國立交通大學

資訊科學與工程研究所

碩士論文

在行動裝置上作視訊及文字之資訊隱藏研究

A Study on Techniques and Applications of Information Hiding in Videos and Messages via Mobile Phones

研 究 生：王彝琳

指導教授：蔡文祥　教授

中 華 民 國　九 十 五　年 六 月

在行動裝置上作視訊及文字之資訊隱藏研究

A Study on Techniques and Applications of Information Hiding in Videos and Messages via Mobile Phones

研 究 生：王彝琳　　Student：Yi-Lin Wang

指導教授：蔡文祥　　Advisor：Wen-Hsiang Tsai

國 立 交 通 大 學

資 訊 科 學 與 工 程 研 究 所

碩 士 論 文

A Thesis
Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

June 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年六月

A Study on Techniques and Applications of Information Hiding in Videos and Messages via Mobile Phones

Student: Yi-Lin Wang Advisor: Dr. Wen-Hsiang Tsai

Institute of Computer Science and Engineering

National Chiao Tung University

ABSTRACT

With the increase of population having mobile phones as well as the advance of high-speed transmission capability of the wireless network, more and more multimedia data are transmitted through and displayed on public networks for mobile devices. In this study, several methods for information hiding applications, such as copyright protection, covert communication, secret sharing, and text authentication, on mobile devices are proposed. In order to restrict a user account to a specific mobile phone for downloading videos, a copyright protection method is proposed. Because users do not want to see their video productions suffering from downloading by illegal users, a copyright protection method by a lossless visible watermarking technique for videos is proposed. In order to transmit large-volume secret messages on the Internet securely, a covert communication by videos is proposed. Because of the convenience of getting shares by mobile networks, storing shares in mobile devices, and transmitting shares by the SMS, a secret sharing technique by mobile devices is proposed. Furthermore, because of the large number of usages of short messages, an authentication method for short messages is proposed. Good experimental results show the feasibility of the proposed methods.

在行動裝置上作視訊及文字之資訊隱藏研究

研究生：王彝琳　　指導教授：蔡文祥 博士

國立交通大學資訊科學與工程研究所

摘要

隨著行動無線網路的進步以及手機使用率的普及，越來越多樣化的多媒體資料在網路上供使用者利用行動裝置瀏覽、使用。本論文主要針對行動裝置上普遍使用的 3GP 影音格式以及文字訊息，利用資訊隱藏和數位浮水印的技術，在行動裝置的平台上作版權保護、秘密通訊、秘密分享與驗證之研究與應用。在版權保護方面，我們提出一個架構，來做到限定一個帳號只能在一個特定的行動裝置上使用以及一個可無失真還原的數位浮水印技術來保護視訊檔案。在秘密傳輸方面，我們將秘密資訊隱藏在視訊檔案中，再透過網路來傳遞秘密資訊。在秘密分享方面，我們改善前人所提出的方法，將系統移植到行動裝置上實現。針對廣泛使用的手機簡訊，我們提出一個方法，可以驗證所接收到的簡訊內容的真偽及發送方。最後，我們提出相關的實驗結果證明所提出的方法的可行性。

ACKNOWLEDGEMENTS

The author is in hearty appreciation of the continuous guidance, discussions, support, and encouragement received from his advisor, Dr. Wen-Hsiang Tsai, not only in the development of this thesis, but also in every aspect of his personal growth.

Thanks are due to Mr. Tsung-Yuan Liu, Mr. Chih-Jen Wu, Mr. Kuo-Feng Chien, Mr. Yen-Long Chen, Mr. Kai-Li Chiang, Miss Yu-Tzu Wang, Miss Chia-Yu Hsu, and Miss Pei-Pei Chen for their valuable discussions, suggestions, and encouragement. Appreciation is also given to the colleagues of the Computer Vision Laboratory in the Institute of Computer Science and Engineering at National Chiao Tung University for their suggestions and help during her thesis study.

Finally, the author also extends her profound thanks to her family for their lasting love, care, and encouragement. She dedicates this dissertation to her parents.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

1.1. Motivation

With the increase of population having mobile phones as well as the advance of high-speed transmission capability of the wireless network, more and more multimedia data are transmitted through and displayed on public networks for mobile devices. Some websites adopt the Windows Media DRM (Digital Right Management) to place restrictions on the display of multimedia data to users. But the Windows Media DRM works only on the Windows system. Unfortunately, the Windows system is not popular on mobile phones. In order to solve this problem, we propose a technique for copyright protection of videos for displays on specified mobile phones.

Besides, more and more people like to share their videos recorded by mobile phones on the Internet. And on account of the limit storage size of the mobile phone, an user may delete the original copy of a video in the mobile phone. Because illegal users may download videos for misuses, authors may want to protect the copyright of their videos. In order to solve this problem, a lossless visible watermark technique is proposed in this study. Illegal users cannot get the correct video content without the protection key.

On the other hand, since video files can be distributed via the Internet quickly, they might be good camouflages for secret transmissions. Because of the large volumes of video files, they become suitable cover-media for carrying secret messages from one site to another with mobile phones. This application of covert communication needs an extraction program running on a mobile phone to extract the secret message from the

cover-media. We propose a covert communication method by embedding secrets into videos for use in the mobile network to solve this problem.

Because of the popularity of carrying mobile phones, sharing secret messages by mobile phones is very convenient. In order to achieve the goal of sharing secret messages via mobile phones, we also propose a method of secret message sharing on mobile phones.

Nowadays, the so-called short message service (SMS) is used frequently. But some swindlers might send short messages pretending messages sent by banks to defraud people of money. The authentication of short messages is urgently necessary. So we finally propose a message authentication method to solve this problem.

1.2. Overview of Related Works

In this study, some new techniques for information hiding applications are proposed. These applications are about embedding information within videos and text messages. A review of video and text information hiding researches will be described in Chapter 2. In addition, because the proposed information hiding and watermarking techniques are applied to H.263 videos, we will also make a review of the H.263 standard in Chapter 2.

1.3. Overview of Proposed Methods

1.3.1. Definitions of Terms

The definitions of some related terms used in this study are described in this section as follows.

1. *Cover media*: Cover media, such as images, text-type documents, or videos, are

files in which messages are embedded.

2. *Protected video*: A protected video is a video in which the content has been changed by a certain technique like information hiding, etc., for copyright protection.
3. *Watermarked video*: A watermarked video is a video in which watermarking information and a visible watermark have been embedded.
4. *Recorded token*: A recorded token is a code with the fixed length code (FLC) format used to record the watermarking information.
5. *Random key*: A random key is a user-specified input key used to randomize the content of a video.
6. *Stego-video*: A stego-video is a video in which a secret message has been embedded.
7. *Share*: A share is one of several pieces of the result after applying a steganographic technique on a piece of information.
8. *Authenticated message*: An authenticated message is a piece of short messages in which authentication signals have been embedded.

1.3.2. Brief Descriptions of Proposed Methods

1.3.2.1. Copyright Protection of Videos for Displays on Specified Mobile Phones

In this study, a copyright protection technique is proposed for displays of videos on specified mobile phones. The basic idea is to randomize the video content by a random key. Then, a copyright protection scheme by use of the International Mobile Equipment Identity (IMEI) code is applied for key transmission. The purpose of this step is to achieve the goal to place restriction on specified mobile phones for video displays.

1.3.2.2. Lossless Visible Watermarking for Copyright Protection of Videos Displayed on Mobile Phones

In this study, a lossless visible watermarking for copyright protection for videos is proposed. The idea is to take a macroblock as a unit and then change the DC value of the DCT of the macroblock by the exclusive-OR operation with the user-specified random key. In addition, a technique of embedding the watermarking information to record the positions of changed macroblocks is also proposed.

After the server program receives the key from the client, it will recover the watermarked video with the random key, remove the recorded token, and then deliver the recovered video to the client.

1.3.2.3. Covert Communication by Videos on Mobile Phones

In this study, we adopt video files as cover media for covert communication. We use the user-specified random key to randomize the secret message and transform the result into binary form. And then a process of changing the encoding format of the AC coefficients which exist originally in the video files is applied to embed the binary secret message.

And when a receiver gets stego-videos and provides the correct key, a client program will extract the embedded message and use the key to recover the message.

1.3.2.4. Secret Message Sharing on Mobile Phones by An Information Sharing Technique

In this study, we propose a secret message sharing technique to share the text-type secret message with steganographic effects on mobile phones. The first technique of secret sharing is based on the exclusive-OR operation. We use the exclusive-OR operator to generate several pieces of share data. A second technique we propose is to translate pieces of share data into simple sentences to form meaningful texts, called stego-texts. The purpose of this technique is to make each share of data meaningful to

increase the steganographic effect. By combining the two techniques together, a secret pure text can be shared among several participants.

1.3.2.5. Text Secret Authentication on Mobile Phones

In this study, we also propose a technique of short message authentication on mobile phones. First, we use the message-digest algorithm 5 (MD5) to produce a 16-byte authentication data for a short message content. And then we combine the authentication data with an identification code of the short message sender to form 1-byte authentication signal. Finally, the 1-byte authentication signal is embedded into the original short message and then the resulting authenticated message is delivered.

When the receiver gets the authentication message, a client program will help the user to determine the correctness of the content of this short message and the identity of the publisher.

1.4. Contributions

Several contributions are made in this study, as described in the following:

1. A system with server and client programs for copyright protection for displays on specified mobile phones is proposed.
2. A lossless visible watermarking technique for copyright protection is proposed.
3. A method of covert communication by videos is proposed.
4. A method of secret sharing for text messages on mobile phones is proposed.
5. A technique for authentication of short messages is proposed.

1.5. Organization of Thesis

In the remainder of this thesis, a review of related works about video data hiding, video copyright protection, visible watermarking, text secret sharing, and text message

authentication, as well as the H.263 and the IMEI standards is given in Chapter 2. In Chapter 3, the proposed method for copyright protection for videos is described. In Chapter 4, the proposed method for lossless visible watermarking is described. In Chapter 5, the proposed method of cover communication is presented. In Chapter 6, the proposed method of text message sharing is stated. In Chapter 7, the proposed method of text message authentication is described. Finally, conclusions and some suggestions for future researches are made in Chapter 8.

Chapter 2
Review of Related Works and Standards

2.1. Introduction

Nowadays, many data hiding techniques for videos or text messages have been proposed. These techniques are applied to achieve the goal of copyright protection, authentication, cover communication, secret sharing, etc. In this chapter, some related work of information hiding for videos and text message will be described. And some standards which are used in this study will be stated.

2.2. Review of Related Works

2.2.1. Review of Information Hiding Techniques for Videos

Nowadays, many techniques of data hiding in videos have been proposed [1-5]. The purpose is to achieve the functions of copyright protection, covert communication, and authentication. Chen and Tsai [1] proposed a method to hiding data in the median frequency of the DCT domain in the I-frame and the proper motion vectors in the P-frame and the B-frame. The reason why data are hidden in the median frequency regions is that the human visual system is less sensitive to the change in the high frequency regions than in the low frequency ones. And according to the property of the P-frame and the B-frame, the data are embedded in two motion vectors where the difference between them is smaller than a threshold. The threshold is selected according to a rule that the change to these two motion vectors is not sensitive to the

human visual system.

A method of embedding information in H.263 videos is proposed by Song and Liu [3] in which the information is embedded by changing the half-pixel motion estimation in every macroblock coded in INTER mode.

Many techniques about visible watermarking techniques have been proposed for copyright protection of videos [1, 4]. Chen and Tsai [1] proposed a method for embedding visible watermarks in MPEG videos. In their method, the black pixel of a watermark is embedded by changing the DC value in the DCT domain and the watermarking information is embedded in the medium frequency region in the DCT domain of the corresponding macroblock. Meng et al. [4] proposed a method for visible watermarking in MPEG videos. The method is to adjust the watermark strength dynamically according to the local content features derived in the DCT domain.

2.2.2. Review of Information Sharing of Text-type Documents

An information sharing technique of text-type documents with steganography capability is proposed by Huang and Tsai [6], which shares the text message by a quick exclusive-OR operation and disguise the noise data as one paragraph of English sentences.

2.3. Review of Standards

The most popular multimedia container format for mobile devices is the 3GP defined by the 3rd Generation Partnership Project (3GPP) for use on third-generation technology (3G) mobile phone [8]. The file extension of 3GP is .3gp or .3gp2. One of

the video compression techniques adopted in the 3GP format is ITU-T Recommendation H.263 [9].

In this study, we adopt the 3GP file with the H.263 video compression technique as our video file. We will give a brief description of the H.263 standard in Section 2.3.1. In Section 0, we will give an introduction to the International Mobile Equipment Identity (IMEI) code.

2.3.1. H.263 Video Format

An H.263 video consists of a hierarchical structure with four layers. From top to bottom the layers are: Picture, Group of Blocks (GOB), Macroblock, and Blocks. There are five standardized picture formats: CIF, 4CIF, 16 CIF, Sub-QCIF and QCIF.

The data of each picture consists of a picture header which indicates the frame type of this picture. There are various types of frames in the H.263 format, including intra-coded frame (I), predictive-coded frame (P), bi-directionally predictive-coded frame (B), and PB frame which consist of two pictures coded as one unit: one P-frame and one B-frame. The structure of an H.263 video is described in Fig 2.1.

H.263 pictures are divided into regions of macroblocks which are in order from the left border of the picture to the right border of the picture in the same row, each of which is called a Group of Blocks (GOB). Fig 2. 2 shows the arrangement of GOBs in a CIF, 4CIF, 16 CIF, Sub-QCIF, and QCIF picture format.

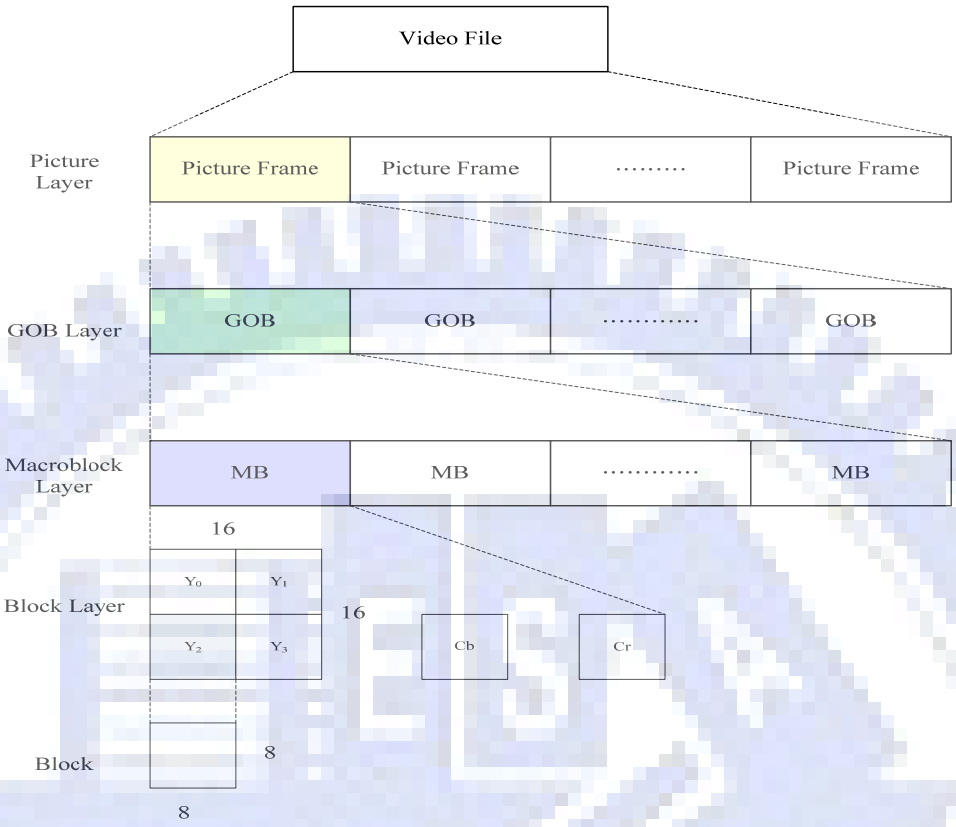


Fig 2.1 The structure of H.263 video stream

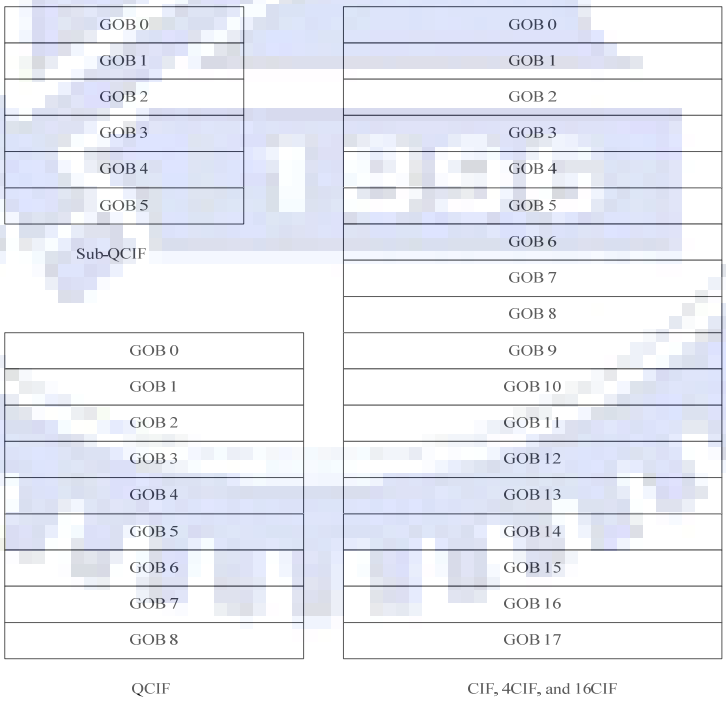


Fig 2. 2 Arrangement of GOBs in H.263 format

In the H.263 standard, not only the transform coding method similar to the JPEG is applied to reduce the spatial redundancy, but also the motion-compensation algorithm is adopted to reduce the temporal redundancy between frames. A macroblock (MB) is adopted as a motion-compensation unit. There are different types of macroblocks in each type of frames. Table 2.1 shows the relationships between frames and macroblocks.

Table 2.1 The relationships between frames and macroblocks

	Intra-coded MB	Forward-coded MB	Backward-coded MB	Bidirectionally-interpolated MB
I frame	●			
P frame	●	●		
B frame	●	●	●	●
PB frame	●	●	●	●

2.3.2. Mobile Phone Identification Number ---- International Mobile Equipment Identity (IMEI)

IMEI is a short term for International Mobile Equipment Identity. It is a unique identification number to every Global System for Mobile Communication (GSM) and Universal Mobile Telecommunications System (UMTS) mobile phone. The IMEI number is used by the GSM network to check if the mobile device is valid. If the mobile device is reported as stolen, the IMEI will be marked as invalid to access the mobile network. In other mobile networks, the IMEI number is only used to identify

the device.

The IMEI is a 15-digit number which is combined of the origin, model, and serial numbers of the device. The model and the origin comprise the initial 8-digit of the IMEI as six-digit Type Approval Code (TAC) and two-digit Final Assembly Code (FAC). The remainder of the IMEI is manufacturer-defined and a Luhn check digit at the end. The structure of IMEI is shown in Table 2.2 and the FAC is shown in Table 2.3 [11].

Table 2.2 The structure of IMEI.

1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th	15th
Type Approval Code						Final Assembly Code		Phone Serial Number						Check Digit

Table 2.3 The FAC.

Code	manufacturer	Code	manufacturer	Code	manufacturer
01	AEG	30	Ericsson	51	Sony
02		40	Siemens	60	Alcatel
07	Motorola	41		70	Sagem
40		44		75	Dancall
10	Motorola	47	Option International	80	Philips
20		50	Bosch	85	Panasonic

The JAVA program in a mobile phone can access the IMEI number by the standard API. The parameters to access the identification numbers of different manufacturers are not the same. Some parameters of different manufacturers are

shown in Table 2.4.

Table 2.4 Parameter of different manufacturers to extract IMEI

Manufacturer	Parameter
Motorola	System.getProperty("IMEI")
Siemens	System.getProperty("com.siemens.IMEI")
Sony Ericsson (Except P910 series)	System.getProperty("com.sonyericsson.imei") [10]
Sony Ericsson (P910 series)	System.getProperty("com.sonyericsson.IMEI") [10]

Chapter 3
Copyright Protection of Videos for Displays on Specified Mobile Phones

3.1. Principle of Proposed Method

The service of downloading multimedia data at users' expenses is very popular recently. Some websites impose a restriction on this service, that is, one account is allowed only on a specified device by the Windows Media DRM.

In the recent year, the mobile devices provide users the ability of multimedia access through the Internet. As the growth of mobile devices, requests for net surfing on mobile devices are increasing. The technique of using the Windows Media DRM is not suitable for the mobile device system because the Windows system is not popular for use as software systems on mobile devices.

An idea is proposed in this study to restrict users, who have registered on the website, to display clips of movies only on specified mobile devices. In Section 3.2, we will describe our idea of this kind of video copyright protection. And detailed algorithms will be given. In Section 3.3, some experimental results will be shown with discussions made.

3.2. Techniques of Proposed Method

In order to achieve the above-mentioned purpose, two steps are proposed. First, the video content is randomized by a random key. Second, a scheme of transmission of the key is adopted. In Section 3.2.1, the structure of an involved multimedia web

server will be shown. In Section 3.2.2, we will propose an idea of copyright protection of videos by randomization of video contents using the random key. In Section 3.2.3, we will propose another idea of how to put restrictions on users so that they can only download and display videos on specified mobile phones. In Section 3.2.4, a detailed algorithm for this purpose will be shown.

3.2.1. System Configuration

Our multimedia web servers provide registered users the service of viewing movie clips. There need some databases in the server for this type of service. First, there is a user database which records the user account and the IMEI number of the mobile phone of each registered user. Second, there needs a database of provided 3GP movie clips and the keys used to randomize the video contents in the database.

More specifically, a manager of the multimedia website assigns a *random key* to randomize the video contents of the movie clips and records the key into the database. He/She then puts the protected movie clips on the website for registered users to download and view. An illustration of the configuration of the proposed system is shown in Figure 3. 1.

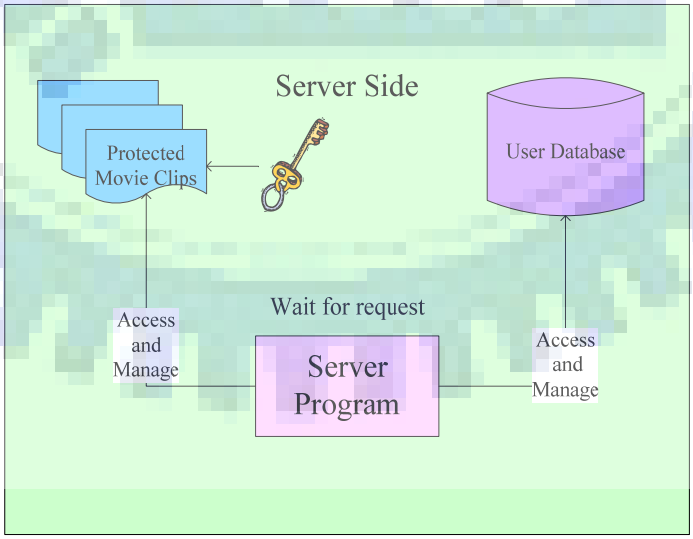


Figure 3. 1 The system configuration of a multimedia website proposed in this study.

