image and call the collection the i^{th} shadow. Denote the i^{th} share of j^{th} group by $(c_i)_j$.

Step 4: Do Step 3 for all groups (j=1, 2…, T).

Step 5: Use key to encrypt all shadows, and use (E, E) Secret image sharing to split the key into E shares.

Step 6: Use JPEG data hiding method [8] to hide the n shadows in the n JPEG code of the n cover images, respectively.

Step 7: Select *E* essential cover images, and use JPEG data hiding method [8] to hide the *E* shares in the selected cover images, respectively.

4.1.2 Decoding

When *E* "essential" shares are collected, use inverse sharing to extract from these *E* shares the *key* which are needed in decryption. Then, when any t_1 of the n JPEG stego codes are received, the t_1 shadows can be extracted from the JPEG stego codes; then, after decryption using the key, the t_1 shadows can reconstruct all images of first group. When any t_2 of the n JPEG stego codes are received, the t_2 shadows can be extracted from the JPEG stego codes; then, after decryption using the key, the t_2 shadows can reconstruct all images of first group and second group. When any t_T of the n JPEG stego codes are received, the t_T shadows can be extracted from the JPEG stego codes; then, after decryption using the key, we can reconstruct all images of all groups.

4.2 Experiments

The input includes six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly}, which are shown in Figure 4.1, and the six 512×512 cover images {Barbara, Baboon, Lena, Peppers, Fish, Couple} shown in Figure 4.2. The input also includes three positive integer parameters {t_1 =4, t_2 =5, t_3 =6} and a positive integer parameter *E*=3. The output will be the n=6 JPEG stego codes whose decompressed images are shown in Figure 4.4. The steps to create these six JPEG stego codes are as follows:

First, according to the secret levels of the six secret images, divide the six images into three groups. Then use (4,6) secret image sharing to split images of lower secret

level group into *c*1~*c*6. Use (5,6) secret image sharing to split images of higher secret level group into *c*1~*c*6*;* use (6,6) secret image sharing to split images of highest level group into *c*1~*c*6.

Then the first shadow is created by binding together the shares *c*1 of groups 1, 2 and 3. The second shadow is created by binding together the shares *c*2 of groups 1, 2 and 3. Repeat this for *c*3-*c*6. Figure 4.3 shows the six shadows. Each shadow has size (15,345/6) + (16,263/6) + (23,127/6) + (22,264/5) + (16,948/5) + (17,007/4) = 21,213 bytes.

Then use key to encrypt all shadows. Then use JPEG data hiding method [8] to hide the 6 shadows in the 6 JPEG codes of the 6 cover images, respectively. Then use (3, 3) sharing to split the key into three shares. Then select three essential cover images, and use JPEG data hiding method [8] to hide the three shares of the key in the three JPEG codes of the selected cover images, respectively. Number of hidden bits and hiding ratio of six cover images are shown in Table 4.1.

When *E*=3 essential stego JPEG codes are collected, we can use (3,3) inverse sharing to extract the key which will be needed in the decryption later. After that, when any one of the remaining 6-3=3 JPEG stego codes are received, we can reconstruct all images in lower secret level group. When any two of the remaining 6-3=3 JPEG stego codes are received, we can reconstruct images in moderate secret level group. And when all JPEG stego codes are received, we can reconstruct images in highest secret level group.

Figure 4.1. The six 180×180 important images {Boat, Jet, Lake, Houses, Map, Butterfly}.

Figure 4.2. The six 512×512 cover images {Barbara, Baboon, Lena, Peppers, Fish, Couple} which are utilized to cover the six important images

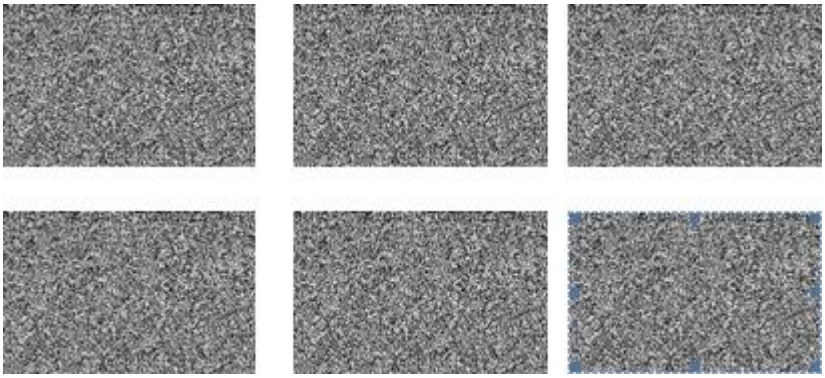


Figure 4.3. Six shadows which bind together, respectively, the *c*1~ *c*6 shares of group 1, group 2, group3.

