Journal of Information Hiding and Multimedia Signal Processing ©2022 ISSN 2073-4212
Ubiquitous International Volume **13**, Number **4**, December **2022**

Multilevel Multipurpose Watermarking Schemes for PPT Documents

Lei Li, Hong-Jun Zhang

Command and Control Engineering College
Army Engineering University
Nanjing 210007, P. R. China
605531524@qq.com

Zhe-Ming Lu*

School of Aeronautics and Astronautics
Zhejiang University
Hangzhou, 310027, China
zheminglu@zju.edu.cn

Received September 2022; revised November 2022
(Communicated by Zhe-Ming Lu)

Abstract. *Multimedia courseware is an important resource in the network teaching system. With the spread of the Internet, if not guarded, ppt documents can easily be appropriated by others. In view of the more and more serious infringement problem, it is of practical significance to study how to effectively protect the copyright of multimedia courseware. Nowadays, there are few articles and research contents on watermarking of PPT documents, and there are almost no invisible watermarks in PPT documents. The invisible digital watermarking method for PPT documents in this paper is advanced and practical. Many articles on word document watermarking can be found through paper retrieval. Most of the research methods focus on modifying attributes such as font size and color to embed the watermark information, which has a single form and is not resistant to formatting attacks. However, these methods have given us a lot of inspiration, so that we can expand new methods from different angles, and make reasonable use of the unique components in PPT to embed watermarks, which ensures the security of watermarks and improves the robustness of watermarks. Experimental results demonstrate the effectiveness of the proposed methods.*
Keywords: Digital watermarking, PPT watermarking, Copyright protection, Digital fingerprinting.

1. **Introduction.** Multimedia courseware is an important resource in the network teaching system. However, while they bring great convenience to people, they also bring copyright disputes caused by illegal piracy and malicious tampering. With the spread of the Internet, if no precautions are taken, ppt documents can easily be taken by others. In view of the more and more serious infringement problem, it is of practical significance to study how to effectively protect the copyright of multimedia courseware.

After the online multimedia courseware is put into use, the risks faced by its copyright protection mainly include the following aspects:

(1) Use by illegal users. If the users of the network multimedia courseware are licensed users, it must be ensured that unauthorized users cannot access the network multimedia courseware.

(2) Dangerous behavior of legitimate users. Legal users refer to users who are permitted to use network multimedia courseware, but do not own copyright to network multimedia courseware and its contents. Their dangerous behaviors include: copying the entire courseware or part of the courseware, analyzing the source program of the courseware, etc. These dangerous behaviors are likely to lead to infringement of network multimedia courseware.

(3) Infringement. The infringer often uses his identity as a legitimate user to break through various restrictions and successfully copy the entire courseware or part of the courseware through various technical means. With the authorization of the copyright holder of the courseware, if the copied content is processed, or distributed directly on the network or in other forms, the rights and interests of the copyright holder of the online multimedia courseware will be infringed. It can be seen that the network multimedia courseware faces a variety of copyright protection risks in reality, and it is far from enough to take measures to prevent risks in a certain link of its production, release and post-maintenance.

Digital watermarking technology [1-5] is a mainstream copyright protection technology. It embeds copyright information (digital watermarking) such as numbers, serial numbers, texts, and image signs into carrier objects such as text, images, audio, and video to realize the function of copyright protection. Digital watermarking technology is an important research direction in the field of information hiding. Its basic idea is to make a fuss in the redundant information of the carrier file, change or add information to embed the copyright-related information in the file, such as the user's unique identification, copyright signs, etc. secret information. This technology is a method of applying signal processing, which embeds watermark information into digital multimedia under the condition that the human perception system is imperceptible, and only uses a special watermark detector to extract the watermark information. A multimedia security technology . Digital watermarks are mainly divided into two categories: invisible watermarks and visible watermarks. Invisible watermarks can be used for copyright identification, usage tracking, prevention of tampering and illegal copying, broadcast monitoring, etc.; visible watermarks have the function of advertising or restraint, and mainly realize the function of copyright notice.

Reference [6] embeds visible and invisible watermarks in different areas of the screen color image to realize the screen color image embedded with double digital watermarks. In [7], the visible watermark is adaptively embedded in the spatial domain, and the invisible watermark encrypted by the chaotic sequence is embedded in the remaining area of the screen color image to realize the protection of the screen color image embedded with double digital watermarks. Literature [8] proposes an improved digital watermarking algorithm for multimedia courseware protection, researches digital watermarking based on MATLAB, and uses MATLAB and Visual C to implement the digital watermarking system for multimedia courseware protection. Literature [9] proposes an improved digital video watermarking algorithm based on multimedia courseware protection based on the analysis of traditional multimedia courseware copyright protection methods.

In general, the current watermarking algorithms for copyright protection of multimedia teaching courseware generally have the following problems:

(1) The number of bits hidden by the watermark is uncertain. For watermarking algorithms, it is important to know the specific number of hidden watermarks in the carrier image. According to the current research results, there is no reliable upper limit for the number of hidden watermarks in the watermarking algorithm, but for general images, usually only a few hundred bits of information can be hidden.

(2) There are few detection algorithms for copyright protection of multimedia teaching courseware. The copyright protection of algorithmic multimedia teaching courseware has always been a difficult problem for scholars of digital watermarking research. At present, there are very few watermarking algorithms that can completely resist general image attacks, and among them, there are very few adaptive watermarking algorithms. Various geometric attacks, such as rotation, scaling, distortion, etc., are regarded as image processing operations that are difficult to resist.

In a word, nowadays, there are few articles and research contents on watermarking of PPT documents, and there are almost no invisible watermarks in PPT documents. The invisible digital watermarking method for PPT documents in this paper is advanced and practical. Many articles on word document watermarking can be found through paper retrieval. Most of the research methods focus on modifying attributes such as font size and size to embed the watermark information, which has a single form and is not resistant to formatting attacks. However, these methods have given us a lot of inspiration, so that we can expand new methods from different angles, and make reasonable use of the unique components in PPT to embed watermarks, which ensures the security of watermarks and improves the robustness of watermarks.