Furthermore, a user who wants to be a member of the multimedia website has to download a client program from the multimedia website. The client program will help the user to register to get a user account. Then, the client program sends the user account and the IMEI extracted from the user's mobile phone to the multimedia server. The server records the information into the database. After the registration procedure is done, the user can use the client program to browse the website and enjoy the service of downloading and viewing movies from the multimedia website. A flowchart of the registration procedure is shown in Figure 3.2.

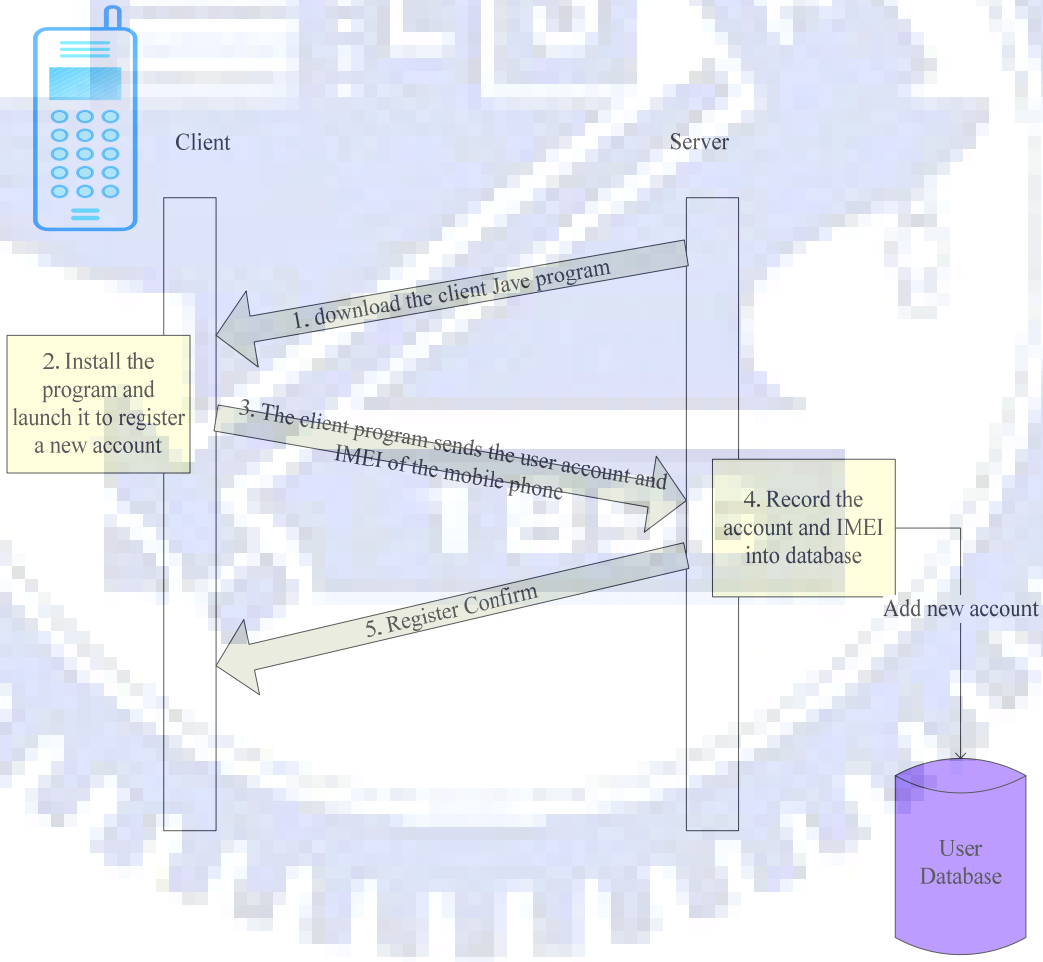


Figure 3.2 The procedure of user registration.

3.2.2. Randomization of Video Contents by Random Keys

In order to prevent the copyright of the movie clips in the multimedia server, we randomize the video contents by user random keys in advance. Without getting a correct key, the server cannot recover the video contents correctly.

After a 3GP video is acquired, the proposed system will extract the H.263 video track from the 3GP file at first. Then the system processes the H.263 video track to randomize the content of the video. Because of the property of the GOB format of the H.263 video, the GOB is taken as a unit of randomization. After using a random number generator to produce a random key of a sufficient length, the produced key is used to randomize the position of the MB in every GOB in the I-frames of the video clip.

After randomization, the randomized H.263 file is finally combined with the audio track into the protected 3GP file. A flowchart of the procedure is shown in Figure 3.3.

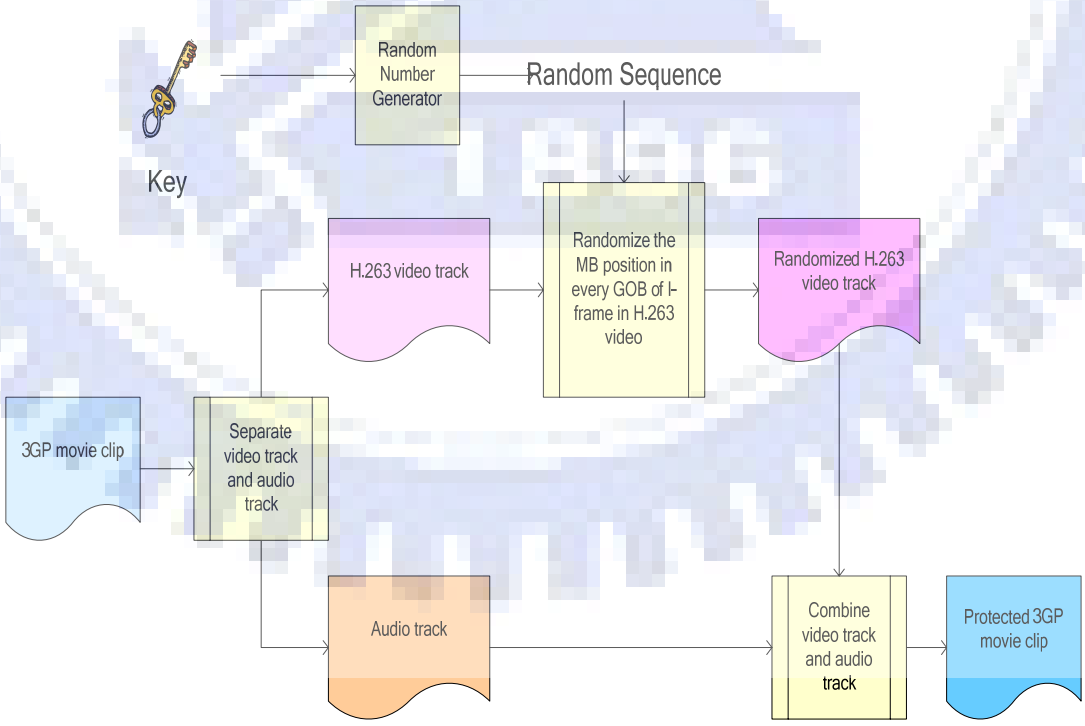


Figure 3.3 A flowchart of proposed copyright protection method.

3.2.3. Copyright Protection Scheme by Use of IMEI and An Encryption Technique via Exclusive-OR Operation

In order to bind the user account to a specified mobile phone, a method is proposed in this study. We use the IMEI of the mobile phone and a symmetric encryption technique to achieve this purpose.

When a user logins a website and requests for the movies supplied by the website, the client program will send the user request to the server. After the server receives the request, it will give a response message. The message is the random key encrypted by the IMEI of the requesting mobile phone. The encryption method will be described in Section 3.2.4.

When the client program receives the response message from the server, the client program will use the same encryption method to solve the random key and then send it back to the multimedia server. After the server receives the response key, the server uses the key to recover the 3GP video file and sends the recovered video to the client for display.

A flowchart of the proposed scheme is shown in Figure 3.4.

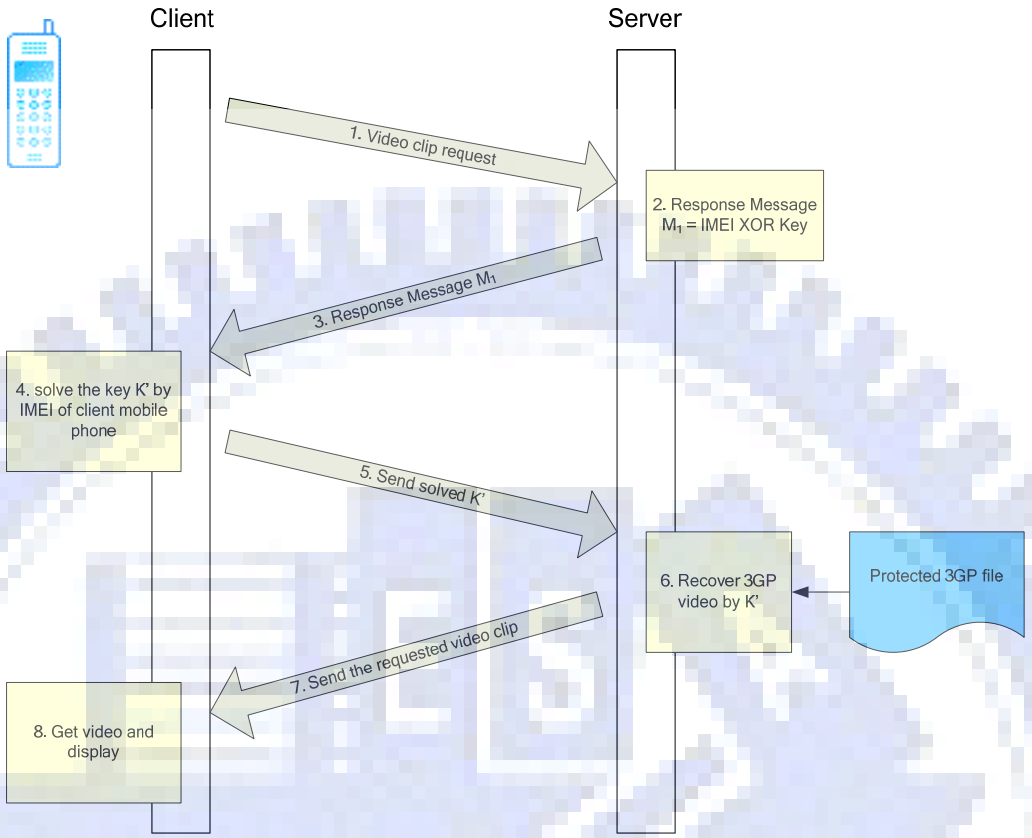


Figure 3.4 A flowchart of proposed scheme.

3.2.4. Detailed Algorithm

After extracting the video track from a 3GP movie clip, a sequence of frames is obtained. An algorithm for randomization of the video content is described as follows.

Algorithm 3.1: Randomization of video contents.

***Input*:** a user key *K* and a sequence of video frames *S*.

***Output*:** a sequence of frames in which the positions of the macroblocks in the I-frames are randomized.

***Steps*:**

1. Use *K* to generate a random byte sequence and separate it into *n* parts:

$$A_1 = (a_{11}, a_{12}, \ldots, a_{1m}), A_2 = (a_{21}, a_{22}, \ldots, a_{2m}), \ldots, A_n = (a_{n1}, a_{n2}, a_{n3}, \ldots, a_{nm})$$

where n is the number of GOBs in every I-frame, m is the number of macroblocks in every GOB, and each a_{ij} is a byte with its value being in the range of 0 through 255.

2. Use A_i to randomize the positions of the macroblocks in the i-th GOB of an I frame in the following way, where $i = 1, 2, \ldots, n$.

   2.1 Take the modulos of every two consecutive elements in A_i as follows:

   $$L_k = (a_{ik})_{\text{mod } m};$$

   $$L_{k+1} = (a_{i(k+1)})_{\text{mod } m}$$

   where $k = 1, 2, \ldots, m-1$.

   2.2 Exchange the L_k-th macroblock with the L_{k+1}-th macroblock in the i-th GOB .

3. Repeat the above steps for every I-frame from S.

In the recovery procedure of a protected video, the random positions of the macroblocks should be found out first. Then, by using of the result, reversions of macroblock positions are conducted to recover the protected video content. The recovery process is described as follows.

Algorithm 3.2: Recovery of the contents of a video.

Input: A sequence S of randomized frames and a key K' got from the client side.

Output: A recovered video.

Steps:

1. Create a matrix T with $n \times m$ elements which records the position of every macroblock as follows:

   $$T = \{T_{ij} \mid i = 1, 2, \ldots, n \text{ and } j = 1, 2, \ldots, m\}$$

   where n is the number of GOBs in every I-frame and m is the number of

macroblocks in every GOB. Every T_{ij} represents the position of a macroblock, in which i means that this macroblock belongs to Number i GOB, and j means that this is the j-th macroblock in the i-th GOB.

2. Initialize the elements in T in the following way:

$$T_{ij} = j.$$

3. Use K' to generate a random byte sequence and separate it into n parts:

$A_1 = (a_{11}, a_{12}, \ldots, a_{1m})$, $A_2 = (a_{21}, a_{22}, \ldots, a_{2m})$, …, $A_n = (a_{n1}, a_{n2}, a_{n3}, \ldots, a_{nm})$

where each a_{ij} is a byte with its value being in the range of 0 through 255.

4. Use A_i to randomize the i-th row of T in the following way, where $i = 1, 2, \ldots, n$.

   i. Take the modulos of every two consecutive elements in A_i as follows:

$$x = (a_{ik})_{\text{mod } m};$$

$$y = (a_{i(k+1)})_{\text{mod } m}$$

   where $k = 1, 2, \ldots, m-1$.

   ii. Exchange $T_{i,x}$ and $T_{i,y}$.

5. Make a new matrix T' with $n \times m$ elements as follows:

$$T' = \{T'_{ij} \mid i = 1, 2, \ldots, n \text{ and } j = 1, 2, \ldots, m\}.$$

6. Set every T'_{ij} as follows:

   i. Search every element in the i-th row of T, find out the T_{ik} whose value is equal to j and set T'_{ij} as k.

7. Search for every I-frame IF in S, and recover the content of IF to be a new I-frame IF' in the following way.

   i. For the j-th macroblock in the i-th GOB in IF', get the data from the T'_{ij}-th macroblock in the i-th GOB in IF.

   ii. Repeat Step i above until all the macroblocks of IF' are processed.

   iii. Replace IF with IF' in S.

The symmetric encryption method used in the key transmission scheme is described as follows.

Algorithm 3.3: Encryption used in key transmission.

Input: A key *K* used to randomize video contents and the IMEI *I* of the requested client.

Output: An encrypted sequence *B* of bytes.

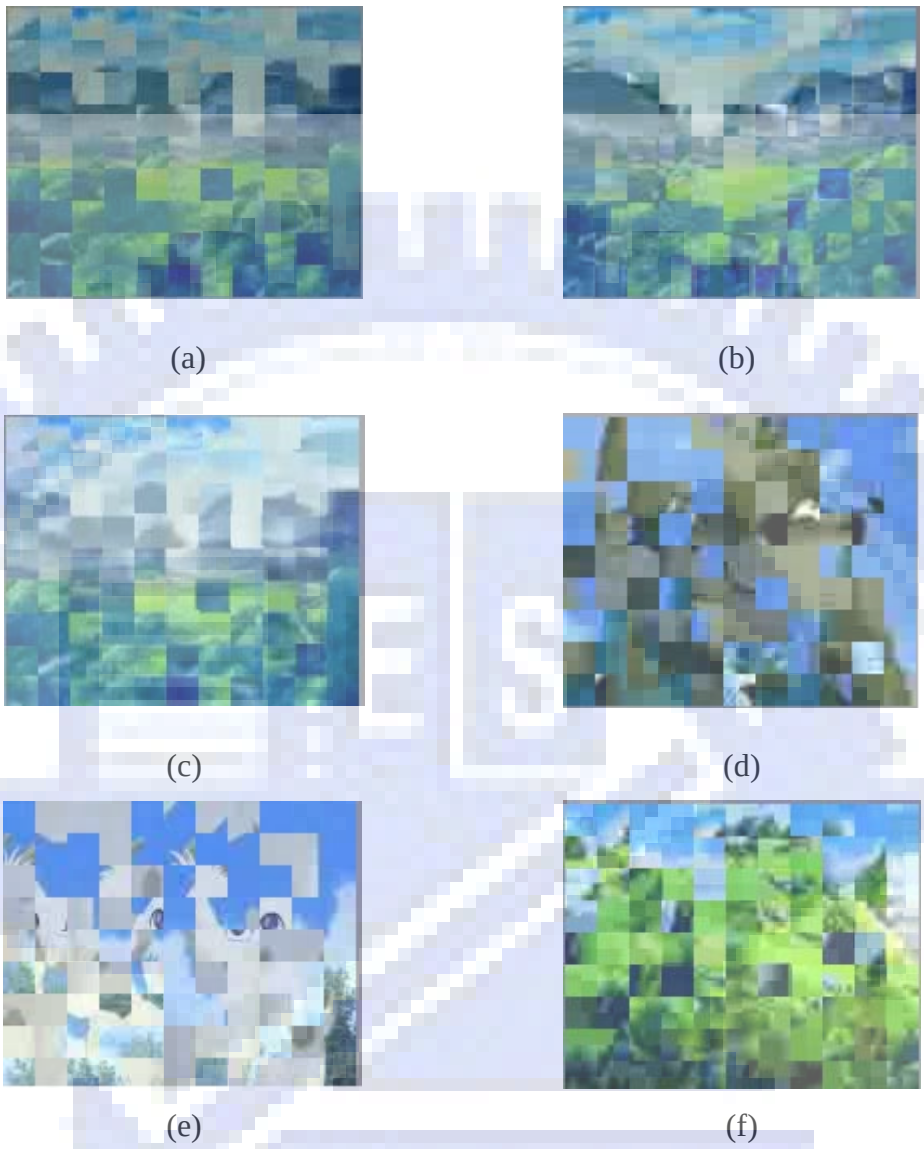
Step:

1. Take *I* as the input of a random number generator to generate a sequence *S* of bytes with the same size as *K*.
2. For every byte in *K*, perform the Exclusive-OR operation on the *i*-th byte of *K* and the *i*-th byte of *S* to generate the *i*-th byte of *B*.
3. Take the generated byte sequence *B* as the one to be transmitted to the client program.

After the client program receives the encrypted sequence, it recovers the random key and then transmits it to the server to get the desired movies.

3.3. Experimental Results and Discussions

In our experiments, the application of copyright protection of videos for displays on a specified mobile phones is presented. First, Figure 3. 5 shows some frames of protected videos by a random key.



(a) (b)

(c) (d)

(e) (f)

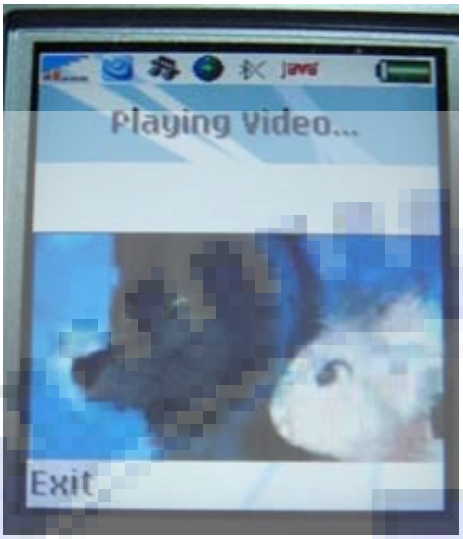
Figure 3. 5 Result of protected video.(a)(b)(c)(d)(e)(f) Some frames of the protected video.

Second, an example of the process of viewing videos on mobile device is shown in Figure 3. 6.

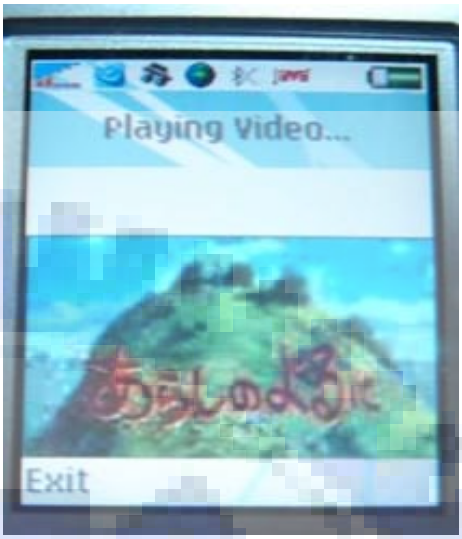
(a) (b)

(c) (d)

Figure 3. 6 (a) The main menu of client program. (b) The login menu of client program. (c) After login, the movie list of this website will be shown and the user can choose the movie to display. (d) (e) (f) Some pictures caught when displaying.



(e)



(f)

Figure 3. 6 (a) The main menu of client program. (b) The login menu of client program. (c) After login, the movie list of this website will be shown and the user can choose the movie to display. (d) (e) (f) Some pictures caught when displaying(continued).

Finally, an example of results that an illegal user uses the same account in an unregistered mobile phone to get a movie for display is shown in Figure 3. 7.

In this chapter, we have proposed a copyright protection scheme of videos for display on mobile devices. The proposed method can protect videos on the Internet from suffering from illegal downloads for display. And the protected videos are safe on the Internet because the contents of the protected videos will not be stolen without the user key.

(a) (b)

(c) (d)

Figure 3. 7 (a) User use the same user account to login the web server. (b) Getting the movie list of web server. (c) (d) The result of displaying the same movie on an illegal mobile device.

Chapter 4
Lossless Visible Watermarking for Copyright Protection of Videos Displayed on Mobile Phones

4.1. Introduction

With the advance of mobile computing technologies, more and more cellular phone provides the ability of video recording. People may use these functions to record something interesting in their daily life.

And because of the improvement of net surfing on mobile devices, users can use mobile devices to access the Internet. Many kinds of multimedia data are shared on the Internet. When users produce a video, they may want to share their productions on the Internet. Because of the limit of the storage of mobile phones, a user may copy a video clip to a personal computer and does not keep it in his/her mobile device.

When the users share their recorded videos on the Internet, they do not want to see their productions suffering from downloading by illegal users. In order to protect the copyright of the videos, a lossless visible watermarking technology is proposed in this study.

In Section 4.2, we will describe the proposed lossless visible watermarking method for copyright protection of video contents. In Section 4.3, some experimental results of the proposed method will be shown. Finally in Section 4.4, some discussions will be made.