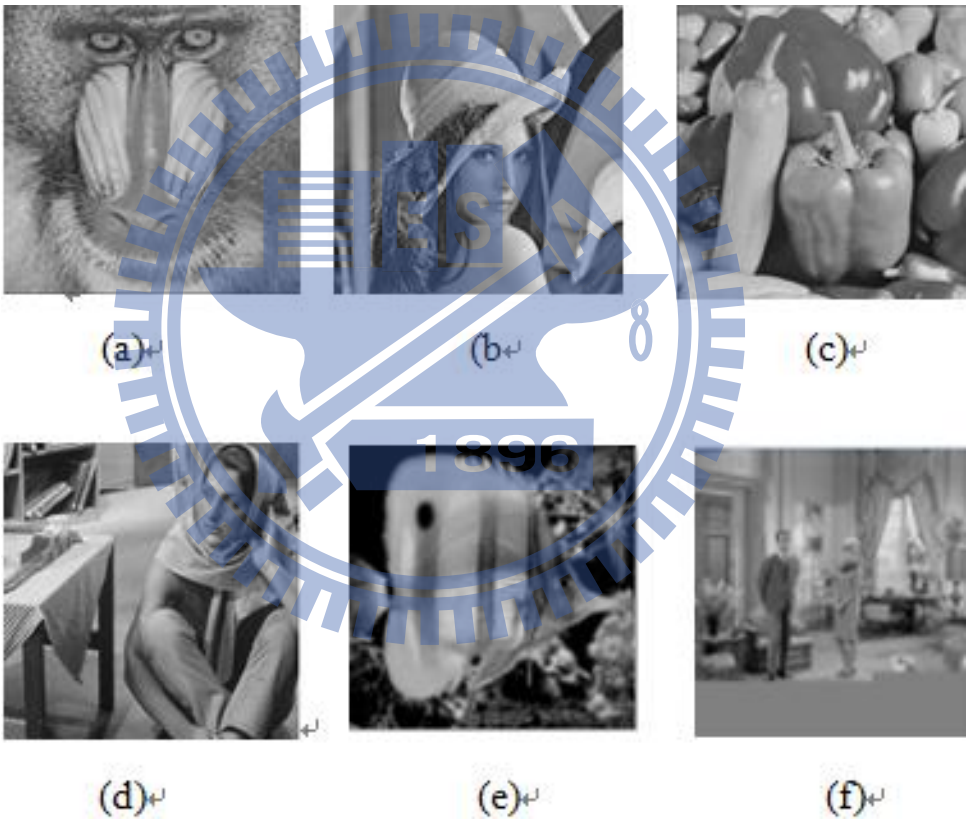


Figure 4.4. The six JPEG stego codes. (a) The 37.80-dB stego-image Baboon, (b) The 37.77-dB stego-image Lena, (c) The 38.26-dB stego-image Peppers, (d) The 37.97-dB stego-image Barbara, (e) The 38.75-dB stego-image Fish, (f) The 38.15-dB stego-image Couple.

(a)

(b)

(c)

Figure 4.5. When the total of the received shadows reach 4, 5, and 6, we can obtain the secret images in (a), (b) and (c), respectively; as long as the received stego codes include the three essential stego codes which shared and hid the key.

Cover Image	Jpeg file size (bits) of cover image	Number of hidden bits	Hiding ratio
Barbara	792,008	169,704	21.42%
Baboon	926,144	169,704	18.32%
Lena	742,872	169,704	22.84%
Peppers	762,680	169,712	22.25%
Fish	610,504	169,712	27.79%
Couple	544,000	169,712	31.19%

Table 4.1. Number of hidden bits and hiding ratio of six cover images

4.3 Summary

This chapter proposes a sharing method whose decoding is also image-by-image. However, a necessary condition for decoding is that all E essential shares must be collected. Images of distinct security level are unveiled when the number of received shadows reach certain pre-specified thresholds, as long as all essential shares are collected.