The rest of this paper is organized as follows. Section 2 introduces some related works. The detailed description of our proposed methods is then presented in Section 3. In Section 4, the experimental results and analysis are reported. Finally, the conclusion is provided in Section 5.

2. **Related Works.**

2.1. **Copyright Notification and Protection of Multimedia Courseware Based on Double Digital Watermark.**

Reference [7] proposes a digital watermarking solution for color images of screen flow in multimedia courseware. The visible watermark is adaptively embedded in the spatial domain, and the invisible watermark encrypted by the chaotic sequence is embedded in the remaining area of the screen color image to realize the protection of the screen color image embedded with double digital watermarks.

2.1.1. *Visible Watermark Embedding Algorithm.* The area in the upper left corner of the screen color image that is the same size as the visible digital watermark is the area where the visible watermark is embedded, called R_v. A visible digital watermark using a grayscale image of size 176×56 will be embedded in the R_v in the upper left corner of the screen's color image. First, the R_v is converted from a color image to a grayscale image, the R_v and the visible digital watermark image are divided into 8×8 pixel blocks, and their respective block variances are calculated separately.

Assuming that the maximum variance of the R_v grayscale image is $v_{\mathtt{max}}$, the minimum variance of the R_v grayscale image is $v_{\mathtt{min}}$, and the variance of the R_v grayscale image block B_{mn} is v_{mn}, the normalized variance α_{mn} of the R_v grayscale image can be calculated as follows:

$$\alpha_{mn} = \frac{v_{mn} - v_{\min}}{v_{\max} - v_{\min}} \tag{1}$$

Let P_{ij}^{w} be the pixel value of the Rv grayscale image block B_{mn} embedded in the visible digital watermark at point (i, j), P_{ij} be the pixel value of the R_v grayscale image block B_{mn} at point (i, j), and W_{ij} be the corresponding visible digital watermark. The pixel value of the block B_{mn} at point (i, j), the visible digital watermark can be embedded in

the R_v grayscale image as follows:

$$P_{ij}^{w} = \begin{cases} P_{ij}, & W_{ij} = 255 \\ round[P_{ij} + k \times (1 - \alpha_{mn}) \times P_{ij}], & W_{ij} = 0 \end{cases} \tag{2}$$

In the formula, k represents the strength factor of the visible watermark, where k=0.02 to improve the vulnerability of the visible watermark.

2.1.2. *Invisible Watermark Embedding Algorithm.* The visible watermark size is much smaller than the screen stream color image. After embedding the visible watermark, use the remaining area to embed the invisible watermark to avoid crossing the visible watermark. The remaining area is called R_{iv}. First, convert the R_{iv} color image to the R_{iv} grayscale image as follows:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.148 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3}$$

The R_{iv} grayscale image is then split into 8×8 pixel blocks. Then calculate the average gray value of each block, and perform discrete cosine transform on each block separately. All blocks are divided into three groups according to the average gray value and DCT coefficients of each block: blocks with low luminance and sample texture, blocks with high luminance and complex texture, and the remaining blocks. Using different embedding strength factors according to the HVS characteristics, the visual concealment is better.

Let g_{mn} be the average gray value of the block B_{mn} of the R_{iv} grayscale image, δ_{mn} be the DCT coefficient of the block B_{mn} of the R_{iv} grayscale image, and $p(i, j)$ be the pixel value of the (i, j) point of the B_{mn} block of the R_{iv} grayscale image :

$$\delta_{mn} = \frac{1}{8 \times 8} \sum_{(i,j) \in B_{mn}} [P(i, j) - g_{mn}]^2 \tag{4}$$

Assuming that T_1, T_2, T_3, T_4 are 4 thresholds, all blocks are divided into three groups according to the following rules: if $g_{mn} < T_1$ and $\delta_{mn} < T_3$, then B_{mn} belongs to the first group; if $g_{mn} > T_2$ and $\delta_{mn} > T_4$, then B_{mn} belongs to the second group; the remaining blocks belong to the third group.

Since the human visual system (HVS) is sensitive to the color image on the screen, if the coefficients in the low-frequency region change, the coefficients in the high-frequency region are easily destroyed. Therefore, after rearranging the DCT coefficients with the Zigzag method, the invisible watermark sequence is used to modulate the coefficients in the intermediate-frequency region [10]. There are also many other schemes [11-14] can be adopted as our image watermarking method.

The specific steps are: perform two-dimensional discrete cosine transform on n×n blocks, rearrange the DCT coefficients with Zigzag, and then obtain a one-dimensional vector $Y = \{y_1, y_2, \ldots, y_{n \times n}\}$. The IF coefficients from l+1 to l+m in the one-dimensional vector of Y are also a one-dimensional vector $Y_l = \{y_{l+1}, y_{l+2}, \ldots, y_{l+m}\}$. The invisible digital watermark sequence $W = \{w_1, w_2, \ldots, w_m\}$ is a sequence of random real numbers conforming to the normal distribution. Modify the intermediate frequency region coefficients from $l + 1$ to $l + m$ in the one-dimensional vector Y with the invisible watermark sequence W as follows:

$$y_i' = \begin{cases} y_i, & i \leq l \texttt{or} i > l + w \\ y_i(1 + \beta w_i), & l < i \leq l + w \end{cases} \tag{5}$$

where β is the embedding strength, $\beta = 0.005$ for the first group, $\beta = 0.0075$ for the second group, and $\beta = 0.01$ for the third group. $Y' = \{y'_1, y'_2, \ldots, y'_{n\times n}\}$ is the changed DCT coefficient sequence reversely rearranged by the Zigzag method.