4.2. Proposed Lossless Visible Watermarking Method

Our goal is to protect the copyright of 3GP videos by a lossless visible watermarking technology. The video is protected by a user input key. The watermarked videos will not be displayed correctly without the correct key. The watermarking information is embedded in order to help the server to recover the videos without recording the watermark image.

A user uses a program designed in this study to embed a watermark into a recorded video and then uploads the protected video to the server. The server is like a multi-function web server. First, it receives HTTP requests, processes the requests, and then gives appropriate responses. Second, it provides spaces to users to share their videos. Furthermore, it recovers the watermarked video and transmits to the client. A system configuration of the proposed method is shown in Figure 4.1.

In Section 4.2.2, we will describe the proposed method. In Section 4.2.3, the process of embedding a watermark will be described and in Section 4.2.4, the process of embedding watermarking information will be elaborated. In Section 4.2.5, the procedure of recovery of the original video will be described.

4.2.1. Review of Block Layer Encoding of H.263 Format

As mentioned in Section 2.3.1, every intra-coded MB comprises four luminance blocks and two chrominance blocks. The structure of the block layer encoding format

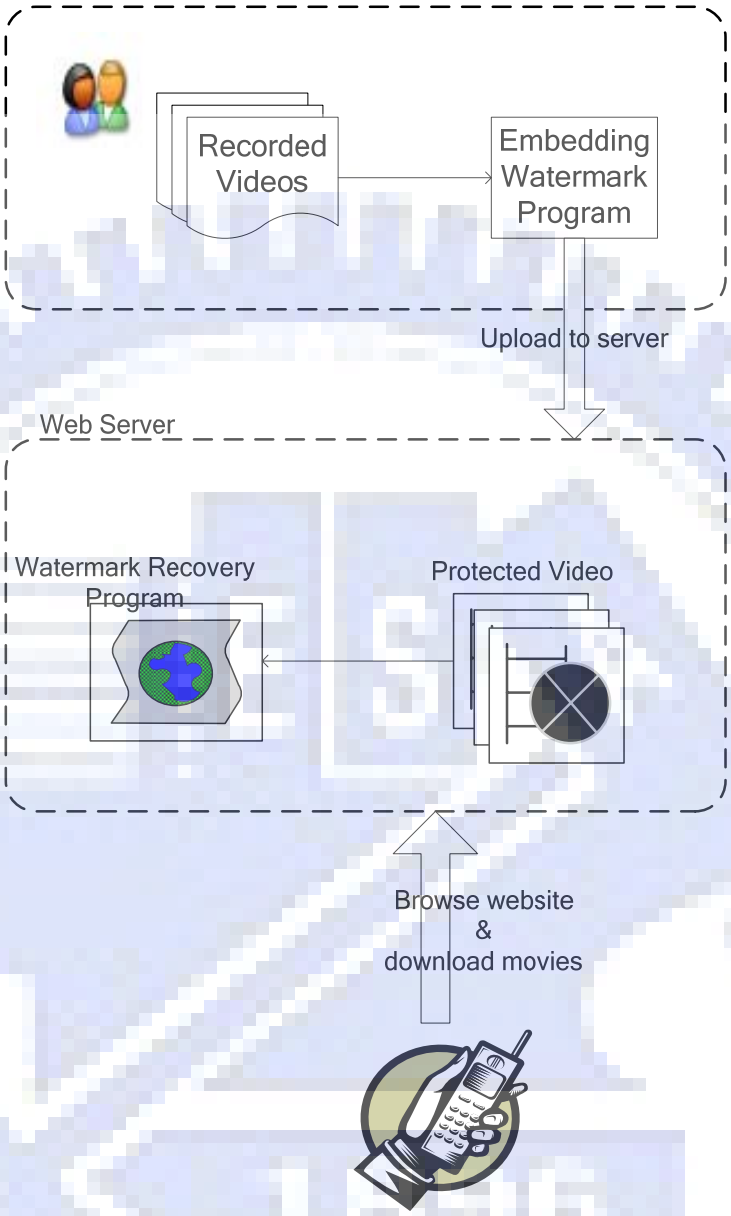


Figure 4.1 A system configuration of the proposed copyright protection method.

is shown in Figure 4.2. The field INTRADC, a codeword of 8 bits, and TCOEF represent the DC and AC of the quantized DCT coefficients for intra-coded blocks, respectively.

INTRADC	TCOEF

Figure 4.2 The structure of block layer.

The TCOEF field represents a combination of a last nonzero coefficient indication (LAST: “0” and “1” mean that there are more or no more nonzero coefficients in this block, respectively) and a Run-level Coding (RLC).

There are two different coding methods of the TCEOF field: variable length coding (VLC) and fixed length coding (FLC). The most commonly occurring RLC is encoded by VLC and the remaining combination of (LAST, RUN, LEVEL) are fixed length coded with a 22-bit codeword consisting of 7 bits ESCAPE, 1 bit LAST, 6 bits RUN and 8 bits LENGTH. The FLC and VLC codes for TCOEF are given in Table 4.1 and Table 4.2.

Table 4.1 The FLC table for RUN and LEVE [9].

Index	Run	Code
0	0	000 000
1	1	000 001
2	2	000 010
.	.	.
.	.	.
63	63	111 111

Index	Level	Code
-	-128	FORBIDDEN
0	-127	1000 0001
.	.	.
125	-2	1111 1110
126	-1	1111 1111
-	0	FORBIDDEN
127	1	0000 0001
128	2	0000 0010
.	.	.
253	127	0111 1111

Table 4.2 VLC table for TCOEF (s means the sign of the level) [9].

INDEX	LAST	RUN	LEVEL	BITS	VLC CODE
0	0	0	1	3	10s
1	0	0	2	5	1111 s
2	0	0	3	7	0101 01s
3	0	0	4	8	0010 111s
4	0	0	5	9	0001 1111 s
5	0	0	6	10	0001 0010 1s
6	0	0	7	10	0001 0010 0s
7	0	0	8	11	0000 1000 01s
8	0	0	9	11	0000 1000 00s
9	0	0	10	12	0000 0000 111s
10	0	0	11	12	0000 0000 110s
11	0	0	12	12	0000 0100 000s
12	0	1	1	4	110s
13	0	1	2	7	0101 00s
14	0	1	3	9	0001 1110 s
15	0	1	4	11	0000 0011 11s
16	0	1	5	12	0000 0100 001s
17	0	1	6	13	0000 0101 0000s
18	0	2	1	5	1110 s
19	0	2	2	9	0001 1101 s
20	0	2	3	11	0000 0011 10s
21	0	2	4	13	0000 0101 0001s
22	0	3	1	6	0110 1s
23	0	3	2	10	0001 0001 1s
24	0	3	3	11	0000 0011 01s
25	0	4	1	6	0110 0s
26	0	4	2	10	0001 0001 0s
27	0	4	3	13	0000 0101 0010s
28	0	5	1	6	0101 1s
29	0	5	2	11	0000 0011 00s
30	0	5	3	13	0000 0101 0011s
31	0	6	1	7	0100 11s
32	0	6	2	11	0000 0010 11s
33	0	6	3	13	0000 0101 0100s
34	0	7	1	7	0100 10s
35	0	7	2	11	0000 0010 10s
36	0	8	1	7	0100 01s
37	0	8	2	11	0000 0010 01s
38	0	9	1	7	0100 00s
39	0	9	2	11	0000 0010 00s
40	0	10	1	8	0010 110s
41	0	10	2	13	0000 0101 0101s
42	0	11	1	8	0010 101s
43	0	12	1	8	0010 100s
44	0	13	1	9	0001 1100 s
45	0	14	1	9	0001 1011 s
46	0	15	1	10	0001 0000 1s
47	0	16	1	10	0001 0000 0s
48	0	17	1	10	0000 1111 1s
49	0	18	1	10	0000 1111 0s
50	0	19	1	10	0000 1110 1s
51	0	20	1	10	0000 1110 0s
52	0	21	1	10	0000 1101 1s
53	0	22	1	10	0000 1101 0s
54	0	23	1	12	0000 0100 010s
55	0	24	1	12	0000 0100 011s
56	0	25	1	13	0000 0101 0110s
57	0	26	1	13	0000 0101 0111s

INDEX	LAST	RUN	LEVEL	BITS	VLC CODE
58	1	0	1	5	0111 s
59	1	0	2	10	0000 1100 1s
60	1	0	3	12	0000 0000 101s
61	1	1	1	7	0011 11s
62	1	1	2	12	0000 0000 100s
63	1	2	1	7	0011 10s
64	1	3	1	7	0011 01s
65	1	4	1	7	0011 00s
66	1	5	1	8	0010 011s
67	1	6	1	8	0010 010s
68	1	7	1	8	0010 001s
69	1	8	1	8	0010 000s
70	1	9	1	9	0001 1010 s
71	1	10	1	9	0001 1001 s
72	1	11	1	9	0001 1000 s
73	1	12	1	9	0001 0111 s
74	1	13	1	9	0001 0110 s
75	1	14	1	9	0001 0101 s
76	1	15	1	9	0001 0100 s
77	1	16	1	9	0001 0011 s
78	1	17	1	10	0000 1100 0s
79	1	18	1	10	0000 1011 1s
80	1	19	1	10	0000 1011 0s
81	1	20	1	10	0000 1010 1s
82	1	21	1	10	0000 1010 0s
83	1	22	1	10	0000 1001 1s
84	1	23	1	10	0000 1001 0s
85	1	24	1	10	0000 1000 1s
86	1	25	1	11	0000 0001 11s
87	1	26	1	11	0000 0001 10s
88	1	27	1	11	0000 0001 01s
89	1	28	1	11	0000 0001 00s
90	1	29	1	12	0000 0100 100s
91	1	30	1	12	0000 0100 101s
92	1	31	1	12	0000 0100 110s
93	1	32	1	12	0000 0100 111s
94	1	33	1	13	0000 0101 1000s
95	1	34	1	13	0000 0101 1001s
96	1	35	1	13	0000 0101 1010s
97	1	36	1	13	0000 0101 1011s
98	1	37	1	13	0000 0101 1100s
99	1	38	1	13	0000 0101 1101s
100	1	39	1	13	0000 0101 1110s
101	1	40	1	13	0000 0101 1111s
102	ESCAPE			7	0000 011

4.2.2. Idea of Proposed Method

A watermark to be embedded is assumed to be a binary image (a black and white logo, for instance). Each pixel in the watermark will be referred to as a black watermark pixel or a white one according to its binary value 1 or 0, respectively. The idea of the proposed embedding process for I-frames is to use each macroblock to embed a watermark pixel. The idea is illustrated in Figure 4.3.

The embedded watermark pixels can be losslessly removed by checking the embedded watermarking information in the recovery process. The method of embedding a watermark depends on the random signals generated by a user input key. Without getting the correct key, the video will not be recovered losslessly and correctly.

Because the P, B and P-B frames are coded by referencing to the forward or backward I-frames, the watermarked I-frames will influence the appearance of what they display. The change of the I-frames is sufficient to cause damages to the video content.

The method of embedding watermarking information is to insert a token which fits in with the H.263 encoding but should not be shown in a normal encoding procedure into the corresponding macroblock of the first I-frame. In the recovery procedure, the server extracts the watermarking information from the first I-frame and then recovers the video content by the received key.

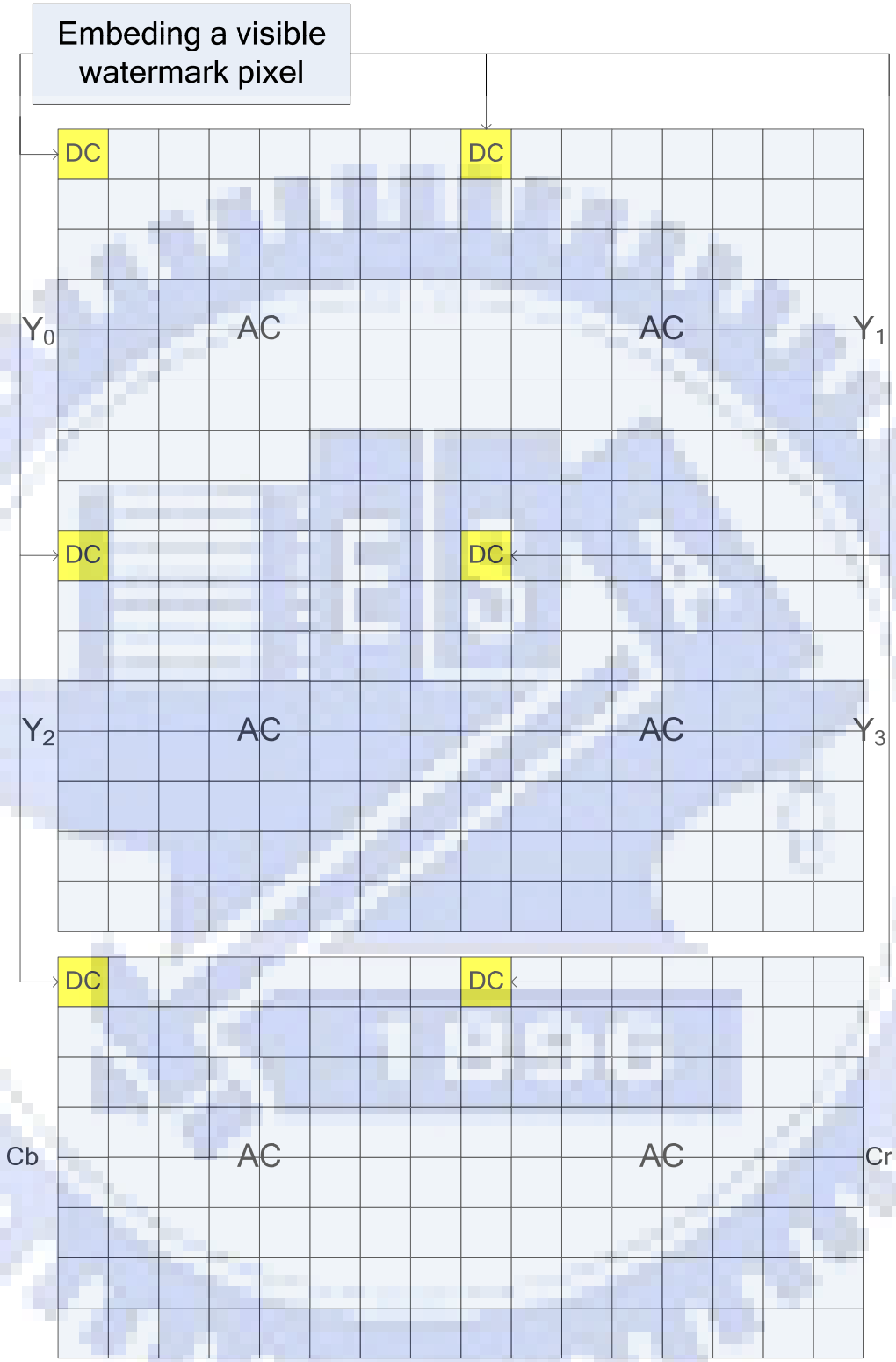


Figure 4.3 The idea of embedding a watermarking pixel.

The procedure of the proposed copyright protection method is shown in Figure 4.4. The detailed process is described in Section 4.2.3 and Section 4.2.3.

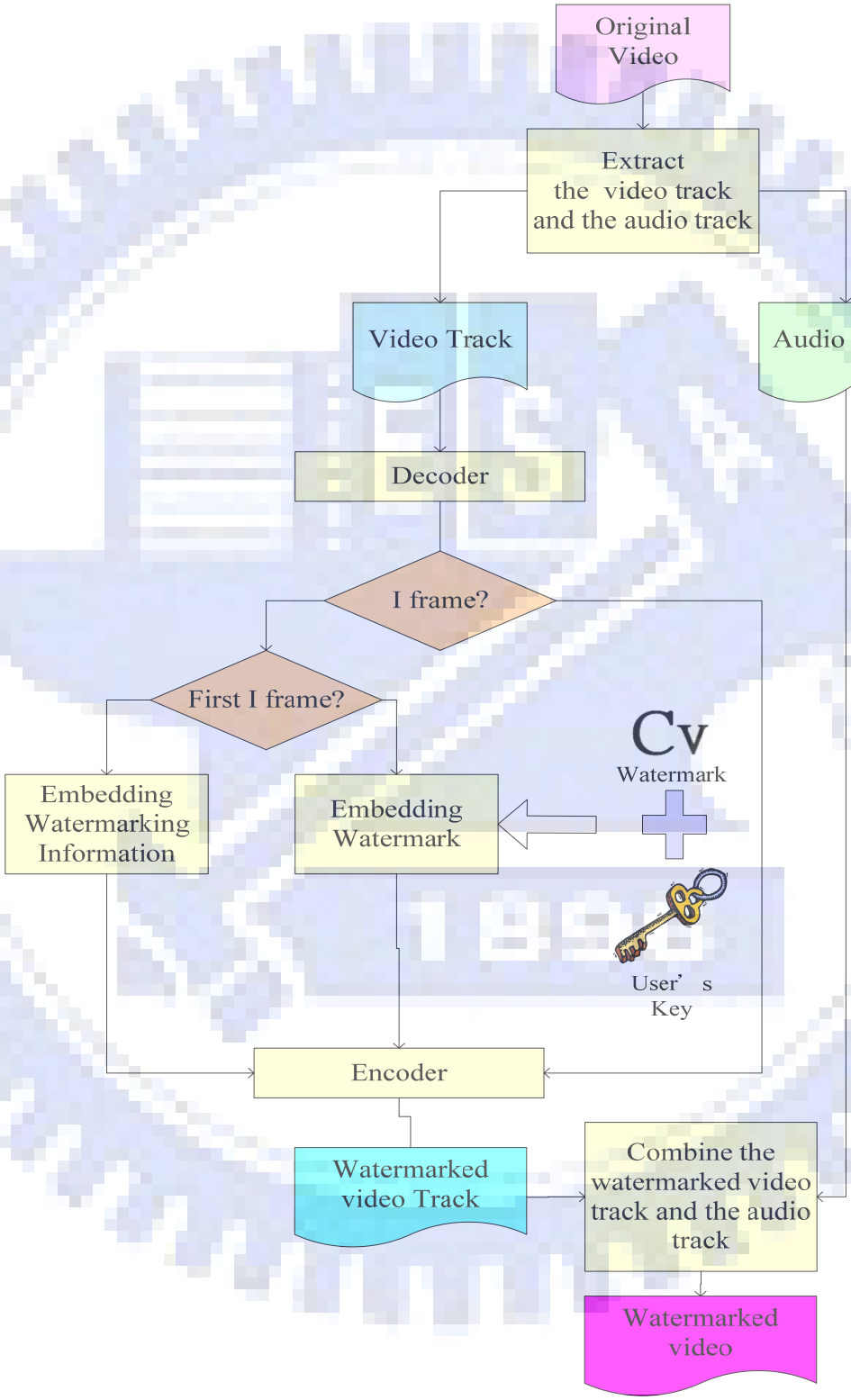


Figure 4.4 The process of proposed method of watermarking.

4.2.3. Watermark Image Embedding Process

In the process of embedding of watermark image, the macroblock of each I-frame are taken as a unit. Each macroblock includes six blocks of 8×8 pixels, four luminance blocks, and two chrominance blocks. In order to retain the black watermark pixels in a macroblock, a modification is made in the DC coefficient in every block in the macroblock. If the embedded watermark pixel is a white pixel, no change will be done in the DC coefficient in the macroblock. A flowchart of the watermark image embedding process is shown in Figure 4.5 and a corresponding detailed algorithm is described in the following.

Algorithm 4.1: Watermark image embedding process

Input: An I-frame F expect the first one, a user key K, and a binary watermark W.

Output: A watermarked I-frame F' with a visible watermark.

Steps:

1. Use the user input key K as a seed for a random number generator to produce two random sequence S and S'. S is a sequence of $m \times n$ elements with its value being in the range of 0 through 255 and S' is a sequence of $m \times n$ elements which are either 0 or 1, where m is the number of GOBs in every frame and n is the number of macroblocks in every GOB. Denote S and S' as follows:

   $$S = \{S_{ij} \mid i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n\}$$

   $$S' = \{S'_{ij} \mid i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n\}.$$

2. For each macroblock in the input I-frame, the modification of the DC coefficient

C_{dc} will be made to every block in this macroblock as follows:

$$\begin{cases} \text{set } C_{dc} = C_{dc} \text{ XOR } S'_{ij}, \text{ if } W_{ij}=1 \text{ and } S_{ij}=1; \\ \text{set } C_{dc} = C_{dc} \text{ NXOR } S'_{ij}, \text{ if } W_{ij}=1 \text{ and } S_{ij}=0; \\ \text{keep } C_{dc} \text{ unchanged, otherwise.} \end{cases}$$

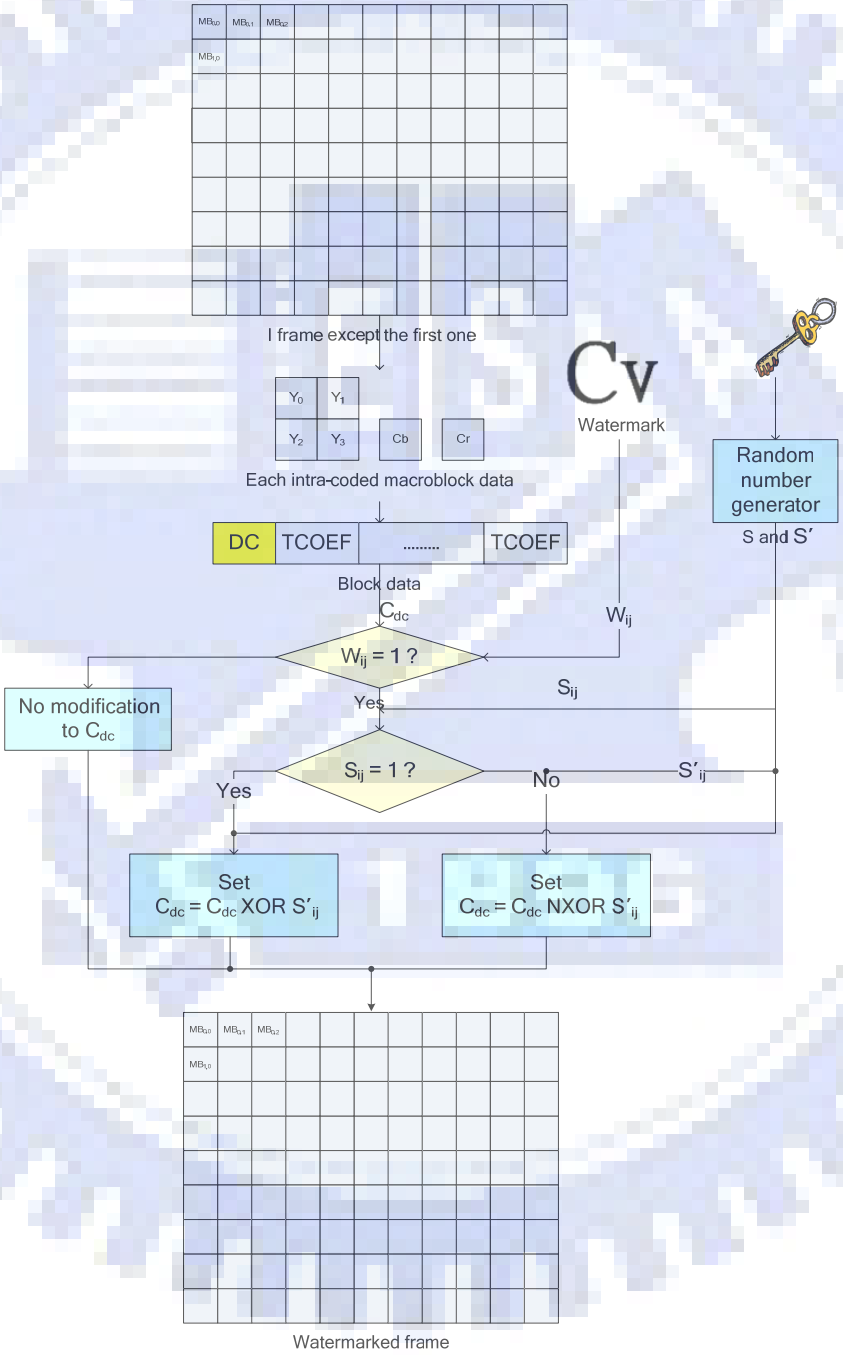


Figure 4.5 Flowchart of the watermark image embedding process.