Chapter 5

Conclusions

We proposed in the thesis three kinds of sharing and decoding for multiple images. The proposed methods include the basic form of image-by-image decoding in Chapter 2, an image-by-image decoding using weights in Chapter 3, an image-by-image decoding with essential shares in Chapter 4.

In Chapter 2, the decoding is according to the security level of each image. When the numbers of received shadows reach certain pre-specified thresholds, the secret images using those thresholds are unveiled.

In Chapter 3, the decoding is according to the total of the received weights. Images of lower security level are unveiled when the total sum of received weights reaches a smaller threshold. Images of higher security level are unveiled when the total sum of received weights reaches a larger threshold;

In Chapter 4, the decoding requires that all essential shares must show up. When all essential shares appear, then images of a pre-specified security level will be unveiled if and only if the number of received shadows (including the count of the essential shadows) reaches the corresponding threshold.

Although all three methods are related to sharing of multiple images, the three methods are according to different criteria to do the unveiling of the secrets. So the readers can have more options to deal with the secret images that they have. Among the three options, the final choice should be according to the intention of the company's boss, i.e. according to the manner that a company's boss deals with the company's

secrets.

References

[1] A. Shamir, “How to share a secret,” *Commun. ACM,* Vol. 22(11), pp. 612–613, 1979.

[2] G. R. Blakley, “Safeguarding cryptographic keys,” in *Proc. AFIPS 1979 National Computer Conference*, Vol. 48, pp. 313–317, 1979.

[3] M. Naor and A. Shamir, “Visual cryptography,” Advances in Cryptography – EUROCRYPT ‘94, *Lect. Notes Comput. Sci.*, Vol. 950, pp. 1–12, 1995.

[4] C. C. Thien and J. C. Lin, “Secret image sharing,” *Comput. Graphics,* Vol. 26(5), pp. 765–770, 2002

[5] S. J. Shyu, S. Y. Huang, Y. K. Lee, R. Z. Wang, and K. Chen, "Sharing multiple secrets in visual cryptography," *Pattern Recognition*, Vol. 40(12), pp. 3633-3651, 2007.

[6] J. B. Feng, H. C. Wu, C. S. Tsai, Y. F. Chang, and Y. P. Chu, "Visual secret sharing for multiple secrets," *Pattern Recognition*, Vol. 41(12), pp. 3572–3581, 2008.

[7]R.Z. Wang and S.J. Shyu, “Scalable secret image sharing,” S*ignal Processing: Image Communication*, Vol. 22(4), pp. 363–373, 2007.

[8] L. S. T. Chen, S. J. Lin and J. C. Lin, "Reversible JPEG-Based Hiding Method with High Hiding-Ratio," *Intern. J. Pattern Recognition and Artificial Intelligence (IJPRAI)*, Vol. 24(3), pp. 433−456, 2010.

[9]W. P. Fang, "Friendly progressive visual secret sharing," *Pattern Recognition*, Vol. 41(4), pp. 1410–1414, 2008.

[10] K. H. Hung, Y. J. Chang, and J. C. Lin, "Progressive sharing of an image," *Optical Engineering*, Vol.47(4), p. 047006, 2008.

[11] C.S. Tsai, C.C. Chang, T.S. Chen, Sharing multiple secrets in digital images, *Journal of Systems and Software*, Vol. 64(2), pp.163−167, 2002.

[12] J. B. Feng, H. C. Wu, C. S. Tsai, and Y. P. Chu, "A new multi-secret images sharing scheme using Largrange's interpolation," *Journal of Systems and Software*, Vol. 76(3), pp. 327−339, 2005.

[13] L. S. T. Chen and J. C Lin, "Multithreshold progressive image sharing with compact shadows," *Journal of Electronic Imaging*, Vol. 19(1), p. 013003, 2010.

[14] S. J. Lin, L. S. T. Chen, and J. C Lin, "Fast-weighted secret image sharing," *Optical Engineering*, Vol. 48(7), p. 077008, 2009.

[15] L. S. T. Chen, W. K. Su, and J. C. Lin, "Secret image sharing based on vector quantization," *International Journal of Circuits, Systems and Signal Processing*, Vol. 3(3), pp. 137−144, 2009.

Vita

TZU-MIN YEH was born in Hsinchu, Taiwan, in 1981. She received her BS degree in information management from Fu Jen Catholic university, Taiwan, in 2003.