Two-dimensional inverse discrete cosine transform is performed on $n\times n$ blocks to obtain R_{iv} grayscale images with invisible digital watermarks.Take the U and V component values converted from the screen color image as chromaticity values, take the grayscale values of the screen grayscale images embedded with visible and invisible watermarks as the Y component values, and take the screen grayscale images embedded with visible and invisible watermarks convert to screen color image with embedded double digital watermark. In order to reduce the influence and distortion of the screen image quality caused by the conversion between color images and grayscale images, the Y component value should be rounded up and cannot be truncated. When converting from a screen grayscale image to a screen color image, the value should change to zero if the values of the R, G, and B components are all less than zero, and to 255 if the value exceeds 255. The DCT coefficients corresponding to the changed pixels can be adjusted. It has little effect on visible watermarks. It can be seen that the watermark is not sensitive to the adjustment of pixel values.

2.1.3. *Invisible watermark extraction.* Get the remaining area from the point (200,200) in the screen color image, this is the area where the invisible watermark is embedded, called Riv. Convert R_{iv} color image to R_{iv} grayscale image, then split R_{iv} grayscale image into 8×8 pixel blocks. Discrete cosine transform is performed on each block separately. Rearrange the DCT coefficients with Zigzag to get a one-dimensional vector $Y' = \{y'_1, y'_2, \ldots, y'_{n\times n}\}$. The intermediate frequency coefficients from $l+1$ to $l+m$ in the one-dimensional vector of Y' are also one-dimensional vectors $Y'_l = \{y'_{l+1}, y'_{l+2}, \ldots, y'_{l+m}\}$. The invisible watermark sequence $W' = \{w'_1, w'_2, \ldots, w'_m\}$ that needs to be checked can be calculated by formula (4). The original invisible watermark sequence $W = \{w_1, w_2, \ldots, w_m\}$ is a random real sequence conforming to the normal distribution. The normalized correlation coefficient can be calculated as follows:

$$z = \mathtt{cov}(W, W') = \frac{\sum_{i=0}^{m-1} P_i \times P'_i}{\sqrt{\sum_{i=0}^{m-1} P_i^2 \times \sum_{i=0}^{m-1} P_i'^2}} \tag{6}$$

In the above equation, the factors P_i and P'_i are defined as follows:

$$P_i = w_i - \frac{1}{m}\sum_{j=0}^{m-1} w_j, 0 \le i \le m-1 \tag{7}$$

$$P'_i = w'_i - \frac{1}{m}\sum_{j=0}^{m-1} w'_j, 0 \le i \le m-1 \tag{8}$$

The normalized correlation coefficient z is compared with the specified threshold N_c, if $z \ge N_c$, the screen color image has an invisible digital watermark embedded, otherwise there is no invisible digital watermark embedded. In the text, $N_c = 0.5$.

2.2. **Multimedia Courseware Copyright Protection Algorithm Based on Modified I Frame DCT Coefficients.** Reference [9] proposes an improved video watermarking algorithm based on DCT coefficients for copyright protection of multimedia courseware according to the current method for copyright protection of video streams. This watermarking method for changing DCT coefficients is a watermarking method based on MPEG4 compression, which modifies the DCT coefficients of I-frames in the process

of compressing the video stream. It completes watermark embedding between variable-length decoding and re-variable-length encoding, and the watermark extraction process occurs after variable-length decoding of the input compressed video. The I frame is the reference frame of the B frame and the P frame. If the DCT coefficient of the I frame is directly modified, it may cause the accumulation of distortion and make the image distortion appear from the B frame and the P frame. Therefore, the watermarking algorithm based on DCT coefficients can reduce distortion by adopting motion compensation measures, thereby ensuring video quality. The watermark embedding process of this algorithm is shown in Fig. 1.

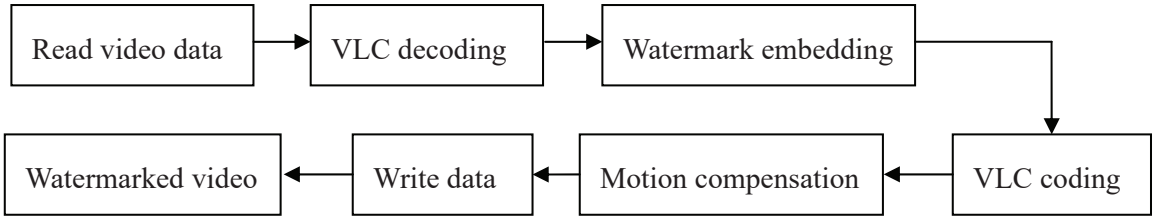


FIGURE 1. Watermark embedding process

2.2.1. *Embedding Point Selection.* According to the Video Compression Coding Standard (MPEG4), the final decoded DC coefficients of the inner macroblocks are derived from the difference of the differential decoding score and the added value of the predictor to recover the encoded data, and the DCT coefficient-based watermarking algorithm selects the differential score as the embedding point of the watermarking algorithm. In order to prevent the obvious distortion of the embedded watermark in the monotonous area of the image, the algorithm skips the DC coefficient according to the variance of the color block. Let each I frame have N chrominance matrices U_k of size 16×16, $k = 0, 1, ..., N-1$:

$$U_k = \begin{bmatrix} u^k_{0,0} & u^k_{0,1} & \dots & u^k_{0,15} \\ u^k_{1,0} & u^k_{1,1} & \dots & u^k_{1,15} \\ \dots & \dots & \dots & \dots \\ u^k_{15,0} & u^k_{15,1} & \dots & u^k_{15,15} \end{bmatrix} \tag{9}$$

The mean and variance of these matrices are calculated as follows:

$$m_k = E(u) = \frac{1}{16 \times 16} \sum_{i=0}^{15} \sum_{j=0}^{15} u^k_{ij} \tag{10}$$

$$\sigma_k^2 = \sigma^2(u) = E[(u - m_k) \times (u - m_k)] = \frac{1}{16 \times 16} \sum_{i=0}^{15} \sum_{j=0}^{15} [u^k_{ij} - \frac{1}{16 \times 16} \sum_{i=0}^{15} \sum_{j=0}^{15} u^k_{ij}]^2 \tag{11}$$

Then the overall variance is calculated as follows:

$$\sigma^2 = \frac{1}{N} \sum_{k=0}^{N-1} \sigma_k^2 \tag{12}$$

If $\sigma_k^2 > \sigma^2$, the chroma block DC coefficients are used as the watermark embedding point, otherwise it will be ignored. After such processing, the video effect will be improved.