4.2.4. Watermarking Information Embedding Process

In the proposed method, we use the first I-frame of the videos to record the watermarking information for lossless recovery. Hence, the watermarked videos can be recovered without the watermark image. We use a recording token which is fit in with the H.263 format but will not be used in normal encoding to record a black watermark pixel.

As mentioned in Section 4.2.1, some codes of LEVEL in FLC are not used, such as $(00000000)_2$ and $(10000000)_2$. Therefore, we use the token which is a combination of the ESCAPE code, the LAST with its code being $(1)_2$, the RUN with its code being $(000000)_2$, and the LEVEL with its code being $(00000000)_2$. The structure of the recording token is shown in Figure 4.6 and a flowchart of the embedding process is shown in Figure 4.7. A detailed algorithm is described as follows.

Algorithm 4.2: Process of watermarking information embedding.

Input: The first I-frame F of a video, the watermark image W, and the recording token T.

Output: A recorded I-frame F'.

Steps:

1. For every macroblock in F, take the first luminance block as the recording block.
2. Find the last coefficient code C in the block and modify the block data as follows:

$$\begin{cases} \text{change } C \text{ into the non-last code and insert the recording token } T, \text{ if the } W_{ij} = 1; \\ \text{no change to } C, \text{ otherwise.} \end{cases}$$

	ESCAPE	LAST	RUN	LEVEL
Binary format	0000 011	1	0000 00	0000 0000

Figure 4.6 Structure of the recording token.

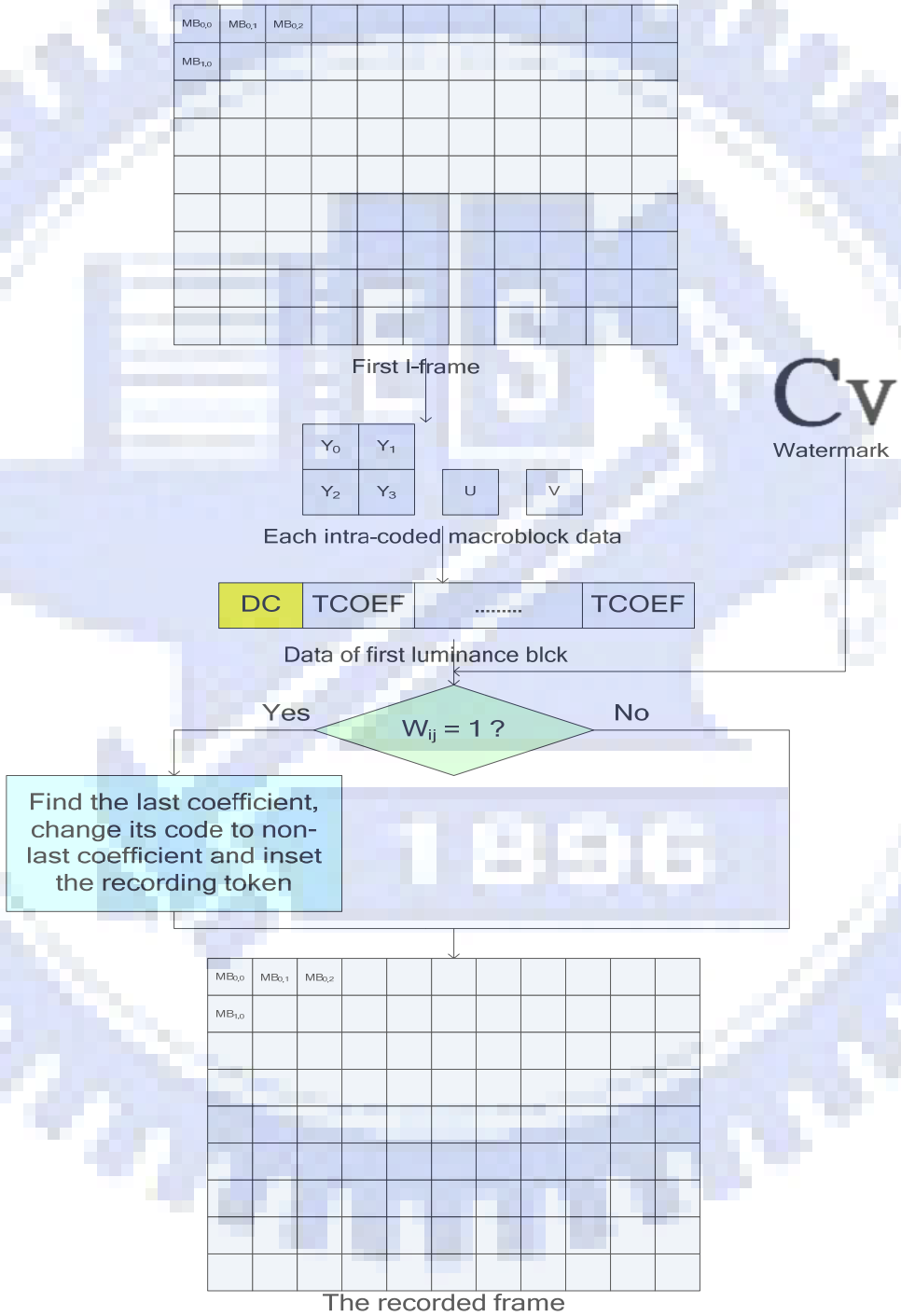


Figure 4.7 Flowchart of watermarking information embedding process.

4.2.5. Recovery of Watermarked 3GP Video by Removing Visible Watermarks

When the server receives requests for the watermarked video, it uses the user input key to recover the watermarked video and then delivers it to the client. If the input key is wrong, the watermark can not be removed losslessly, thus achieving the purpose of copyright protection. The recovery process is shown in Figure 4.8 and a detailed algorithm is described as follows.

Algorithm 4.3: The process of recovery of watermarked videos

Input: A user input key K and a watermarked video V.

***Output*:** A recovered video V'.

***Steps*:**

1. Use the user input key K as a seed for a random number generator to produce two random sequence S and S' different types, where S is a sequence of $m \times n$ elements with its value being in the range of 0 through 255 and S' is a sequence of $m \times n$ elements which are either 0 or 1, where m is the number of GOBs in every frame and n is the number of macroblocks in every GOB. Denote S and S' as follows:

   $S = \{S_{ij} \mid i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n\}$;

   $S' = \{S'_{ij} \mid i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n\}$.

2. Extract the video track T_v and audio track T_a from V.

3. For the first I-frame in T_v, search and remove the recording token and record the watermarking information in W which is an $m \times n$ matrix. If there is a recording

token in the corresponding macroblocks, set W_{ij} as 1; else, as 0.

4. For every macroblock M_{ij} of the following I-frames in T_v, modify the DC coefficient C_{dc} of the blocks as follows:

$$\begin{cases} \text{set } C_{dc} = C_{dc} \text{ XOR } S'_{ij}, \text{ if } W_{ij}=1 \text{ and } S_{ij}=1; \\ \text{set } C_{dc} = C_{dc} \text{ NXOR } S'_{ij}, \text{ if } W_{ij}=1 \text{ and } S_{ij}=0; \\ \text{keep } C_{dc} \text{ unchanged, otherwise.} \end{cases}$$

5. After every I-frame is processed, the recovered video track T_v' is produced. Combine T_v' and T_a into a 3GP video V'.

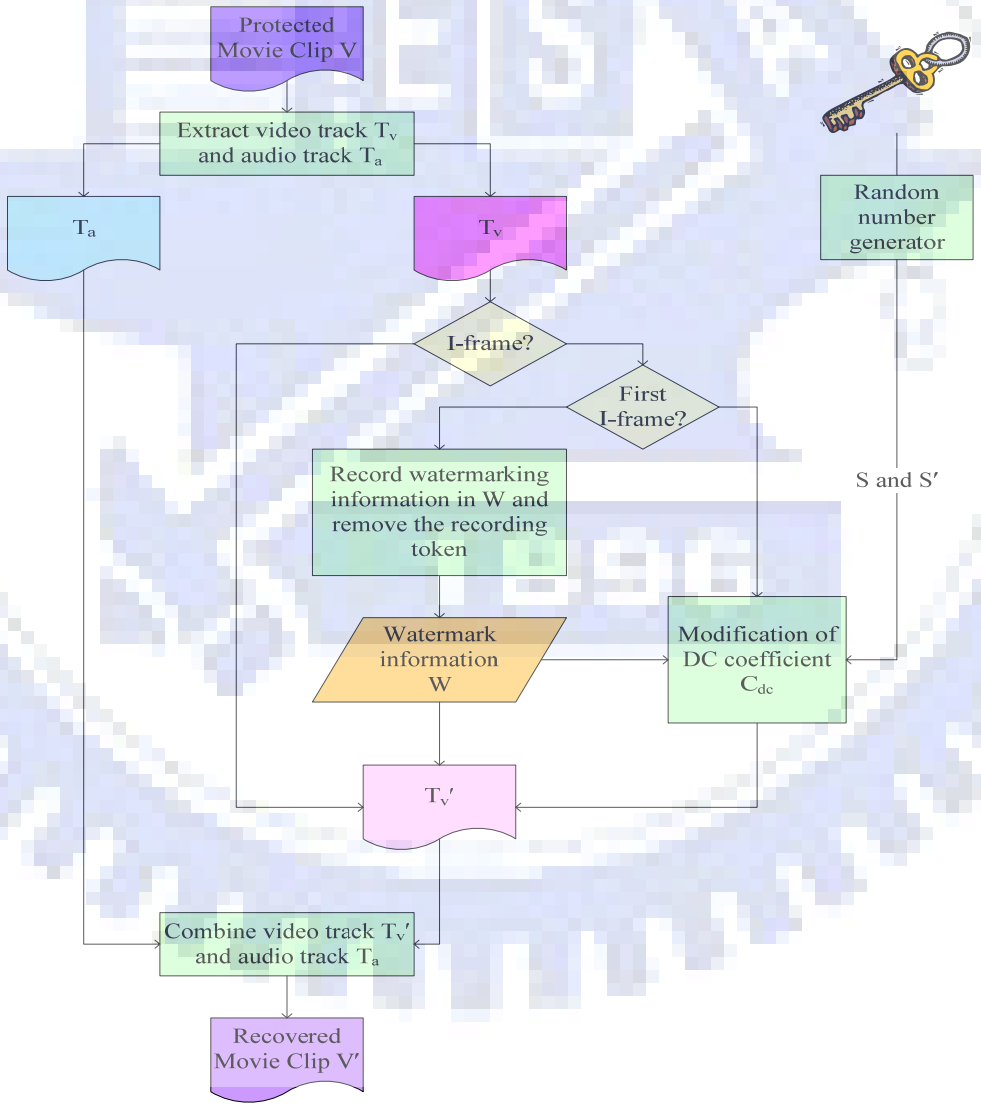


Figure 4.8 Flowchart of recovery process.

4.3. Experimental Results

In our experiments, a watermarked video was put on a public web page. When a user browses this web page, the key is needed to recover the watermarked video and then the recovered video can be download. A watermarked video is shown in Figure 4.9 and the procedure of downloading the shared videos from the web server is shown in Figure 4.10.

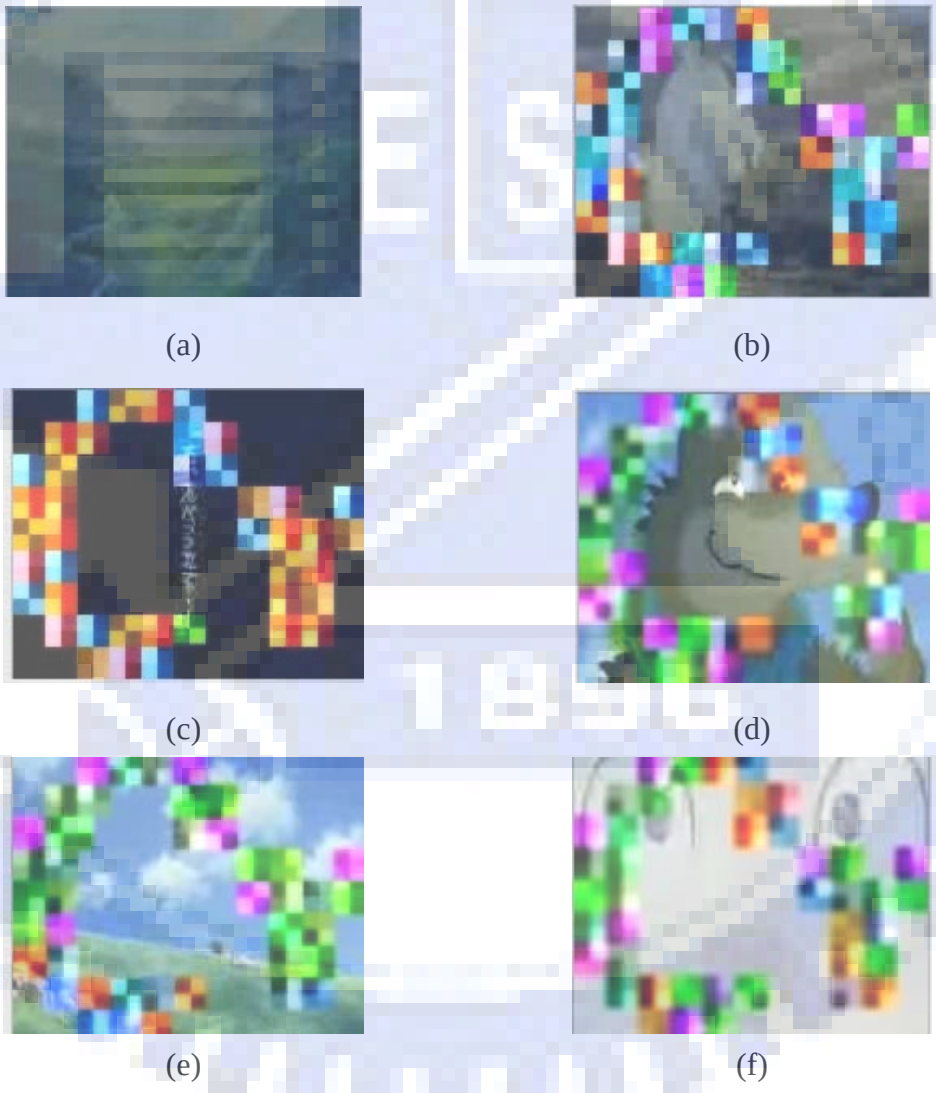
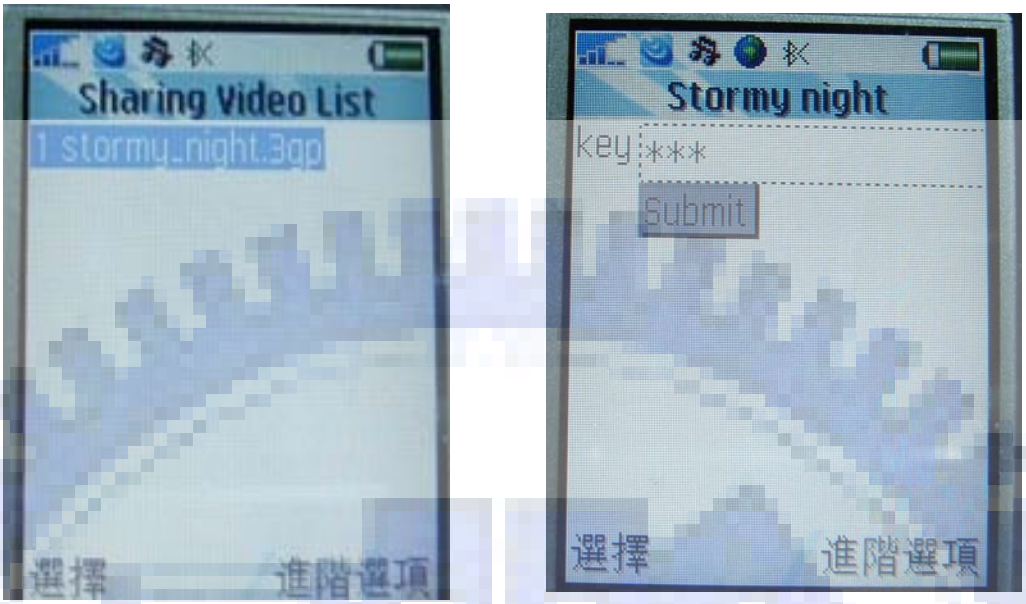


Figure 4.9 Some frames extracted from the watermarked videos. (a) The first I-frame with the watermark information embedded. (b) (c) (d) (e) (f) Some frames after watermarking with user input key.

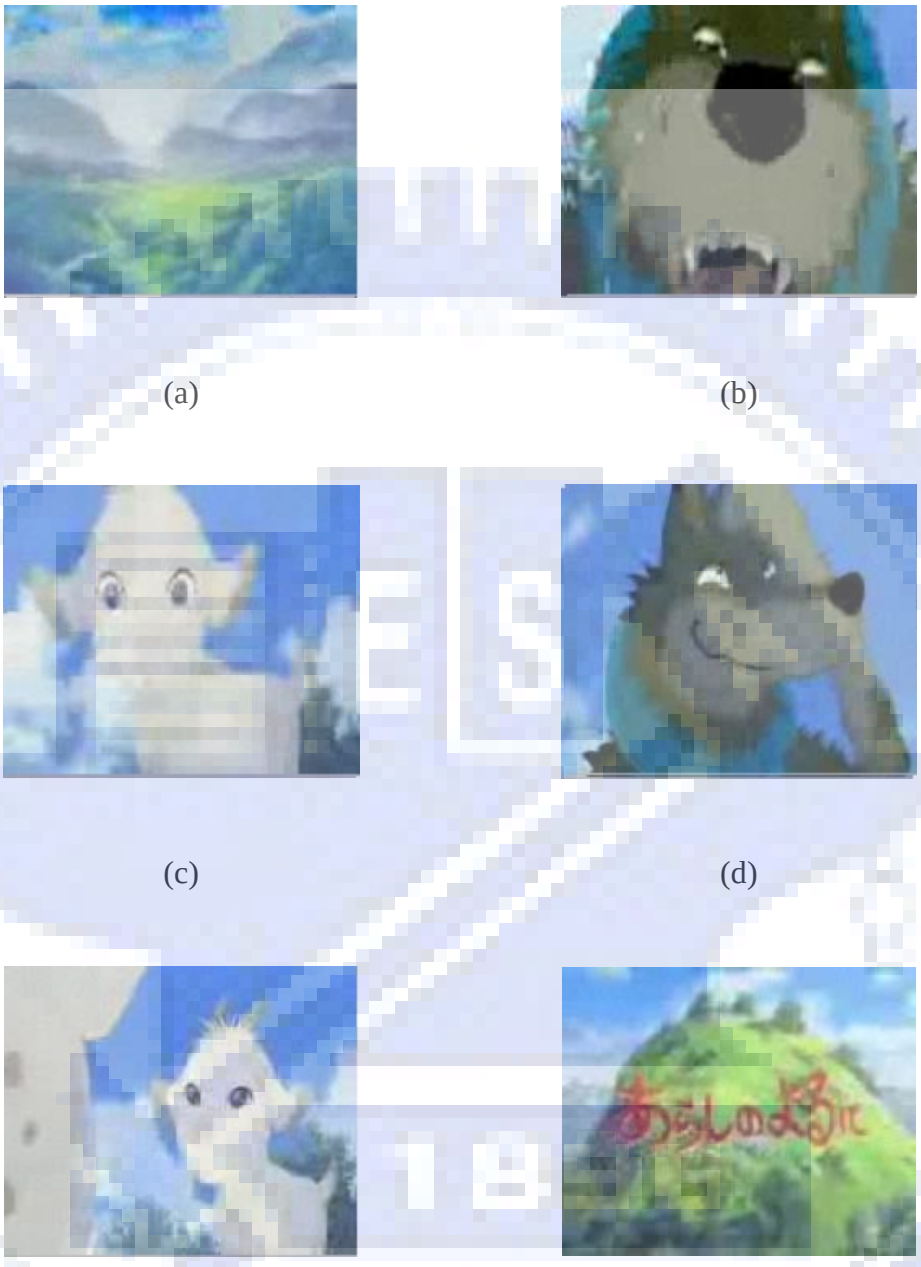


(a) (b)

(c) (d)

Figure 4.10 Procedure of downloading the shared videos. (a) Browse the movie list of the shared watermarked videos by browser. (b) After choosing the movies, the key is requested. (c) The downloading process. (d) Display of the downloading video.

Some frames from the recovered video with correct and incorrect keys are shown in Figure 4.11 and Figure 4.12, respectively.



(a) (b)

(c) (d)

(e) (f)

Figure 4.11 Illustration of the recovered video with the correct key. (a) (b) (c) (d) (e) (f) Some frames extracted from the recovered video with the correct key.

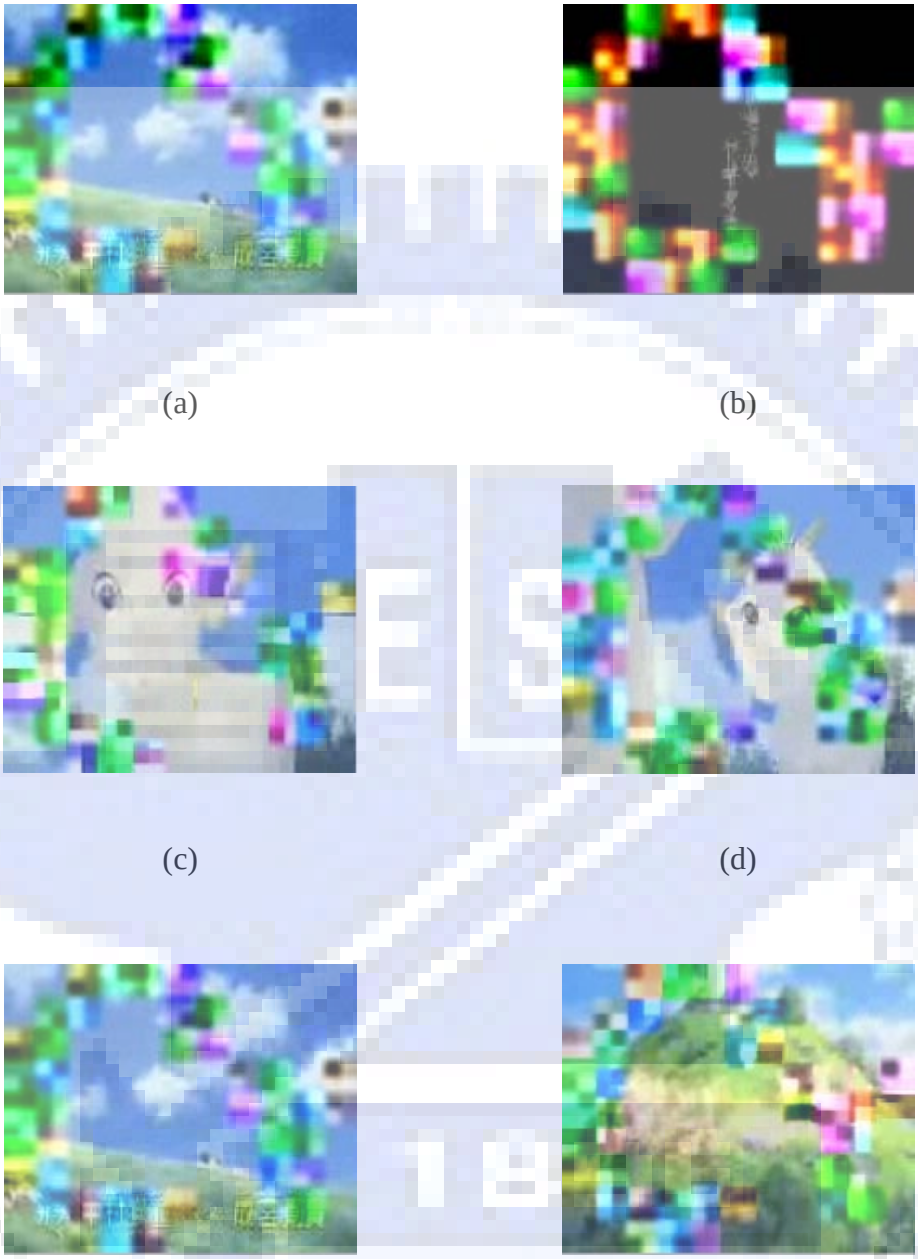


Figure 4.12 Illustration of the recovered video with an incorrect key. (a) (b) (c) (d) (e) (f) Some frames extracted from the recovered video with the incorrect key.

4.4. Discussions

In this chapter, a lossless watermarking technique for videos has been proposed. Users can share their videos on the Internet with more security. Even though the web server is cracked by hackers, the watermarked videos are still defended by the proposed watermark technique for copyright protection. Furthermore, the watermarked videos can be shared to the people who know the protected key of the videos. Without the correct key, the watermarked videos cannot be losslessly recovered. The complexity of recovery of the watermarked videos with the correct key is up to $(256)^n$, where n is the number of macroblocks whose DC values are changed by the watermark embedding process.

Chapter 5
Covert Communication by Videos on Mobile Phones

5.1. Introduction

Due to the advance of mobile computing technologies and the high capacity network, more and more applications on mobile phones are developed. Because of the popularity of using mobile phones, the application of data hiding on mobile phones is useful.

Since video transmission on the Internet is common, a method of covert communication by videos is proposed in this chapter. In Section 5.2.1, the proposed method is described and in Section 5.2.2, a detailed algorithm is presented. In Section 5.3, some experimental results will be shown and in Section 5.4, some discussions will be made.

5.2. Proposed Data Hiding Method for Covert Communication

The idea of the proposed application of covert communication is achieved by transmission of covert videos, in which a secret message is embedded. This application on mobile phones is composed of two parts: the web server in which there are some covert videos in which secret messages are embedded, as well as the client application on a mobile phone which may be used to download covert videos and extract the secret message. A system configuration is shown in Figure 5.1.

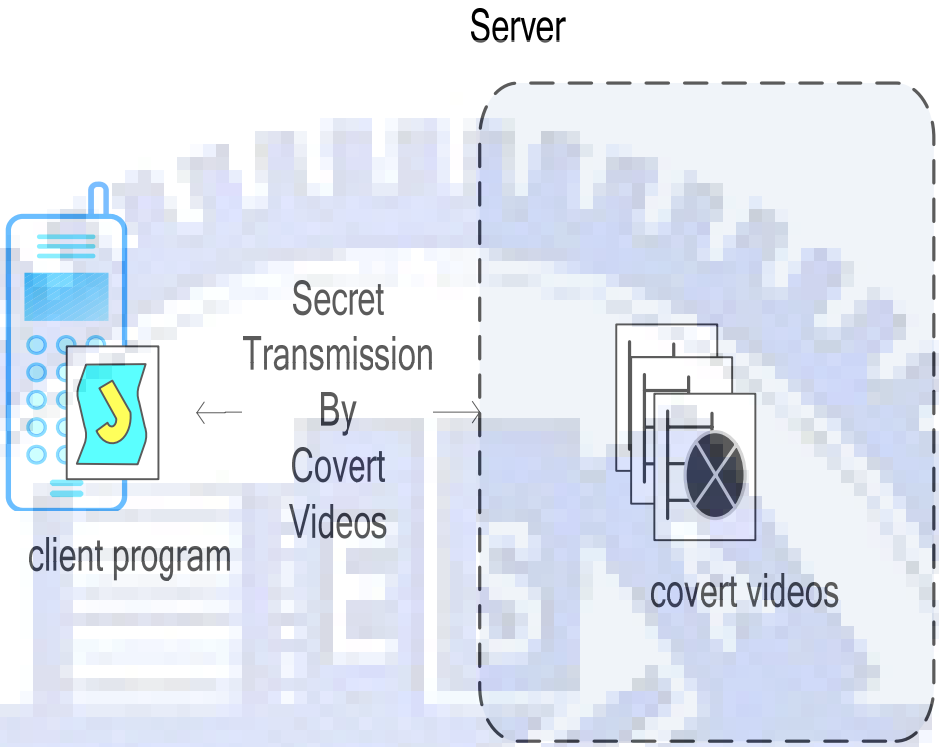


Figure 5.1 System configuration of proposed covert communication method.

5.2.1. Proposed Idea Using AC Coefficients of DCT

As mentioned in Section 4.2.1, there are two kinds of encoding formats of the AC coefficients of the DCT in the block layer in the H.263 format: the VLC encoding and the FLC encoding. There are 127 codes encoded by the VLC and the others are encoded by the FLC.

The proposed method of embedding secret messages in videos is achieved by changing the VLC code to the FLC code. For every intra-coded macroblock, if there are VLC codes, we can use them to embed the secret message. A flowchart of the secret embedding process is shown in Figure 5.2.

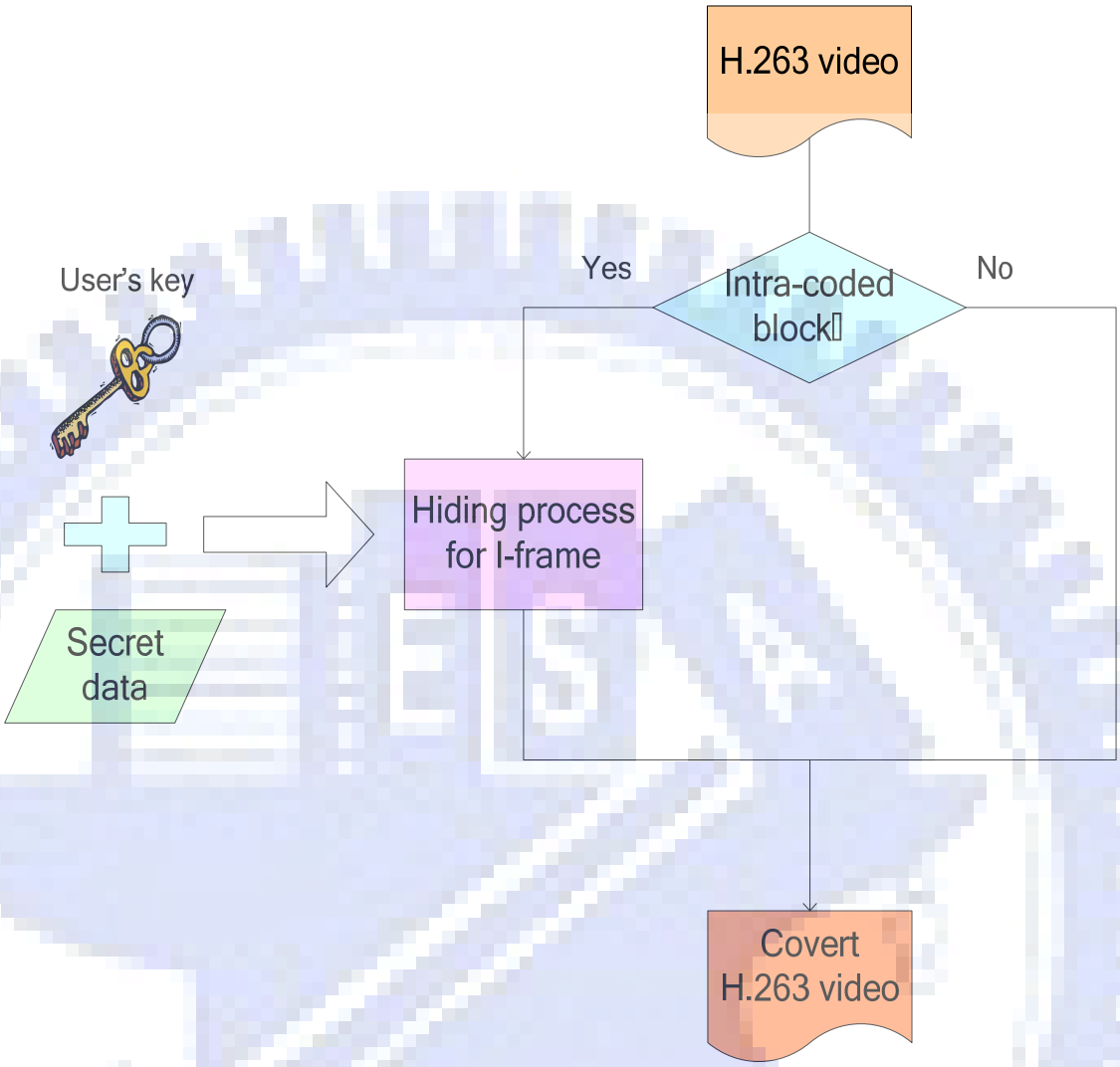


Figure 5.2 Flowchart of the secret embedding process.

5.2.2. Detailed Algorithm

Because I-frames are coded without referencing to other frames, all macroblocks in an I-frame are intra-coded. Every intra-coded macroblock is composed of six blocks. The coding fashion is similar to the compression technique of the JPEG standard. A DCT-based method is applied.

Every block can be used to hide secret messages by making a slight modification of the AC coefficients of the DCT. A detailed algorithm is described as follows and

the process is shown in Figure 5.3.

Algorithm 5.1: Hiding process for videos.

Input: A video V, a secret message D, and a user's key K.

Output: A stego-video V'.

Steps:

1. Take K as the input of a random number generator to generate a sequence S of bytes with the same size as D.
2. For every byte in D, perform the Exclusive-OR operation on the i-th byte of D and the i-th byte of S to generate the i-th byte of D', where D' is a sequence of bytes with the same length as D.
3. Set S as the binary format of D' with N bits. Denote S as follows:

   $$S = \{S_j \mid j = 1, 2, \ldots, N\}.$$

4. For every I-frame in V, read the data of every block and search the VLC code token. If the encoding bit S_k is 1, change the VLC token to the corresponding FLC code; else, do nothing. That is, perform the following operation:

   $$\begin{cases} \text{Change the VLC token into FLC token, if } S_k \text{ is 1;} \\ \text{Make no modification, otherwise.} \end{cases}$$

5. Repeat Step 4 until all bits in S are encoded completely.

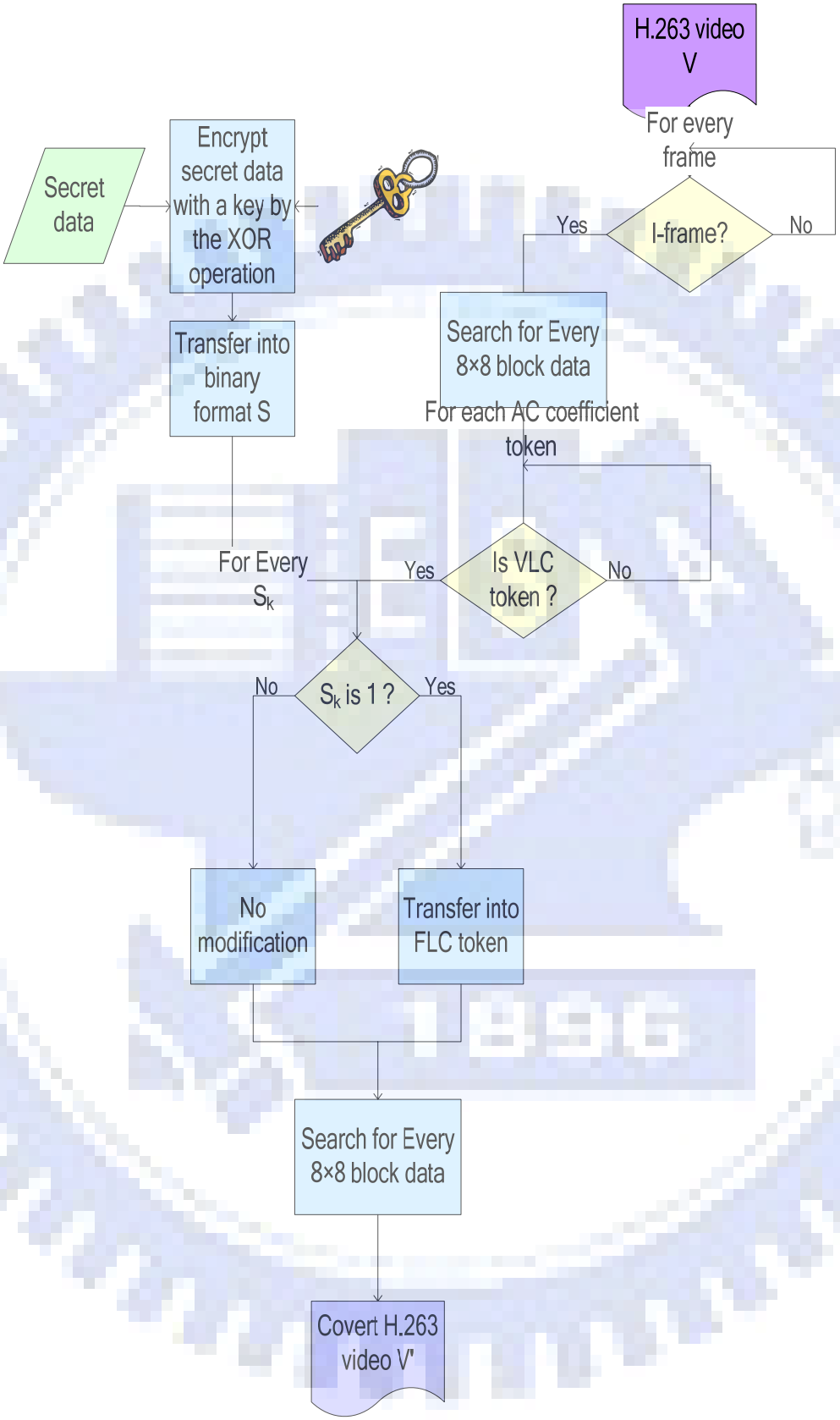


Figure 5.3 The flowchart of embedding process

When a user uses the client program on a mobile phone, he/she inputs the address of the covert video to download it and then inputs a key to recover the secret message. A detailed algorithm is described in the following and the process is shown in Figure 5.4.

Algorithm 5.2: Extracting of a secret message.

Input: The covert video *V* and a user key *K*.

Output: A secret message *S*.

Steps:

1. Search for every AC coefficient token *T* in the intra-coded blocks of *V*.
2. Check the encoding type of *T*. If the token is an FLC token, go to Step 3; else, go to Step 4.
3. Check if the coded pair of (LAST, RUN, LEVEL) were encoded in the VLC code originally. If so, take the hidden bit to be 1; else, ignore the FLC token. Go to Step 5.
4. If the encoding type of *T* is a VLC token, take the hidden bit to be 0.
5. Transform the binary format secret into byte data S_{pre}.
6. Take *K* as the input to a random number generator to generate a sequence *B* of bytes with the same size as S_{pre}.
7. For every byte in S_{pre}, perform the exclusive-OR operation on the *i*-th byte of S_{pre} and the *i*-th byte of B to generate the *i*-th byte of *S*.

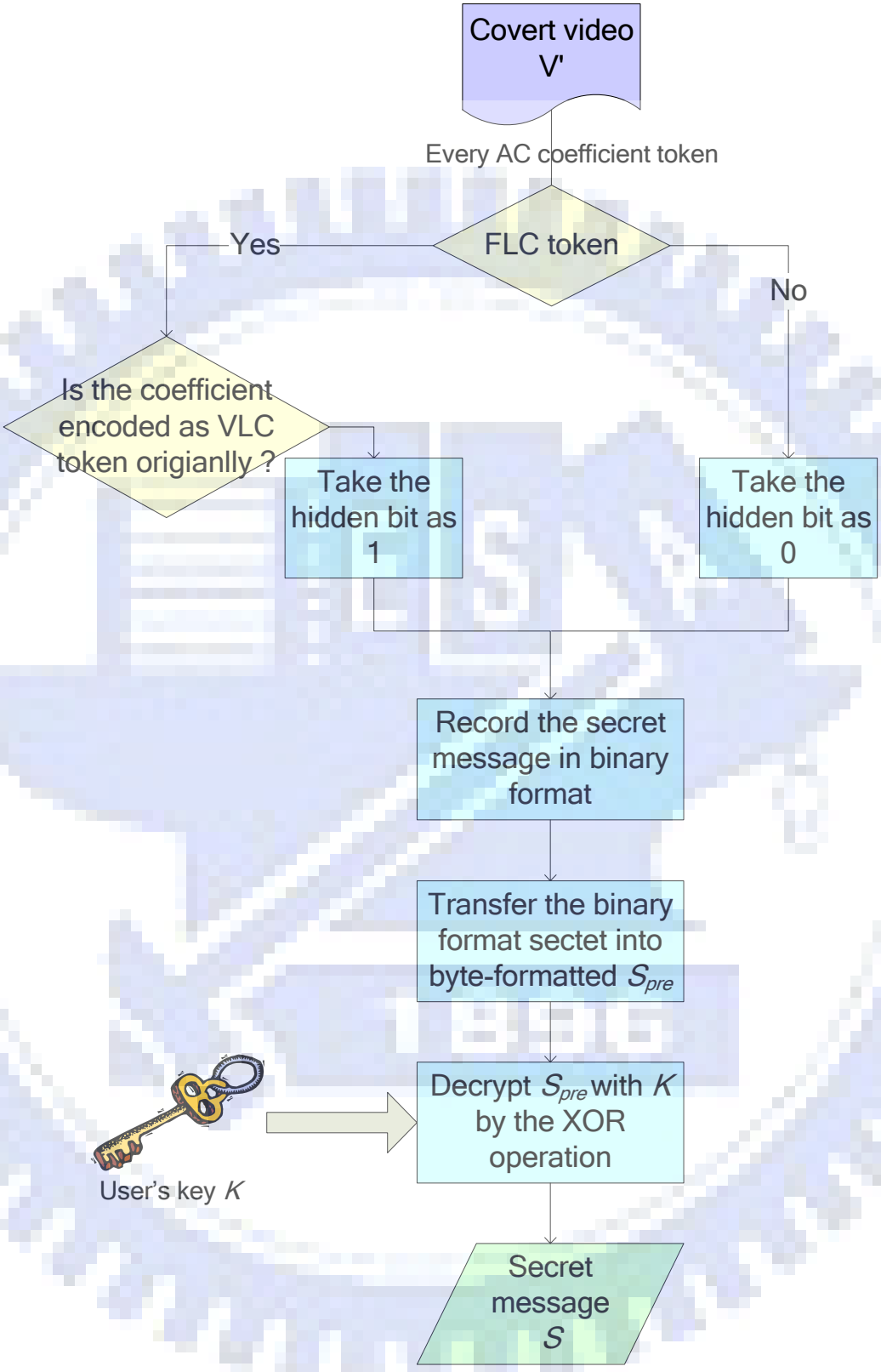
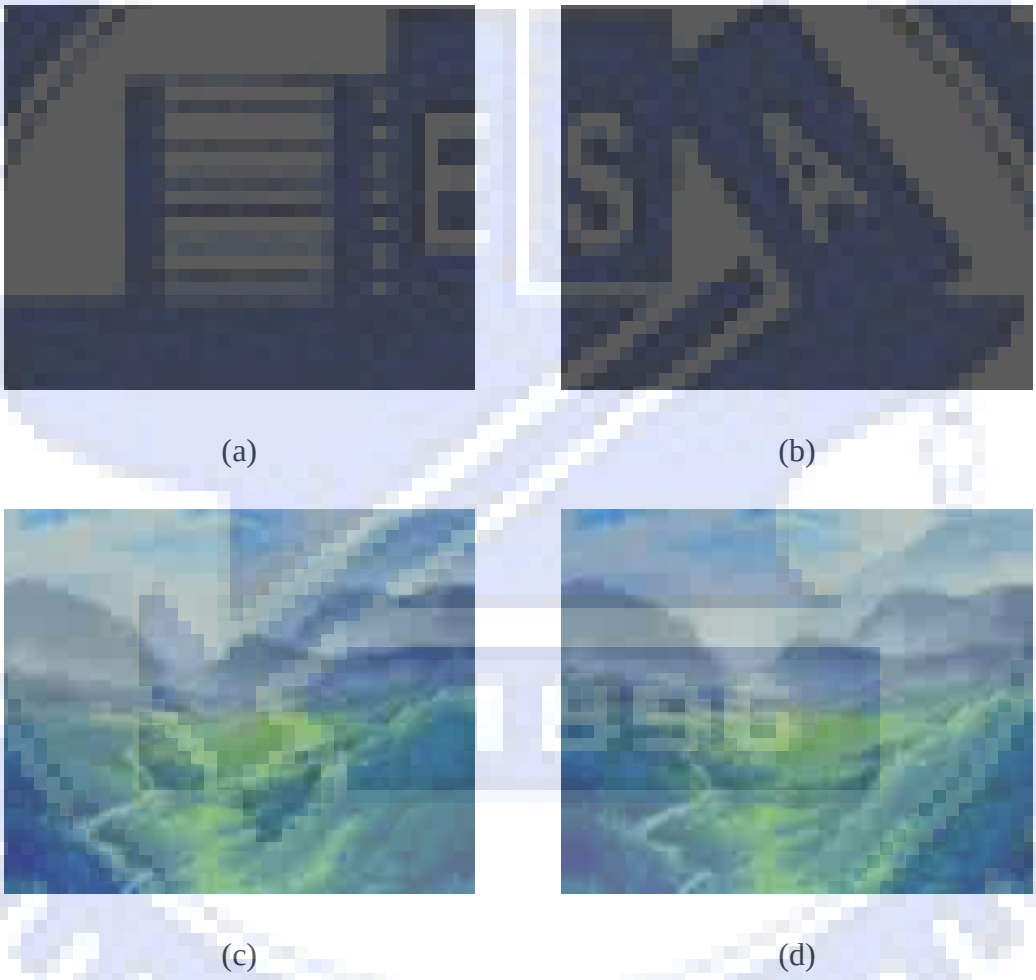


Figure 5.4 A flowchart of secret message extracting process

5.3. Experimental Results

In our experiments, a video in which a secret message is embedded is put on the Internet. When a user gets the video, the secret message can be extracted and recovered by the use of the user key. The video with a secret message embedded is shown in Figure 5.5.