2.2.2. *Watermark Embedding Algorithm.* Decode the i-th I frame chrominance DC coefficient (VLC variable length code) in the video stream, obtain the original DC coefficient c_i through inverse quantization, remove the DC coefficient c_i with a fixed quantization factor Q_{DC} to obtain the coefficient c_i', and then modify c_i' to get c_i'' according to certain rules, and complete the embedding process after re-encoding. Assuming that the $i-$th bit of the input watermark is w_i, and D represents the strength of the watermark, the embedding process is as follows:

m =mod(c_i', D)
If($w_i == 1$)
    If($m < D/4$)
        c_i''=c_i'-m-$D/4$
    else
        c_i''=c_i'-m+3D/4
else
    If($m > 3D/4$)
        c_i''=c_i'-m+$5D/4$
    else
        c_i''=c_i'-m+$D/4$
End if

2.2.3. *Watermark Extraction Algorithm.* For the extraction step, first decode the I frame chrominance DC coefficient (VLC variable length code) of the video, obtain the DC coefficient c_i through inverse quantization, divide it by the fixed quantization factor Q_{DC} to obtain the coefficient c_i', and then restore the watermark according to the following rules:

m =mod(c_i', D)
If($0 \leq m < D/2$)
    w_i=0
else
    w_i=1
End if

In general, the watermarking algorithm based on DCT coefficients is resistant to format conversion (AVI-MPEG) processing, and the existing watermarking module can be directly embedded into the video decoder. Taking advantage of the speed of the decoder can improve the efficiency of watermark embedding and extraction. efficiency.

3. **Proposed Algorithm.** The two existing typical algorithms introduced above do not consider the watermark embedding from the PPT file itself, but embed the watermark in each frame of image or video stream of the courseware from the perspective of the video stream formed by the multimedia courseware. There are few researches on the watermarking algorithm of the PPT file itself. This paper designs a variety of watermarking methods according to the characteristics of the PPT file, and achieves a variety of different purposes at the same time.

This paper proposes a multilevel multipurpose framework for PPT to simultaneously achieve different purposes. Based on the proposed framework, we can provide protection and tracking purposes based on four-level independent watermarks embedded in different channels (i.e., different embedding domains) of PPT. Figure 4-6 shows an example interface of the multi-level multi-functional framework of PPT. The interface consists of four parts: the original PPT module, the original watermark module, the watermarked PPT module, and the extracted watermark module. The Browse button is used to select a file,

and the Save As button is used to save the file. The "Information" edit control is used to display relevant information, such as the file path to open the PPT, or to extract the NC value of the watermark. The "Original PPT" control is used to display the open original ppt file or a ppt file that may have a watermark. The Raw Watermark control displays different watermarks, namely Robust Watermark, Extra Information, and Tertiary Fingerprint. An advantage of our framework is that we can achieve multi-level multi-channel embedding, i.e. watermarks at different levels are independent as they are embedded by exploiting different properties (i.e. based on different channels). 'Embedding Mode' or 'Extracting Mode' has four modes, namely CP, FP1, FP2, FP3. "CP" means a copyright watermark (which may contain "additional information"), "FP1" means a level 1 fingerprint, "FP2" means a level 2 fingerprint, and "FP3" means a level 3 fingerprint. If we press the "Embed" or "Extract" button, a dialog box will appear to determine the embedding or extraction parameters. The "Watermarked PP" control is used to display open watermarked or suspicious PPT files. In our framework, we designed three types of

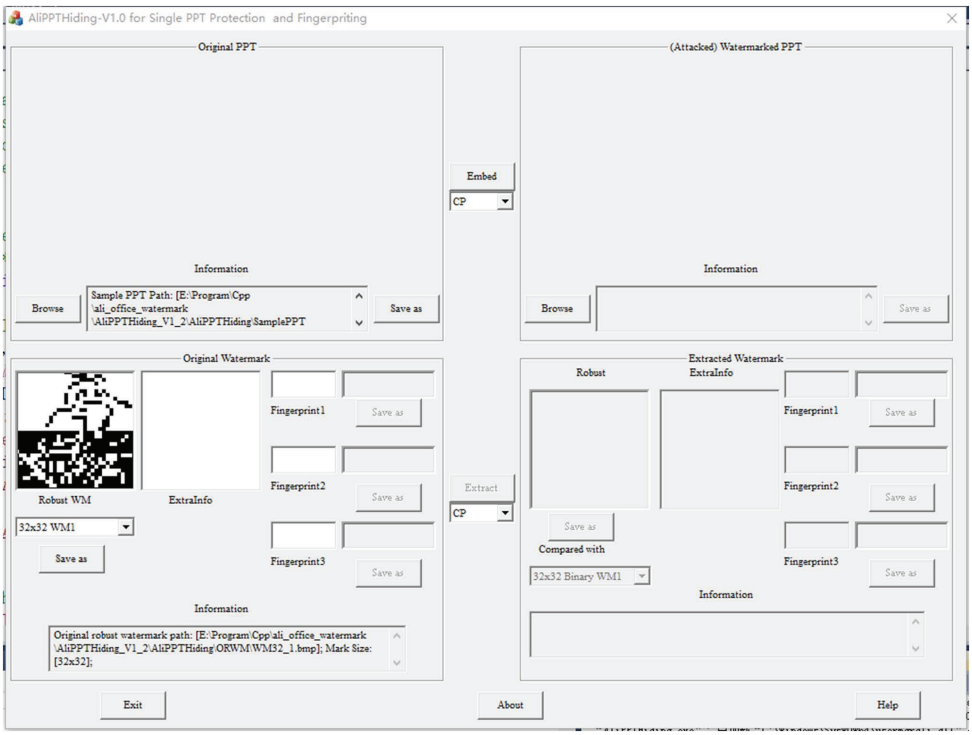


FIGURE 2. Multilevel multipurpose PPT watermark framework

watermarking schemes for PPT, which are: (1) image-based multi-level multi-functional PPT watermarking algorithm; (2) multi-level multi-functional PPT watermarking algorithm based on invisible rectangular frame; (3) Font-based PPT watermarking algorithm. For the first and second schemes, a multifunctional independent watermark can be embedded. For the last scheme, only one layer of watermark may be embedded due to the limited word count.

3.1. **Image-based Multilevel Multipurpose PPT Watermarking Algorithm.** When we make PPT, we usually add some pictures on the PPT. If it is just a text message, most listeners will find the PPT boring and reluctant to watch or listen. If you add pictures to the PPT, the audience will not have visual fatigue, and the audience will continue to be interested in listening. Here, we propose an image-based PPT watermarking method as follows:

3.1.1. *Embedding Process.* The embedding process can be described as follows:

Step1: Save the original ppt (pptx) document in pptx (the purpose is to extract pictures using zip tool) format to the "Results" directory, and take the saved file name according to the embedded method as temp_**.pptx (for CP method, * * means CP; for FP1 mode, ** means FP1; for FP2 mode, ** means FP2; for FP3 mode, ** means FP3);

Step2: Rename temp_**.pptx to temp_**.zip, call the Unzip class library to decompress temp_**.zip, and generate a series of files. The files in the "ppt\media" directory contain various multimedia data appearing in ppt, including images, videos and audios (** means embedded mode, namely CP, ** means CP; for FP1, ** for FP1; for FP2, ** for FP2; for FP3, ** for FP3);

Step3: Obtain the watermark data, take 0/1 information (image or image + data) to be embedded in turn, and get a 0/1 sequence;

Step4: Generate a chaotic sequence according to the input password, and scramble the original 0/1 sequence according to the order of the generated chaotic sequence;

Step5: Add the header bit (twenty 0s) to the front of the scrambling sequence;

Step6: Traverse the "ppt\media" folder in the "Results" directory, filter out all suitable image carriers according to the size of the embedded information, and the number of selected images does not exceed 40 (adjustable, macro definition);

Step7: If there is no suitable image carrier, exit, otherwise continue;

Step8: Embed the scrambled sequence into all suitable images based on dither modulation, and can automatically select 1 or 3 embeddings according to the size of the carrier (redundant embedding to enhance robustness);

Step9: Finish embedding.

3.1.2. *Extraction Process.* The extraction process can be described as follows:

Step1: Save the ppt (pptx) document to be extracted watermark in pptx (the purpose is to use zip tool to extract pictures) format to the "Extraction" directory, and take the saved file name as temp.pptx;

Step2: Rename temp.pptx to temp.zip, call the Unzip class library to decompress temp.zip, and generate a series of files. The files in the "ppt$n$media" directory contain various multimedia data appearing in ppt, including images, videos and audios;

Step3: Traverse the pictures in the "ppt$n$media" folder under the "Extraction" directory, and filter out all suitable picture carriers according to the size of the embedded information;

Step4: Pre-read the image carrier to obtain the first 20 embedded information, which is used to determine whether the image is embedded with a watermark (if the number of 1 is less than 3, the image is embedded with a watermark). If the image meets the conditions, extract all information from the image;

Step5: Take the average value of each bit for all extracted information (because the watermark may be embedded multiple times), if the average value of a bit is greater than 0.5, the bit is considered to be 1, otherwise it is 0;

Step6: Obtain the 0/1 sequence, perform inverse scrambling according to the chaotic sequence generated by the password, and obtain the extracted 0/1 sequence;

Step7: Convert the extracted sequence 0/1 to original information (image or image + data);

Step8: Complete the extraction.

3.1.3. *Dither Modulation Algorithm.* Dither modulation algorithm is a well-known image watermarking scheme. Therefore, we use it in the algorithm. The process can be described as follows:

(1) Embedding process

Step1: After getting the jpg image, convert it to an RGB two-dimensional array;

Step2: Convert the RGB array to a YCbCr array, and select the Y channel for embedding;

Step3: Perform DCT transformation on the Y channel matrix to obtain a coefficient matrix, divide the coefficient matrix into 8×8 blocks, and embed the information of each block;

Step4: Select several bits in the block to embed according to the parameter selection. In this algorithm, each 8×8 block can embed a total of 8 bits of information (the number of embedded bits is optional, but currently 8 bits are preferred), where each 2 bits are assigned to different modes (0 means CP , 1 for FP1, 2 for FP2, 3 for FP3);

Step5: Invert the coefficient matrix of the embedded content through DCT transformation to obtain the Y channel matrix, and then restore it to an RGB array through calculation, and then store it as a jpg file through the storage function to complete the embedding.

(2)Extraction process

Step1: After obtaining the jpg image, restore it to a two-dimensional RGB array;

Step2: Convert the RGB array to a YCbCr array, and select the Y channel for extraction;

Step3: Perform DCT transformation on the Y channel matrix to obtain a coefficient matrix,and the coefficient matrix is divided into 8×8 blocks, and information is extracted for each block;

Step4: Select a few bits in the block to extract according to the parameter selection, where each 2 bits are assigned to different modes (0 for CP, 1 for FP1, 2 for FP2, 3 for FP3), finish the extraction.

3.2. **Invisible Rectangle Frame Based Multilevel Multipurpose PPT Watermarking Algorithm.** The rectangular frame is a commonly used object in Office documents. We can add a rectangular frame to each page of the PPT file. If the data to be embedded can be encrypted as the text of the rectangular frame and put into the rectangular frame, and the rectangular frame is set to be invisible and the size is very small, it will not be easily selected by an attacker and removed.