(a) (b)

(c) (d)

Figure 5.5 Comparison between the original video and the stego-video.(a) (c) (e) (g) (i) (k) The original frames of the video. (b) (d) (f) (h) (j) (l) The frames after a secret message is embedded with a user key.

(e) (f)

(g) (h)

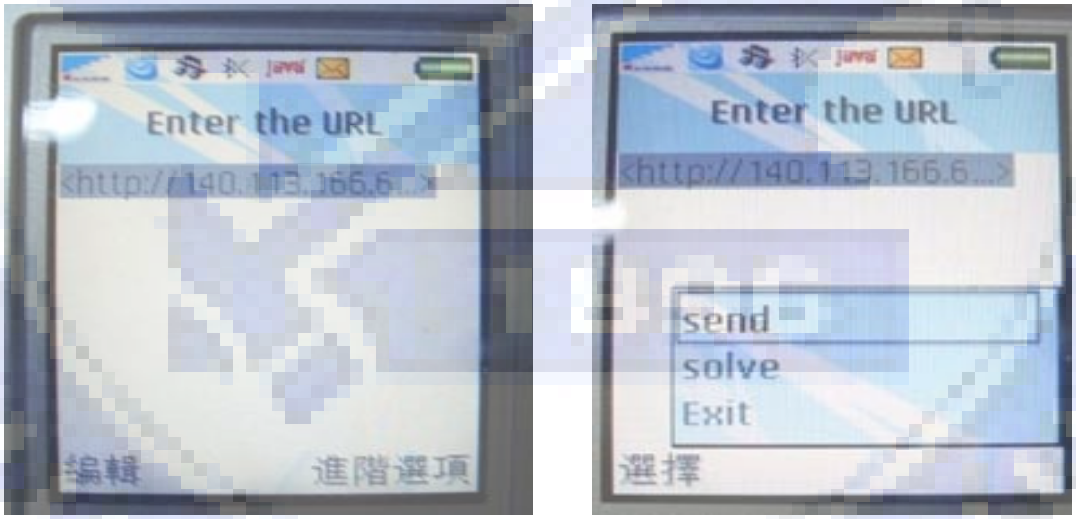
(i) (j)

Figure 5.5 Comparison between the original video and the stego-video.(a) (c) (e) (g) (i) (k) The original frames of the video. (b) (d) (f) (h) (j) (l) The frames after a secret message is embedded with a user key(continued).

(k) (l)

Figure 5.5 Comparison between the original video and the stego-video.(a) (c) (e) (g) (i) (k) The original frames of the video. (b) (d) (f) (h) (j) (l) The frames after a secret message is embedded with a user key(continued).

Then the process of getting a secret message from a stego-video is shown in Figure 5.6.



(a) (b)

Figure 5.6 Process of getting a secret message from a stego-video.(a) Downloading movies by keying in a public URL. (b) Sending the request. (c) Inputting the key to extract and recover the secret message. (d) The secret message extracted by a correct key. (e) Inputting a wrong key. (f) The result of extracting the secret message by a wrong key.

(c)

(d)

(e)

(f)

Figure 5.6 Process of getting a secret message from a stego-video.(a) Downloading movies by keying in a public URL. (b) Sending the request. (c) Inputting the key to extract and recover the secret message. (d) The secret message extracted by a correct key. (e) Inputting a wrong key. (f) The result of extracting the secret message by a wrong key(continued).

5.4. Discussions

In this chapter, a covert communication technique by videos has been proposed. The video files used here are very large. Hence, the size of the secret message can be huge. This is an advantage of the proposed method. Not only text messages but also pictures can be embedded into videos for transmission. A key is used to increase the security of covert communication. Covert communication by mobile devices is very convenient. By the proposed method, large volumes of secret data can be transferred securely.

Chapter 6
Secret Message Sharing on Mobile Phones by An Information Sharing Technique

6.1. Introduction

With the increasing of the population of carrying mobile phones, sharing secret messages by the mobile phone is very convenient. Shares of a secret can be stored in the mobile phone and collected together by the SMS to compose the original secret message.

The size of a message transmitted between mobile phones is usually limited. Hence, the size of a share cannot exceed the size limit of a message. On the other hand, we adopt the basic idea of secret sharing proposed by Huang and Tsai [6] as mentioned in Chapter 2. And some modification of their method is made in this study to fit in with the size limit of short messages.

The proposed secret sharing method is described in Section 6.2. Some experimental results will be shown in Section 6.3. Finally, some discussions will be made in Section 6.4.

6.2. Proposed Secret Sharing Method

The system configuration of the proposed method is described in Figure 6.1. Some application can adopt this method. For example, the manager of several agents

can assign a job to the agents that must be completed together. The manager creates shares of the content of the job and then uploads the shares on the Internet. Then, each agent can download one piece of shares by the client program on his/her mobile phone and transmits the share by SMS to a specific agent who can recover the content of the job.

In this section, the focus is on the employed process of secret sharing. In Section 6.2.1, a process of creation of secret shares from a pure text secret is described. In Section 6.2.2, a process of fitting a large noise share into a mobile phone message is described, and a process of disguising noise shares by a steganographic scheme is also proposed. Finally, a process of pure text secret recovery is described.

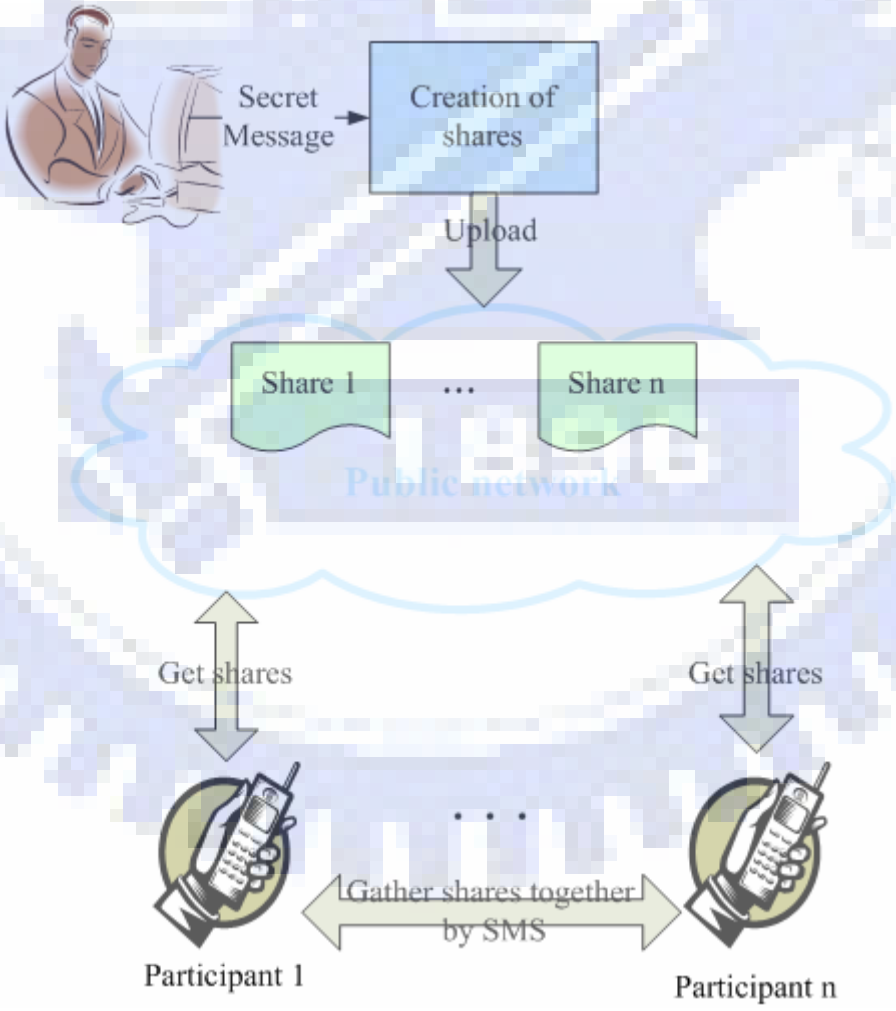


Figure 6.1 System configuration of proposed method.

6.2.1. Creation of Secret Shares

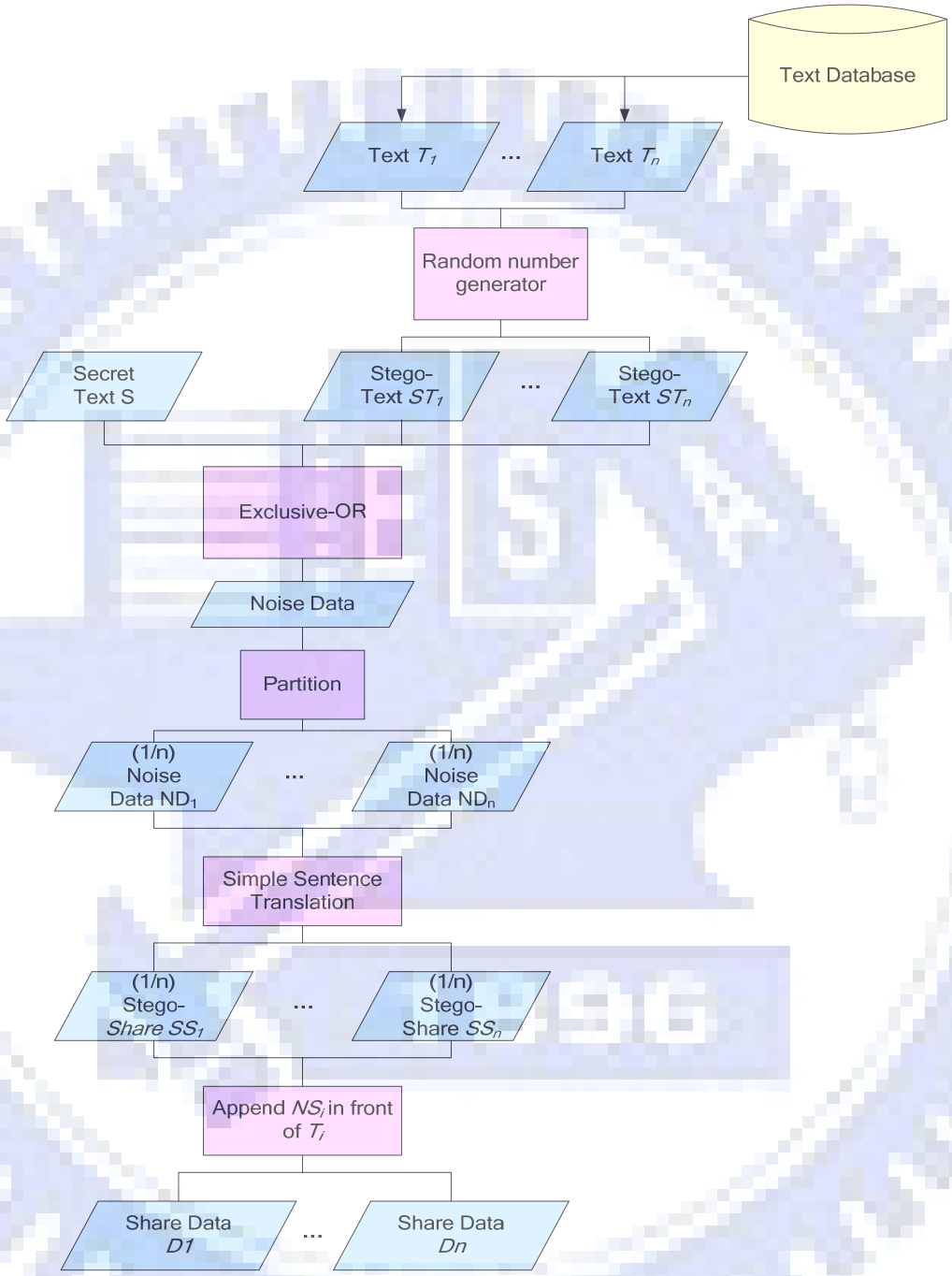


Figure 6.2 Flowchart of secret pure text sharing process.

Figure 6.2 shows the process of sharing a secret text *S* among participants and creating steganographic effects on the noise data yielded by the secret sharing process. Suppose that the number of participants is *n*. First, *n* texts T_1, T_2, …,T_n are selected

from a text database. Second, these n texts are taken as input to a random number generator to generate n stego-texts with their lengths all the same as that of S. Third, exclusive-OR S with the n stego-texts to generate a string of *noise data*, which is a sequence of meaningless bytes. The string is then partitioned into n pieces, denoted as ND_1, ND_2, …, ND_n. They are then translated into simple sentences to form n *stego-shares* SS_1, SS_2, …, SS_n which are meaningful. Finally, each SS_i is appended to the front of T_i to produce n final shares. The procedure of creation of secret shares is described as an algorithm in the following.

Algorithm 6.1: Creation of secret shares.

Input: Secret text S with length L in bytes, a text database *TDB*, and the number n of participants.

Output: n pieces of share data D_i, for $i = 1, 2, \ldots, n$.

Steps:

1. For $i = 1$ to n, choose a piece of text T_i from *TDB*.
2. For $i = 1$ to n, take T_i as input to a random number generator to generate one piece of stego-texts ST_i with L bytes.
3. Create an empty text PN with L bytes. Set $PN = ST_1$.
4. For $i = 2$ to n, do the following steps.
    i. For $k = 1$ to L, set $PN(k) = PN(k) \oplus ST_i(k)$, where $\oplus$ means the exclusive-OR operation.
5. Partition PN into n pieces of data, denoted as ND_i, for $i = 1$ to n.

6. For i=1 to n, translate ND_i into a simple sentence to produce a piece of text, denoted as SS_i.

7. For i=1 to n, append SS_i to the front of Ti to produce a piece of share denoted as D_i.

6.2.2. Fitting Large Noise Share into Mobile Phone Message

There are two steps made in this study to fit the size of shares to the size limit of mobile phone messages. First, the stego-texts are generated as short-lengthed texts. Because of the variation of the lengths of secret texts, the texts selected from the text database cannot be used directly. However, this problem can be solved by using a random number generator to generate stego-texts with lengths all the same as that of the secret message. Hence, if the length of a secret text is out of the size limit of the SMS, we can still use the short texts which fit the size limit of the SMS.

Second, the exclusive-ORed noise data is partitioned into n pieces. This step is to disperse the content of the noise data.

6.2.3. Steganographic scheme for Disguising Noise Share

As mentioned in Section 6.2.1, the n pieces of noise data are translated to simple sentences by a *sentence translation machine* designed in this study. First, the machine transforms the noise data to its binary format. Then, the result is divided into several parts and each part is encoded into a simple sentence by certain sentence patterns and a database of words.

The technique uses four sentence patterns, the subject-verb-object pattern, the subject-be-adjective pattern, the pronominal-be-number-object pattern, and the subject-be-article-career pattern. Each pattern can be represented with two bits. And five databases, a name database, a verb database, an adjective database, a career database, and, a noun database are designed. The index of each word in the databases represents a sequence of bits. Before the binary format of the noise data is encoded, the length and the index of the noise data will be encoded first. A flowchart of the proposed noise data translation process is shown in Figure 6.3 and the details are described in Algorithm 2.

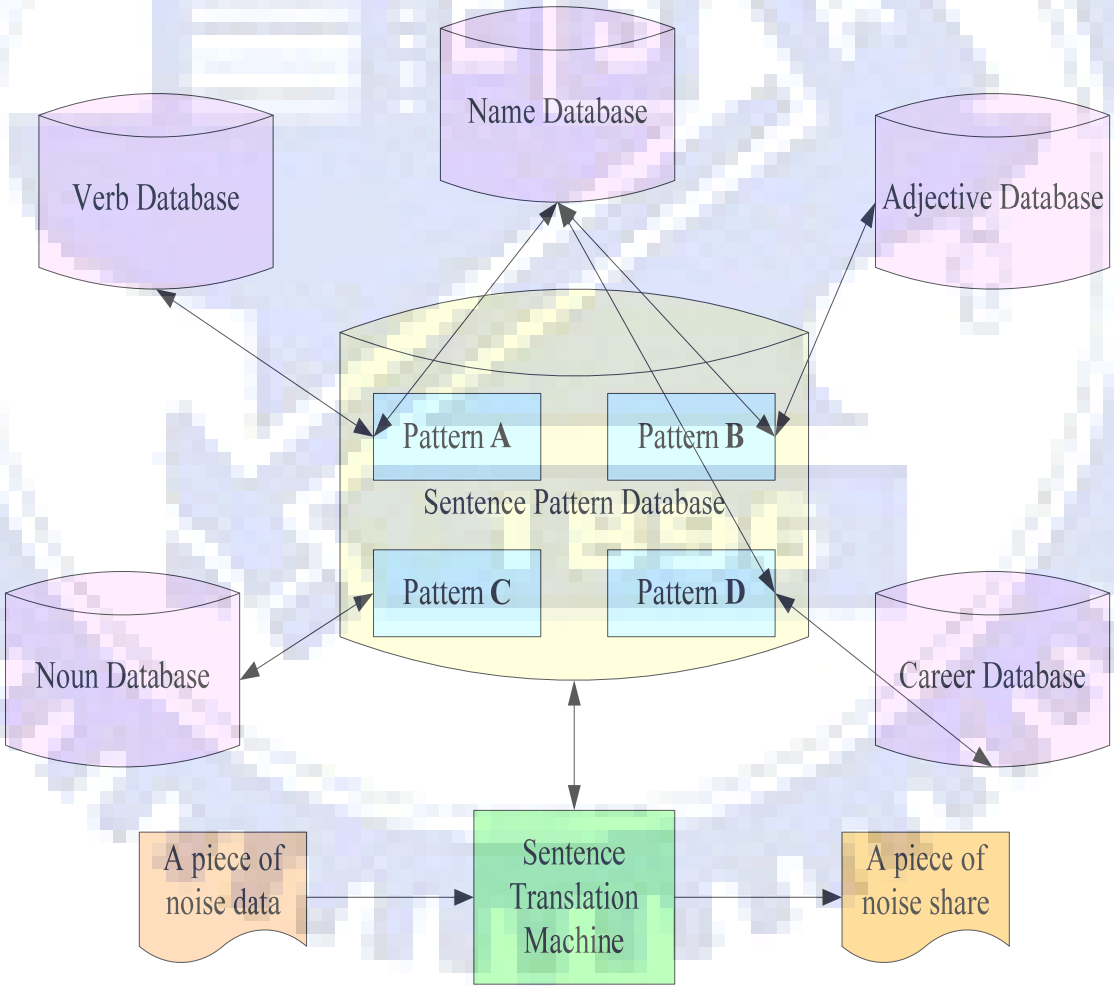


Figure 6.3 Flowchart of noise data translation process.

Some examples of words in the five databases are shown as follows:

(1) The name database:

Aaron	Abiah	Adam	Alric	Alden
Abel	Abner	Addis	Alastair	Aldred
Abelard	Abraham	Adonis	Alban	Alexander
Aberhard	Achilles	Ajax	Albert	Alexis
Adelard	Adair	Alan	Albion	Alfred

(2) The verb database:

love	like	Praise	approach	free
hate	dislike	murmur	explain	catch
beat	scold	hit	stare	find
interview	rebuke	see	forgive	fire
lie	admire	summon	contact	kick

(3) The adjective database:

nascent	implicit	absorbed	crapulous	ideal
dying	independent	abstemious	crapulent	objective
moribund	living	abstinent	edacious	real

last	receptive	ascetic	greedy	thick
dead	thirsty	gluttonous	hoggish	rare

(4) The career database:

accountant	architect	baker	blacksmith	driver
actor	artist	barber	boxer	butcher
actress	astronaut	player	broker	buyer
anchor	attendant	bellhop	agent	carpenter
announcer	auditor	gardener	lawyer	cartoonist

(5) The noun database:

animals	drafts	clips	apples	loops
plants	guns	boxes	grapes	devices
brushes	chops	tags	strawberries	roads
records	headers	flares	watermelons	crutches
blots	liners	castles	melons	bolts
balls	singles	banks	muskmelons	oranges

And some examples of sentences generated with the four sentence patterns are:

(1) Albert hates Albion.

(2) Emerald is hoggish.

(3) There’re 40 apples.

(4) Andrew is a director.

Algorithm 6.2: Translation of noise data to simple sentences.

Input: A noise data ND_k where k is an index.

Output: An encoded string S_e.

Steps:

1. Transform ND_k to its binary format denoted as *BND*. Set n as the number of bits in *BND*.
2. Transform k and n to their binary forms, and denote the results as *bk* and *bn*.
3. Combine *bk*, *bn*, and *BND* together to form a bit string B_c. Let L be the number of bits in B_c. And denote B_c as $B_c = \{b_i \mid i = 1, 2, \ldots, L\}$.
4. Set $j = 1$. While $j <= L$, perform the following steps.
    i. Extract the initial two bits of B_c, starting at j, and denote the result as T. If T is “00,” go to Step 5; if T is “01,” go to Step 6; if T is “10,” go to Step 7; and if T is “11,” go to Step 8. Then set $j = j+2$.
5. Apply the sentence pattern A, the subject-verb-object pattern. Perform the following steps.

i. Extract the initial 9 bits of B_c, starting at j and denote the result as S. Replace S with the item whose index is S in the name database and append the item to S_e. Set $j = j+9$.

ii. Extract the initial 8 bits of B_c, starting at j and denote the result as V. Replace V with the item whose index is V in the verb database and append the item to S_e. Set $j = j+8$.

iii. Extract initial 9 bits of B_c started at j and denote as O. Replace O to the item whose index is O in the verb database and append the item to S_e. Then, the punctuation "." and a space are appended to the back of the sentence. Finally, set $j = j+9$.

iv. Go to Step 4.