3.2.1. *Embedding Process.* The embedding process can be described as follows:

Step1: Save the original ppt(pptx) file to the "Results" directory, and take the saved file name according to the embedding method as temp_ ∗ ∗.ppt(pptx) (for CP mode, ** means CP; for FP1 mode, ** Indicates FP1; for FP2 mode, ** indicates FP2; for FP3 mode, ** indicates FP3);

Step2: Open temp_ ∗ ∗.ppt(pptx) with a dedicated class library function, identify the page number of temp_∗∗.ppt(pptx) (** means embedded mode, that is, for CP, ** means CP; for FP1, ** for FP1; for FP2, ** for FP2; for FP3, ** for FP3);

Step3: Obtain the watermark data, take the 0/1 information (image or image + data) to be embedded in turn, and obtain the 0/1 sequence;

Step4: Generate a chaotic sequence according to the input password, and scramble the original 0/1 sequence according to the order of the generated chaotic sequence;

Step5: Convert the scrambled 0/1 sequence into a string in hexadecimal as the text to be hidden;

Step 6: Traverse each page of the PPT and perform the following steps:

Step 6.1: For the current page, add a rectangular box with a small length and width (for example, the length and width is 0.1) near the upper left corner (CP, FP1, FP2, and FP3 have their respective upper left corner positions);

Step 6.2: The rectangular box is named invisibleInfo_#mode (#mode is 0, 1, 2, 3, corresponding to CP, FP1, FP2, FP3 respectively);

Step 6.3: Write the string to be hidden into the rectangle;

Step 6.4: Set the rectangular box to be invisible, and set the lines of the box and the text in the box to be transparent;

Step7: Save and close the PPT file to complete the embedding.

3.2.2. *Extraction process.* The extraction process can be described as follows:

Step1: Use the dedicated class library function to open the ppt (pptx) file to be extracted, traverse each page to find a rectangular box named in the form of (invisibleInfo_#mode) (# mode is 0, 1, 2, 3, corresponding to CP, FP1, FP2, FP3);

Step2: If it is not found, exit, indicating that there is no embedded information in the ppt (pptx) file;

Step3: If you find a rectangular box named in the form of (invisibleInfo_#mode), directly extract the string in the box;

Step4: Convert the string to a 0/1 sequence;

Step5: For the obtained 0/1 sequence, perform inverse scrambling according to the chaotic sequence generated by the password to obtain the extracted 0/1 sequence;

Step6: Convert the extracted sequence 0/1 into original information (image or image + data);

Step7: Complete the extraction and close the ppt (pptx) file.

3.3. **PPT Watermark Algorithm Based on Font Attributes.** If there is a lot of text in the PPT, information can be embedded by modifying the size of the font in the text box, extraction is performed by distinguishing whether the font size is an integer; information can be embedded by modifying the "AutoRotateNumbers" of the font (0 or -1), extraction is performed by distinguishing whether the "AutoRotateNumbers" of the font is zero or not; or by adjusting the font effect (such as embossing) for information embedding, extraction is performed by distinguishing whether there is this effect or not.

3.3.1. *Embedding Process.* The embedding process can be illustrated as follows:

Step1: Open the ppt (pptx) file to be watermarked with a special class library function, traverse the entire document, and calculate the number of Chinese characters, English characters and punctuation characters contained in it (excluding pictures, tables and other special symbols). Divide the number of characters n by the length L of the binary watermark sequence to be embedded, and then the number of characters d in each group can be obtained (that is, 1 bit information can be embedded every d characters).

For example: Assuming that the number of characters in the ppt(pptx) document is 3000, the size of the binary watermark image is 16*16, and we calculate 3000/(16*16)=11.7 (take 11), then the ppt(pptx) document is divided into groups with each group having 11 characters. Take the first 16*16=256 groups of characters to embed the watermark sequence. As for which character of the 11 characters in each group is used to embed the watermark bit, it can be controlled by parameter1, where parameter1=0 means the first character of each group is used for embedding, parameter1=1 means the second character of each group for embedding, etc.

Step2: Obtain the watermark data, and sequentially obtain the 0/1 information (image or image + data) to be embedded to obtain the 0/1 sequence;

Step3: Generate a chaotic sequence according to the input password, and scramble the original 0/1 sequence according to the order of the generated chaotic sequence;

Step4: According to the value (0 or 1) of each bit of the obtained sequence, different methods can be used to embed the watermark:

(1) Method 1:

If the current bit of the sequence is 0, if the font size of the character to be embedded is an integer, then the font size remains unchanged; if the font size of the character to be embedded is *.5, its font size increases by 0.5 to become an integer.

If the current bit of the sequence is 1, if the font size of the character to be embedded is *.5, then the font size remains unchanged; if the font size of the character to be embedded is an integer, its font size increases by 0.5.

(2) Method 2:

If the current bit of the sequence is 0, if the "AutoRotateNumber" of the character to be embedded is 0, the "AutoRotateNumber" remains unchanged; if the "AutoRotateNumber" of the character to be embedded is -1, the "AutoRotateNumber" is changed to 0.

If the current bit of the sequence is 1, if the "AutoRotateNumber" of the character to be embedded is -1, the "AutoRotateNumber" remains unchanged; if the "AutoRotateNumber" of the character to be embedded is 0, the "AutoRotateNumber" is changed to -1.

(3) Method 3:

If the current bit of the sequence is 0, if the character to be embedded has no embossing effect, the character remains without embossing effect; if the character to be embedded has embossing effect, the character is changed to no embossing effect.

If the current bit of the sequence is 1, if the character to be embedded has an embossed effect, the character remains embossed; if the character to be embedded has no embossed effect, the character is changed to have an embossed effect.

Step5: Save and close the ppt (pptx) file to complete the embedding.

3.3.2. *Extraction Process.* The extraction process can be illustrated as follows:

Step1: Open the ppt (pptx) file to be extracted with a special class library function, traverse the entire document, and calculate the number of Chinese characters, English characters and punctuation characters contained in it (excluding pictures, tables and other special symbols). Divide the number of characters n by the length L of the binary watermark sequence to be extracted (obtained by the key), and then the number of characters d in each group can be obtained. Then according to the parameter1 in the key, confirm from which character in each group of characters to extract the watermark bit;

Step2: Use different methods to extract the value (0 or 1) of each bit of the sequence:

(1) Method 1:

If the font size of the current character is an integer, the current bit of the sequence is 0.

If the font size of the current character is *.5, the current bit of the sequence is 1.

(2) Method 2:

If the "AutoRotateNumber" of the current character is 0, the current bit of the sequence is 0.

If the current character's "AutoRotateNumber" is non-zero, the current bit of the sequence is 1.

(3) Method 3:

If the current character has no embossing effect, the current bit of the sequence is 0.

The current bit of the sequence is 1 if the current character is embossed.

Step3: Obtain the 0/1 sequence, perform inverse scrambling according to the chaotic sequence generated by the password, and obtain the extracted 0/1 sequence;

Step4: Convert the extracted sequence 0/1 into original information (image or image + data);

Step5: Complete the extraction and close the ppt (pptx) file.

4. **Experimental Results.** In this section, we test the effectiveness of our multilevel multipurpose framework for PPT files, where we embed different watermarks independently in different properties of the PPT files. In this framework, we totally presents three methods. In the following experiments, we use the same watermark setting as shown in Fig. 3, where we at most embed four watermarks as given in Fig.4 in a PPT file, where CP (the robust watermark) includes the watermark image (frog) and extra information("OK, Let's start"), FP1 (Level 1 fingerprint) is the text image "12123434",FP2 (Level 2 fingerprint) is the text image "13571357" and FP3 (Level 3 fingerprint) is the text image "33441122". After testing the performance for three methods respectively, we list the overall performance in Table 1, where Method 1 stands for "Method Based on Images", Method 2 stands for "Method Based on Invisible Rectangle Frame", Method 3 stands for "Method Based on Font Attributes". From Table 1, we can see that all of the proposed three methods can resist "Save As" attack, and they can resist some modifications in PPT files. In order to understand the performance of the algorithm in detail, the detailed experiments and performance analysis of Method 1 are given below.

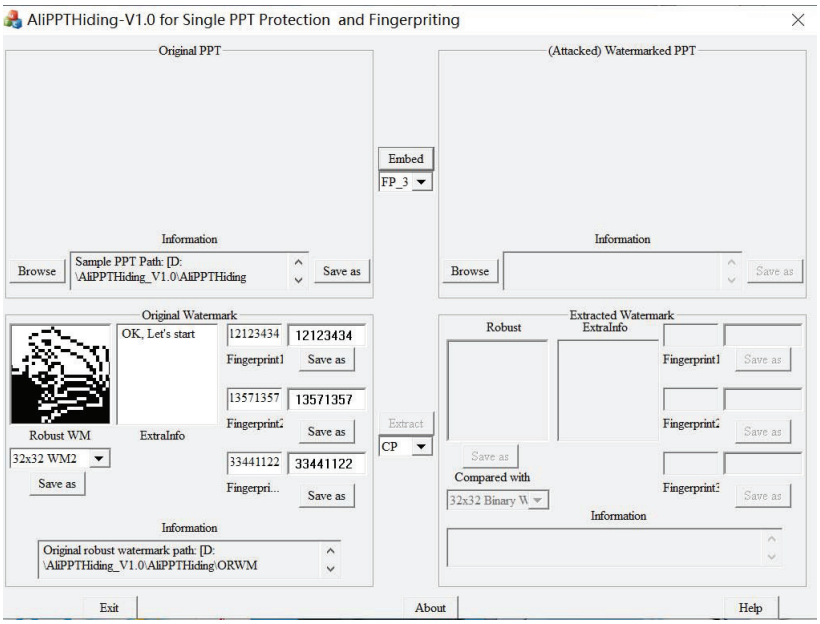


FIGURE 3. The watermark settings of following experiments

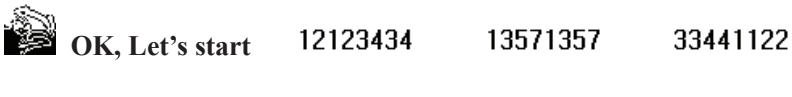


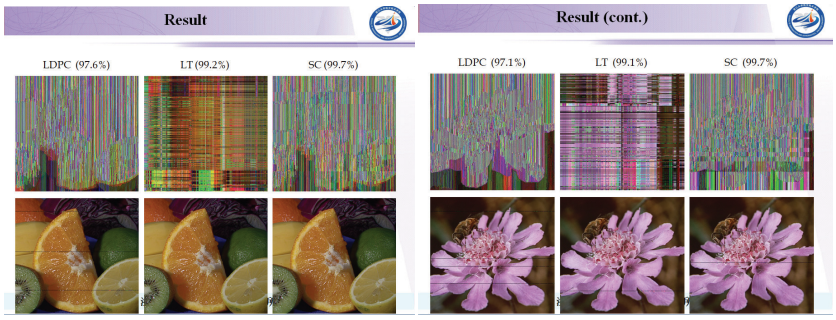
FIGURE 4. The four level watermarks to be embedded

For any PPT file with many images, our first method can embed four level watermarks independently in images containing in the ppt file. The example original ppt slices and the four-level watermarked ppt slices are shown in Fig.5. The average PSNR value between the original pictures in the original ppt file and the watermarked pictures in the

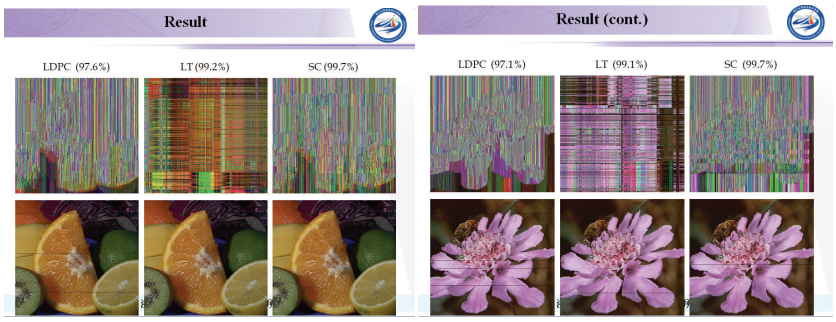
TABLE 1. Test Results of Three PPT Information Hiding Methods

Method	Copy paste	Save As	Format Painter
Method 1	✓	✓	✓
Method 2	✓	✓	✓
Method 3	✓	✓	X

watermarked ppt file is 36.52dB (after one level embedding, 40.21dB; after two level embedding, 38.53dB; after three level embedding, 37.16dB). From Fig. 5, we can see that the invisibility of Method 1 is very good. The extracted watermarks from the watermarked ppt file are shown in Fig. 6. In this paper, we adopt NC to denote the similarity between two marks of the same size, where NC= 1 means two watermarks are identical, while NC= 0 means they are totally different. From Fig. 6, we can see that the four level watermarks (including the extra information "OK, Let's start") can be completely extracted without any error from the watermarked ppt file without attack, i.e., NC=1.0 for each level watermark.