6. Apply the sentence pattern B, the subject-be-adjective pattern. Do as follows.

i. Extract the initial 9 bits of B_c starting at j and denote the result as S. Replace S with the item whose index is S in the verb database and append the item to S_e. Set $j = j+8$.

ii. Extract the initial 1 bits of B_c starting at j. If the result is "0," append "is" to S_e; else, append "was" to S_e. Set $j = j+1$.

iii. Extract the initial 8 bits of B_c starting at j and denote the result as Adj. Replace Adj with the item whose index is Adj in the adjective database and append the item to S_e. Then, the punctuation "." and a space are appended to the back of the sentence. Finally, set $j = j+8$.

iv. Go to Step 4.

7. Apply the sentence pattern *C*, the subject-be-number-object pattern. Do as follows.

   i. Extract the initial 2 bits of B_c starting at j. If the result is "00," append "There're" to S_e. If the result is "01," append "Those're" to S_e. If the result is "01," append "These're" to S_e and if the result is "01," append "They're" to S_e. Set $j = j+2$.

   ii. Extract the initial 10 bits of B_c starting at j and denote the result as *Num*. Set *Num* = *Num*+2. Append *Num* to S_e. Set $j = j+10$.

   iii. Extract the initial 8 bits of B_c starting at j and denote the result as *N*. Replace *N* with the item whose index is *N* in the noun database and append the item to S_e. Then, the punctuation "." and a space are appended to the back of the sentence. Finally, set $j = j+8$.

   iv. Go to Step 4.

8. Apply the sentence pattern D, the subject-be-article-career pattern. Do as follows.

   i. Extract the initial 9 bits of B_c starting at j and denote the result as *S*. Replace *S* with the item whose index is *S* in the name database and append the item to S_e. Set $j = j+9$.

   ii. Extract the initial 1 bits of B_c starting at j. If the result is "0," append "is" to S_e. Else, append "was" to S_e. Set $j = j+1$.

   iii. Extract the initial 8 bits of B_c starting at j and denote as *Ca*. Replace *Ca* with the item whose index is *Ca* in the career database and append the item to S_e. Then, the punctuation "." and

a space are appended to the back of the sentence. Finally, set $j = j+8$.

iv. Go to Step 4.

9. Take S_e as the desired output.

6.2.4. Secret Pure Text Recovery Process

After all shares are collected together by the SMS, the secret message can be recovered by a recovery process. First, the stego-share and the text are taken apart from the share. Second, for each sentence in the noise share, the index of the sentence pattern is extracted and every word is translated to a binary sequence by searching the index in the databases. All the binary data are sorted together by the encoded index into a sequence of bits. Then, the result is converted into the corresponding noise data in byte form. The texts are put into a random number generator to generate sequences of bytes denoted as stego-texts whose lengths are all equal to that of the noise data. For each stego-text, the exclusive-OR operation is performed to every byte of the noise data and the stego-text. The recovered result is then just the desired secret text. A flowchart of the recovery process is shown in Figure 6.4.

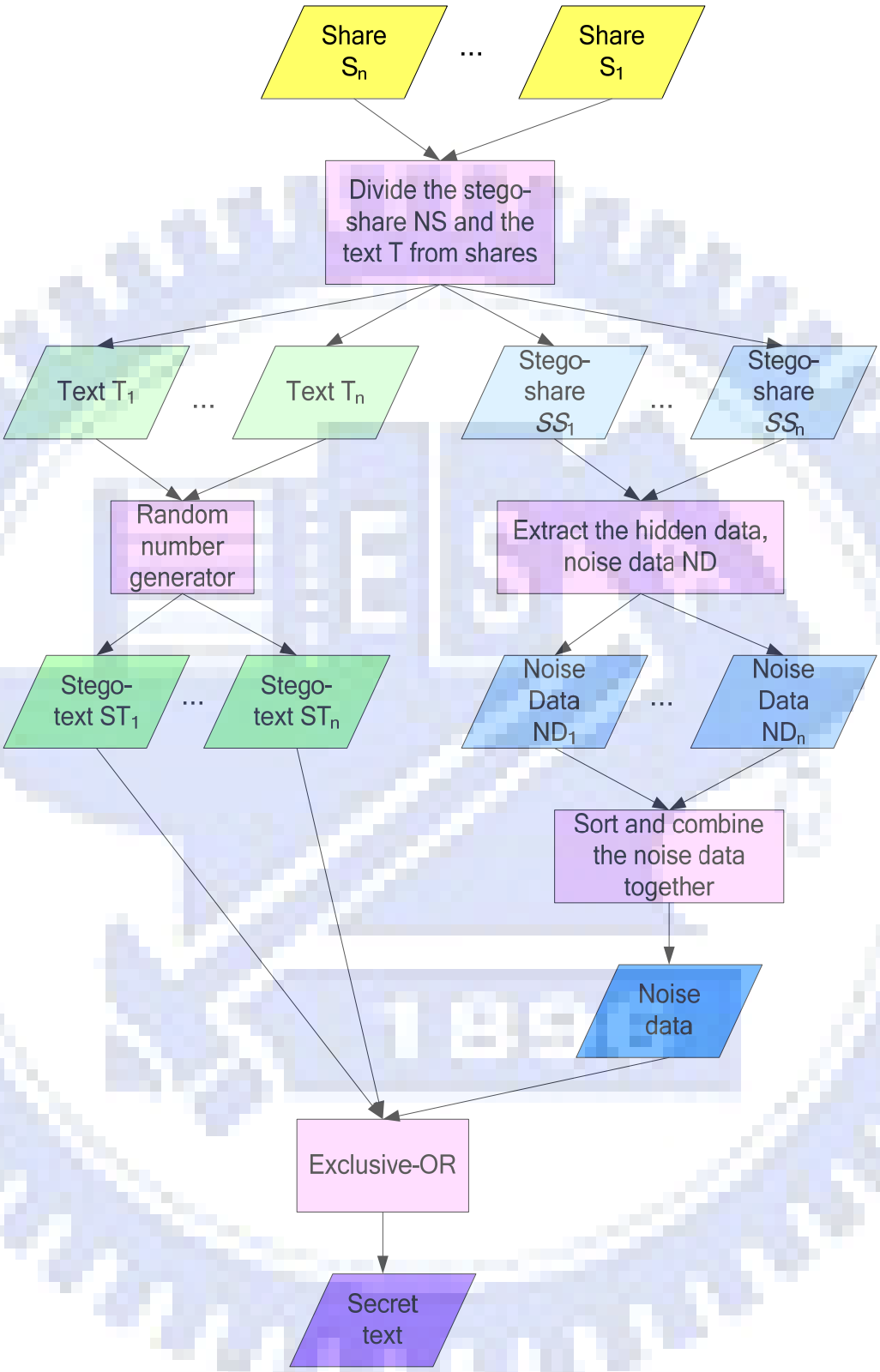


Figure 6.4 Flowchart of the secret recovery process.

6.3. Experimental Results

In our experiments, the shares and the cover texts are shown in Figure 6.5.

Andrew is a director. Emerald is hoggish. Alaster was potential. Becky abducts Jean. It was proposed we go to the station to meet our guests.	Cordell is a merchant. Ishara is a geologist. Emerald was living. Aaron is an accountant. Don't be late for your interview, or you won't get the job.
(a)	(b)
It was proposed we go to the station to meet our guests.	Don't be late for your interview, or you won't get the job.
(c)	(d)

Figure 6.5 Illustration of shares.(a) (b) Two shares produced with secret message “Good morning 123”. (c) (d) The texts.

Then the procedure of sharing secret via mobile devices is shown in.Figure 6.6.

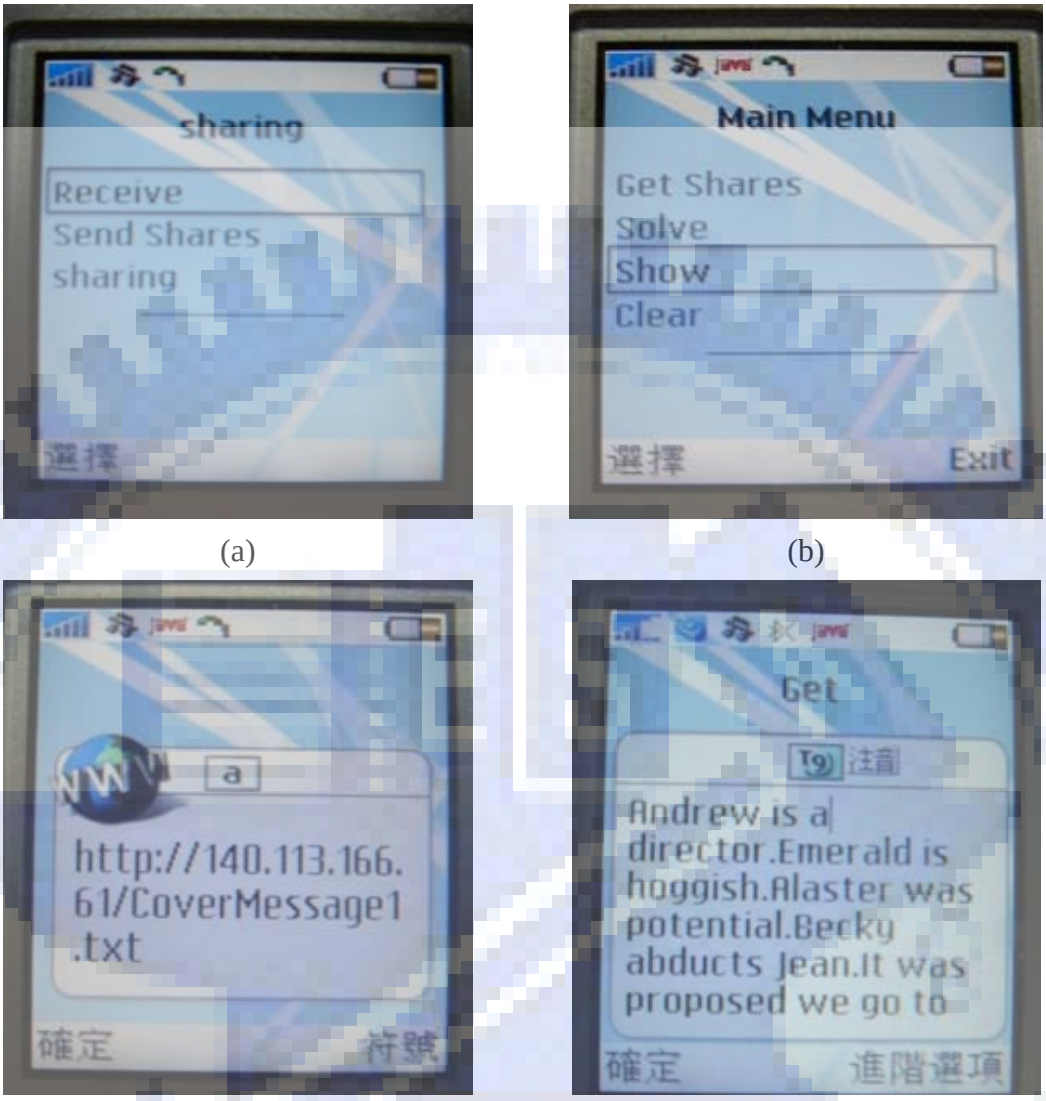
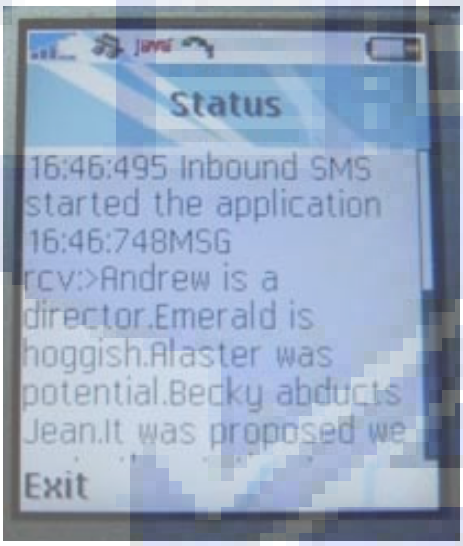


Figure 6.6 Illustration of secret sharing process on mobile phones. (a) The menu of the client program including the “Send shares” which can send the stored shares by the SMS, the “Receive” which can receive the shares by the SMS, and the “sharing” which can get the shares in the Internet, show the shares, solve the secret message and, clear all data. (b) The menu of the sharing program. (c) (d) Get shares in the Internet. (d) (f) The shares. (g) Receive the shares by the SMS. (h) The result of solving the secret message with all shares.

(e)

(f)



(g)

(h)

Figure 6.6 Illustration of secret sharing process on mobile phones. (a) The menu of the client program including the “Send shares” which can send the stored shares by the SMS, the “Receive” which can receive the shares by the SMS, and the “sharing” which can get the shares in the Internet, show the shares, solve the secret message and, clear all data. (b) The menu of the sharing program. (c) (d) Get shares in the Internet. (d) (f) The shares. (g) Receive the shares by the SMS. (h) The result of solving the secret message with all shares(continued).

6.4. Discussions

In this chapter, a text secret sharing method via mobile devices has been proposed. The shares are produced with a steganographic scheme. And because of the size limit of the SMS, some techniques are proposed to solve the problem. Sharing secret by mobile phones is very convenient and securely. It is easy and convenient to get shares by mobile network, store shares in mobile phones and collect shares by the SMS for users.

Chapter 7
Text Secret Authentication on Mobile Phones

7.1. Introduction

The total number of the SMS sent is up to million everyday. In Taiwan, there are almost 1 million pieces of short messages sent per PSP (Personal communication Services Providers). A user uses the SMS to send short messages to his/her friends and family and companies use the SMS to send information to their customers.

Because of the convenience of the SMS, many banks which provide the service of credit card use the SMS to send bills to their customers. But the swindlers also use the SMS. They pretend to be banks and send fake SMSs to cheat customers.

In order to solve this problem, we propose an authentication technique for the SMS which can verify the content of the SMS and the publisher of the SMS. In Section 7.2, the proposed method for message authentication will be described. In Section 7.3, some experimental results will be shown and in Section 7.4, some discussions will be made.

7.2. Proposed Method for Message Authentication

In order to achieve our goal, we propose the technique of creation of authentication signals and embedding the signals into short messages. A receiver can

use a receiving program to verify received short messages. A procedure of the proposed method is shown in Figure 7.1.

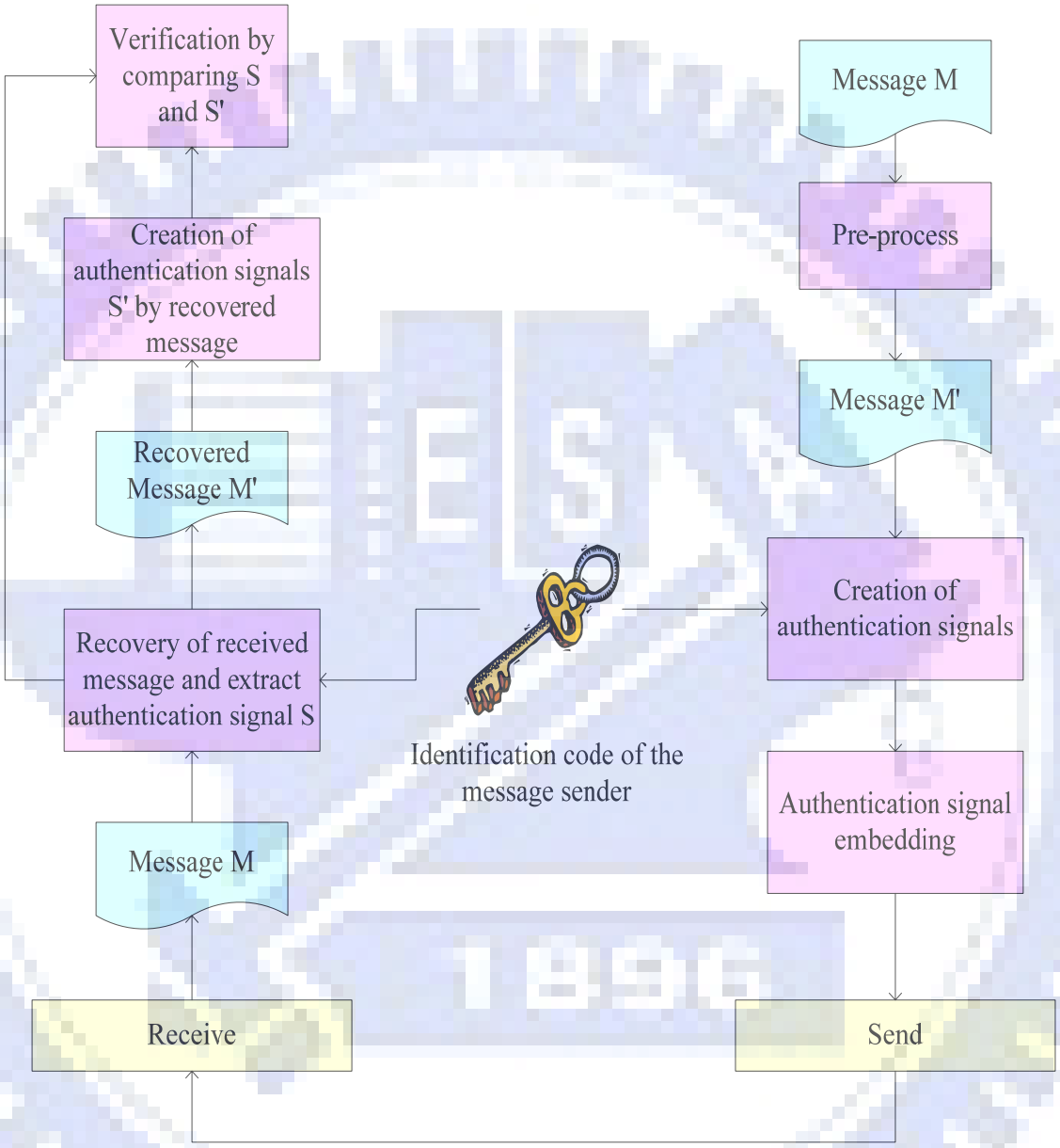


Figure 7.1 Procedure of the proposed method.

Because the property of English messages is different to that of Chinese messages, the methods of embedding authentication signals for English and Chinese message are also different. Hence, the pre-processing works of English and Chinese

messages before creating authentication signals are different. In Section 7.2.1, the proposed method of embedding authentication signals will be described. In Section 7.2.2, the process of creation of authentication signals will be described. And In section 0, the process of verification of the content of the SMS will be described.

7.2.1. Process of Embedding Authentication Signals

The authentication signal embedded into a short message can be viewed as a watermark signal. If the embedded authentication signal is destroyed, we will view the received short message as *unbelievable*.

In this section, the process of embedding authentication signals will be described. Because of the difference between English and Chinese messages, the techniques of embedding authentication signals for them are different. In Section 7.2.1.1 and Section 7.2.1.2, the procedures of embedding in English and Chinese messages will be described, respectively.

7.2.1.1. Embedding Authentication Signals in English Messages

English sentences are composed of words separated by spaces and there is also a space between every two adjacent sentences. In this study, we use this property to embed authentication signals in English messages.

Because one or two spaces between two consecutive words are both accepted by human vision, it does not change the meaning of sentences after adding another space between words. Hence, we may use the number of spaces between words to embed authentication signals.

If the bit we want to embed is 0, the number of spaces between two words is kept to be only one; else, another space will be added between the two words. The procedure of embedding authentication signals in English messages is shown in Figure 7.2.

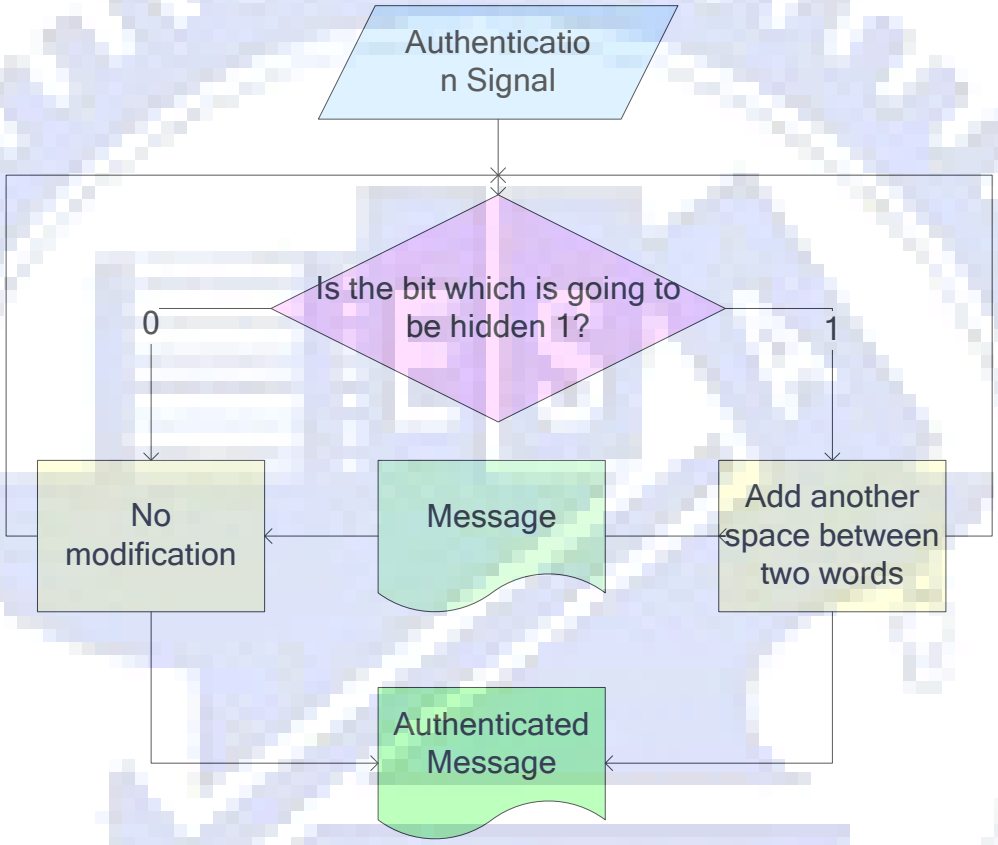


Figure 7.2 Procedure of embedding the authentication signal in English messages.

7.2.1.2. Embedding Authentication Signals in Chinese Messages

In Chinese messages, punctuations are used to separate words or sentences. There are two types of punctuations: English punctuation and Chinese punctuation. When people understand meanings of Chinese messages during reading, there is no difference between using English punctuations and using Chinese ones. Therefore, we use punctuations in different ways to embed bits: using the Chinese punctuation to

embed a bit “1,” and using the English punctuation to embed a bit “0.”

Except the embedding technique described in the previous paragraph, we also propose another data embedding technique. We use a “two-byte space,” a “one-byte space”, and “no space” to encode “11,” “10,” and “0” before sentences in order to construct spaces to embed authentication signals.

The process of embedding authentication signals in Chinese messages is shown in Figure 7.3.

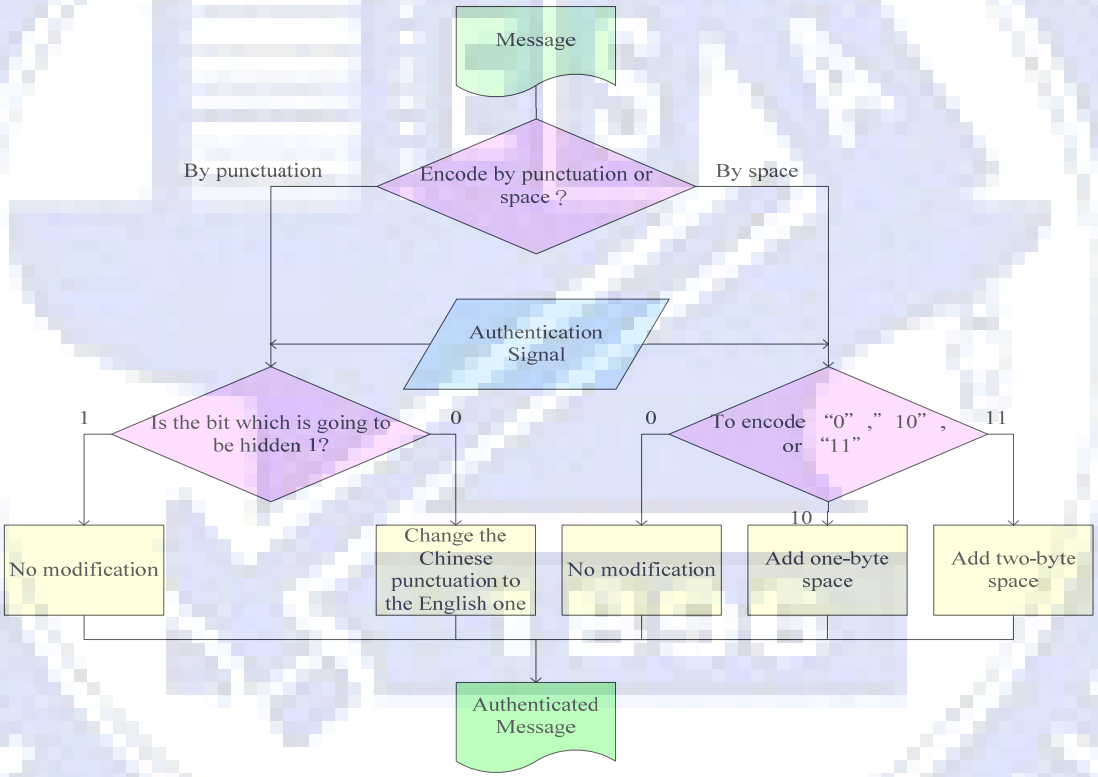


Figure 7.3 Process of embedding authentication signals in Chinese messages.

7.2.2. Creation of Authentication Signals

Each authentication signal proposed in this study is composed of two parts: a hash value of the content of a short message and an identification code of the short message sender. It can help a receiver to verify the content of the message and the

identity of the sender.

Before creating of the authentication signal, the message should pass through a pre-processor before performing the following process of embedding of the authentication signals. Because of the difference between English and Chinese messages, the pre-processors for English and Chinese messages are also different.

The pre-processor of English messages is to remove redundant spaces between words because the technique of embedding authentication signals of English messages depend son the number of spaces between words. The pre-processor of Chinese messages is to change the English punctuation into the corresponding Chinese punctuation and remove the redundant spaces between sentences because the authentication signal is embedded by the difference between English and Chinese punctuations and the types of spaces between sentences.

In the proposed method, we use the MD5 algorithm to generate the hash value of the message content. The reason why the MD5 is adopted is that the length of the message is indefinite. The MD5 algorithm can generate a 16-byte output with inputs of variable lengths.

Every sender has an identification code. The second step of the method is to generate a 1-byte authentication signal which is composed of the identification code and the 16-byte output generated by the MD5 algorithm of the message content. The detailed algorithm of creation of the authentication signal is described as follows and a corresponding flowchart is shown in Figure 7.4.

Algorithm 7.1: Creation of authentication signals.

Input: A pre-processed message M and an identification code K of a short message sender.

Output: A 1-byte authentication signal S.

Steps:.

1. Take M as the input to the MD5 function to generate a 16-byte hash value D and denote D as follows:

   $$D = \{D_i \mid i=1, 2, \ldots, 16\}.$$

2. Take K as the input to a random number generator to generate eight *thresholds* with their values being in the range of 0 through 15. Denote the thresholds as $T_1, T_2, \ldots, T_8$.

3. For i = 1 to 8, count the number of bits with its value being "1" of the i-th bit of $D_1, D_2, \ldots, D_{16}$ and denote the result as N_i.

4. For i = 1 to 8, set the i-th bit of S as follows:

   $$\begin{cases} \text{Set the } i\text{-th bit of } S \text{ as 1, if } N_i > T_i \\ \text{Set the } i\text{-th bit of } S \text{ as 0, else} \end{cases}$$

5. Take the final S as the desired authentication signal.

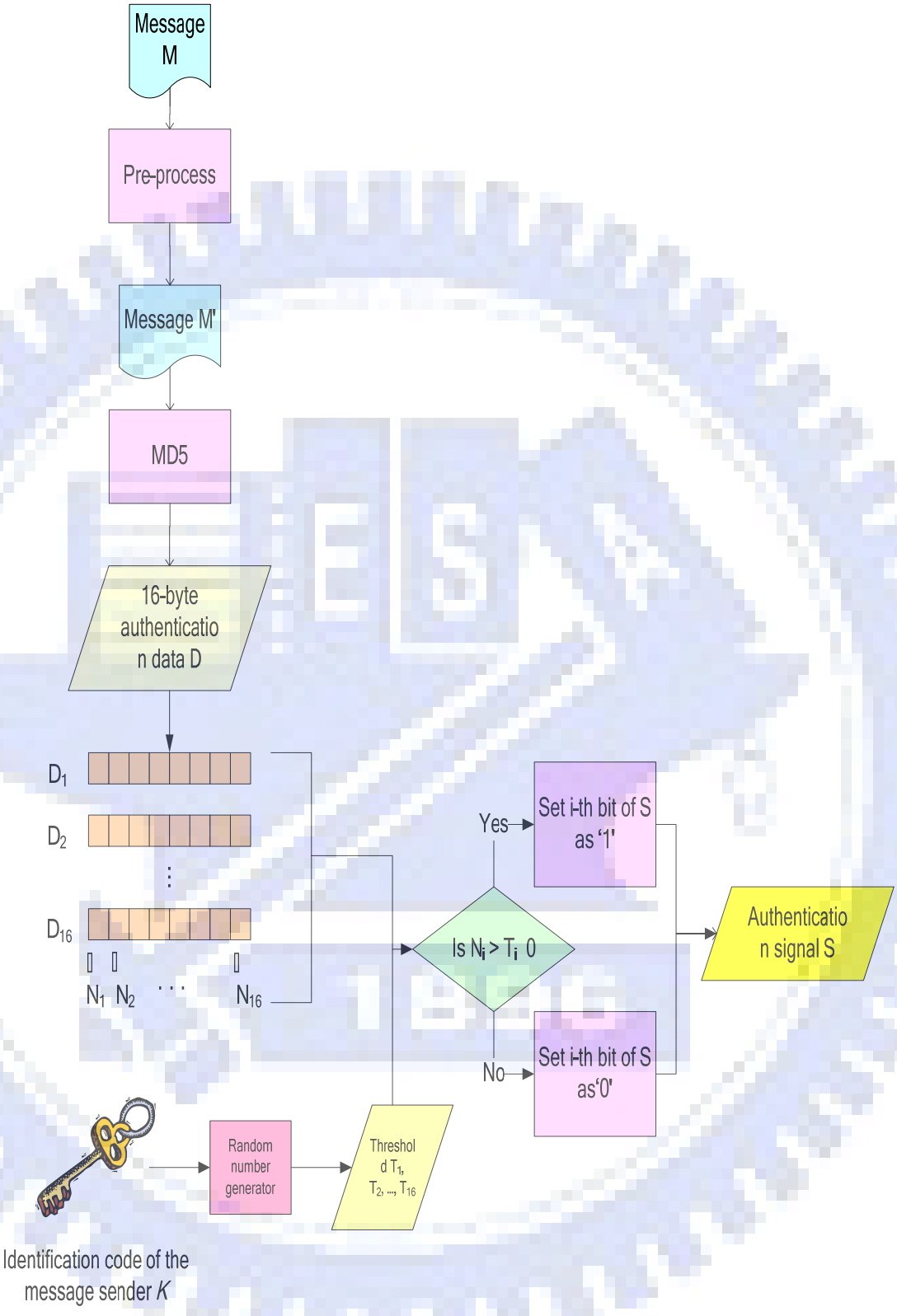


Figure 7.4 Flowchart of creation of the authentication signals.

7.2.3. Verification of Authenticated Messages

After receiving an authenticated message, the client program can help users to verify the validity of the message, which we call an authentication process. Figure 7. 5 shows a flowchart of the proposed authentication process. First, the embedded authentication signal set *AS′* is extracted and the original message set *M′* is recovered from the authenticated message. The detailed process of extracting the authentication signal from the authenticated message depends on the embedding process for English or Chinese messages. Second, another authentication signal *AS*" is generated by the process of creation of authentication signals described in Section 7.2.2. Finally, AS′ and AS" is compared to check the authenticity of received messages. The steps to verify the fidelity of a suspicious short message are described as follows.

Step 1: Extract the hidden data by the rules of embedding of authentication signals, resulting in an authentication signal set *AS'*; and recover the received message to obtain the original message set *M'*.

Step 2: Generate another authentication signal set *AS"* by the process of creation of authentication signals with the identification code of the short message sender for *M'*.

Step 3: Compare *AS′* and *AS"* to verify the fidelity of a suspicious short message.

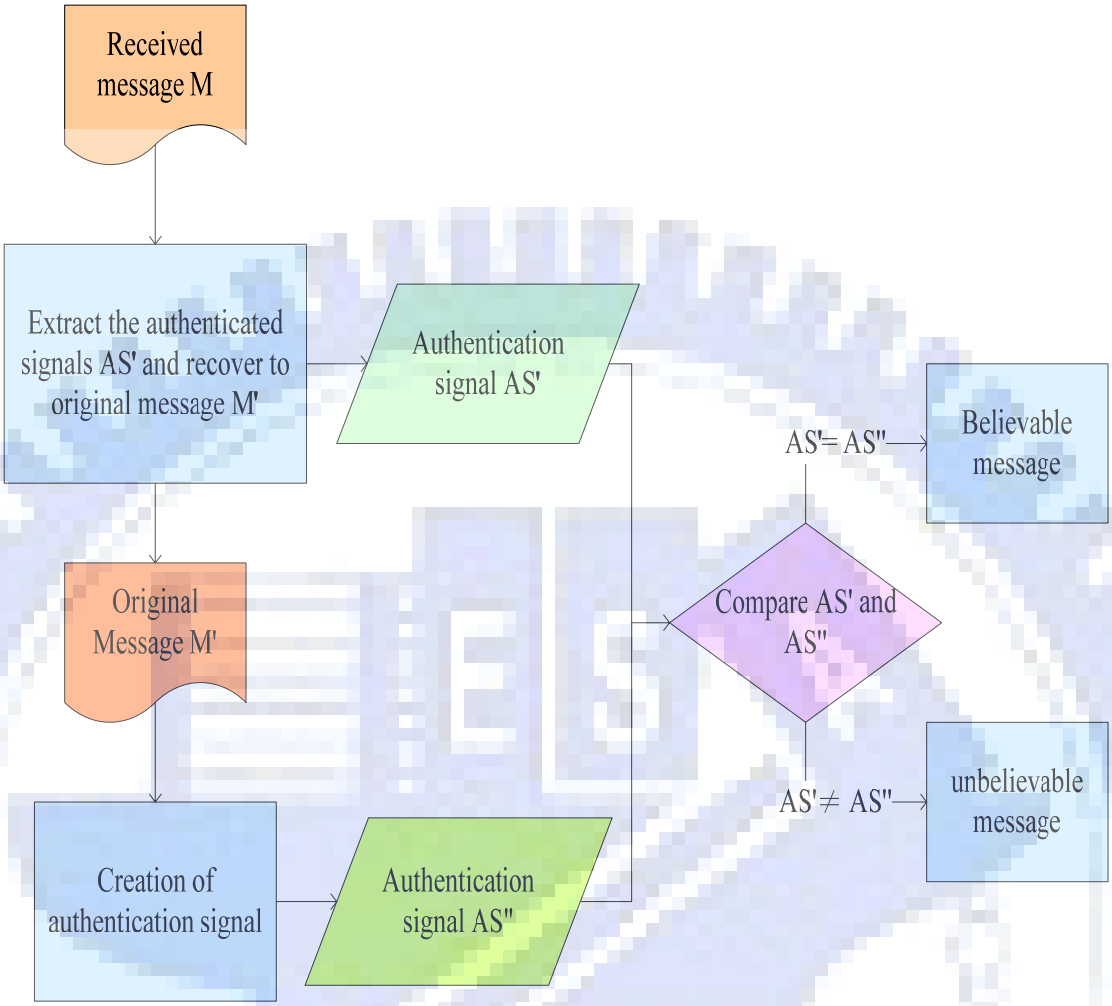


Figure 7. 5 Flowchart of the authentication process.

7.3. Experimental Results

In our experiments, two authentication results of Chinese and English messages are shown in Figure7.6 and Figure 7.7, respectively. After users receive the message, they can verify the content and the sender of this message. The result is shown in Figure 7.8.

這是一個測試。關於簡訊（SMS）的驗證。可以驗證：訊息內容、發送單位。驗證訊號會被嵌入，但是使用者無法察覺。
(a)
這是一個測試。關於簡訊（SMS）的驗證。可以驗證：訊息內容、發送單位。驗證訊號會被嵌入，但是使用者無法察覺。
(b)
這是一個測試。關於簡訊(SMS)的驗證.可以驗證：訊息內容、發送單位.驗證訊號會被嵌入，但是使用者無法察覺。
(c)

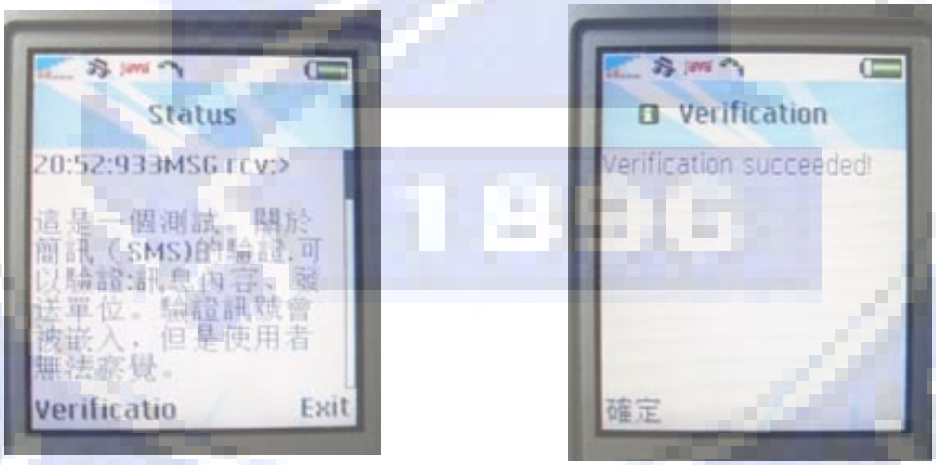
Figure7.6 Illustration of a Chinese short message.(a) The original Chinese short message. (b) The message after preprocessed. (c) The message after the authentication signals is embedded.

Until several years ago, there was ample incentive: the Pakistani economy was doing relatively well while Bangladesh was an economic basket case.
(a)

Figure 7.7 Illustration of an English short message.(a) The original English message. (b) The message after preprocessed. (c) The message after the authentication signals is embedded.

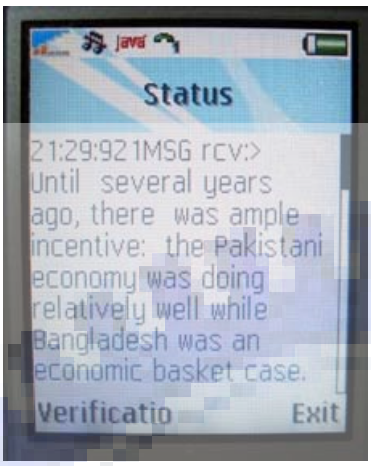
Until several years ago, there was ample incentive: the Pakistani economy was doing relatively well while Bangladesh was an economic basket case.
(b)
Until several years ago, there was ample incentive: the Pakistani economy was doing relatively well while Bangladesh was an economic basket case.
(c)

Figure 7.7 Illustration of an English short message.(a) The original English message. (b) The message after preprocessed. (c) The message after the authentication signals is embedded(continued).



(a) (b)

Figure 7.8 Illustration of an authentication process of an English short message. (a) When user receive a Chinese message. (b) After the “Verification” button is pressed. (c) When user receive an English message. (d) After the “Verification” button is pressed.



(c)

(d)

Figure 7.8 Illustration of an authentication process of an English short message. (a) When user receive a Chinese message. (b) After the “Verification” button is pressed. (c) When user receive an English message. (d) After the “Verification” button is pressed(continued).

7.4. Discussions

In this chapter, an authentication technique for the SMS is proposed. The fidelity of a short message can be verified by this technique. By adding the identification code of the message sender, messages from various senders can be verified by a client program. This method is useful for protecting users from receiving forged messages on mobile phones.

Chapter 8
Conclusions and Suggestions for Future Works

8.1. Conclusions

In this study, we have proposed several information hiding methods with videos and texts as cover media. These methods are proposed for many application purposes, such as copyright protection, covert communication, text message authentication, and text secret sharing on mobile devices.

For copyright protection, we have proposed two kinds of methods for video copyright protection. First, the technique of restriction on a specified mobile device for video display has been proposed. The method can restrict users from sharing user account with unregistered people for download videos for displays. Second, the technique of lossless watermarking for videos has been proposed for copyright protection. The method can prevent the video files from being downloaded by the illegal users.

For covert communication, a method of embedding a secret message into a video file has been proposed. The change to the video with the secret message embedded is not sensitive to the human vision system. When a receiver gets the stego-video and provides the correct key, the secret message can be extracted.

For secret sharing, a method of secret sharing with steganographic effects has been proposed. Because of the limit of data size of the SMS, the shares are created by the proposed technique to fit the size limit.

For authentication, a method was proposed to verify the fidelity of a received short message. When a user gets a suspicious message, the authentication program will authenticate the content and the sender of the message.

8.2. Suggestions for Future Works

Several suggestions for future researches are enumerated as follows.

1. The proposed copyright protection method protects the copyright of videos on the server. Further researches about copyright protection of multimedia data on the local side can be studied.
2. The proposed covert communication method for video files can be improved by utilzing the inter-coded blocks.
3. Information hiding techniques, such as copyright protection, covert communication, and secret sharing, may be integrated in a new method.
4. Other file formats, such as audio files used in mobile devices, may be studied for information hiding techniques.

References

[1] H. Y. Chen and W. H. Tsai, “New Data Hiding and Watermarking Techniques for MPEG Videos and Their Applications,” Master’s Thesis, Department of Computer and Information Science, Nation Chiao Tung University, Hsinchu, Taiwan, R. O. C. , June 2003.

[2] H.Y Chen and W. H. Tsai, “Verification of MPEG Video Contents by Random Signal Hiding,” IPPR Conference on Computer Vision, Graphics, and Image Process, Kinmen, Taiwan, R. O. C., Aug. 16-18, 2003, pp. 692-701.

[3] Jie Song and K. J. R. Liu, “A Data Embedding Scheme for H.263 Compatible Video Coding,” Proceedings IEEE International Symposium on Circuits and Systems 1999, Orlando, U.S.A., May 30- June 02 1999, vol.4 pp. 390 – 393.

[4] J. Meng and S. F. Chang, “Embedding Visible Video Watermarks in the Compressed Domain,” Proceedings of IEEE International Conference on Image Processing, Chicago, IL, U. S. A. ,Oct. 1998, vol. 1, pp. 474-477.

[5] N. K. Lo and W. H. Tsai, “A Study on Active Information Hiding and Applications,” Master’s Thesis, Department of Computer and Information Science, Nation Chiao Tung University, Hsinchu, Taiwan, R. O. C. , June 2004.

[6] K. L. Huang and W. H. Tsai, “A Study on Information Sharing of Text-type Documents with Steganography and Authentication Capabilities,” Master’s Thesis, Department of Computer and Information Science, Nation Chiao Tung University, Hsinchu, Taiwan, R. O. C. , June 2004.

[7] Y. H. Chen and W. H. Tsai, “New Methods and Applications of Data Hiding in Images, Text-Type Documents, and Web Pages,” Master’s Thesis, Department of Computer and Information Science, Nation Chiao Tung University, Hsinchu,

Taiwan, R. O. C. , June 2003.

[8] “Universal Mobile Telecommunications System (UMTS); Transparent end-to-end streaming service; 3GPP file format (3GP),” ETSI TS 126 244 V6.3.0, March 2005.

[9] “Video Coding For Low Bitrate Communication,” DRAFT ITU-T Recommendation H.263, May 1996.

[10] Sony Ercisson, “Getting MIDlet firmware version info and IMEI number,” http://developer.sonyericsson.com/site/global/techsupport/tipstrickscode/java/p_java_100304.jsp

[11] Wikipedia, “International Mobile Equipment Identity,” http://en.wikipedia.org/wiki/IMEI

[12] Yi-Bing Lin, Wireless and Mobile Network Architectures, JOHN WELEY & SONS, N.Y., U.S.A., 2001

[13] Ze-Nian Li and Marks S. Drew, Fundamentals of Multimedia, Pearson Education, U.S.A., 2004.

[14] D. Cross and B. G. Mobasseri, “Watermarking for Self-authentication of Compressed Video,” Proceeding of IEEE International Conference on Image Processing, New York, USA, vol.2, pp. 913-916, Sept. 2002.

[15] Y. Takishima, M. Wada, H. Murakami, “Reversible Variable Length Codes,” Communications, IEEE Transactions, Volume 43, Issue 234, Feb-Mar-Apr 1995 pp. 158 – 162.

[16] Rahouma, K.H., “Utilization of multiple block cipher hashing in authentication and digital signatures,” Proceedings. IEEE International Conference, Las Vegas , U. S. A., 5-8 Sept. 2000 Page(s):257 - 261