(a) Original two slices



(b) Watermarked two slices

FIGURE 5. The original and four level watermarked ppt slices

In fact, we can extract each level watermark independently even if the pictures in the watermarked ppt file have undergone some image processing operations. Table 2 gives the extraction results in NC values between the extracted watermark and the original watermark of each level after all of the watermarked pictures in the watermarked ppt file has undergone several common image processing operations, including JPEG compression with QF=90, JPEG compression with QF=70, cropping the upper-left corner, scaling with factor 1.2, scaling with factor 0.8, adding Gaussian noises by 1% and median filtering with size 3×3. From Table 2, we can see that our scheme is robust to common image processing operations, and for every level, the NC value is above 0.93. From these results, we can see

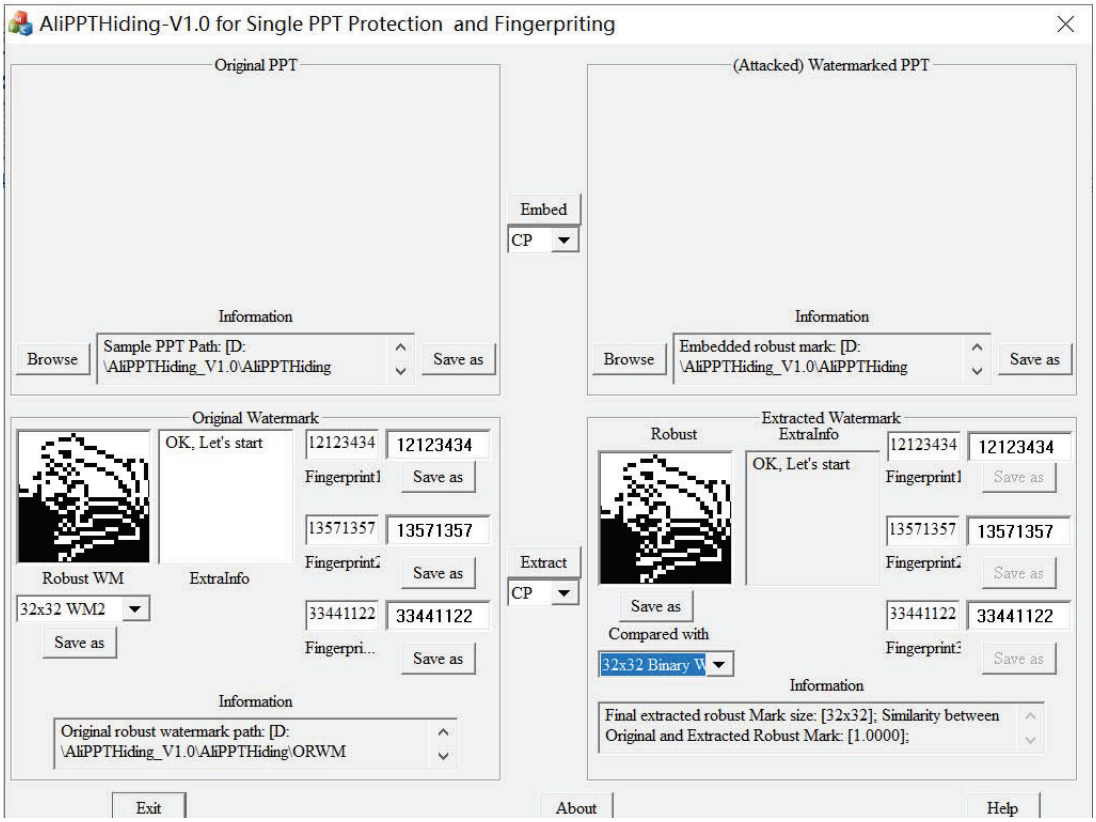


FIGURE 6. The extracted four level watermarks from the watermarked ppt file

that Method 1 is robust to most common image processing operations on the watermarked ppt file.

TABLE 2. NC values of four level watermarks extracted from the watermarked ppt file under different attacks.

Level	CP	FP1	FP2	FP3
JPEG90	1.000	1.000	1.000	1.000
JPEG70	0.994	0.991	0.990	0.985
Cropping25%	0.985	0.981	0.977	0.975
Scaling 1.2	0.962	0.957	0.954	0.951
Scaling 0.8	0.951	0.948	0.945	0.940
Guassian noise 1%	0.954	0.951	0.948	0.944
Median filtering by 3×3	0.944	0.939	0.935	0.931

5. **Conclusions.** This paper proposes a multilevel multipurpose framework for PPT watermarking. Based on this framework, we develop three watermarking schemes that can achieve multiple purposes including copyright protection, traitor tracing, document security management and so on. From experimental results, we can see that our proposed schemes can embed multiple watermarks and can extract all watermarks correctly and independently without attacks or even with some attacks. Future work will concentrate on further improving the performance and security by combining the watermarking scheme with other techniques.

Acknowledgment. This research is supported in part by the National Key Research and Development Program of China under Grant No.2020AAA0140004 and the Public Good Research Project of Science and Technology Program of Zhejiang Province under Grant No. LGG21F020005. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] W. Wan, J. Wang, Y. Zhang, J. Li, H. Yu, and J. Sun, A comprehensive survey on robust image watermarking, *Neurocomputing*, vol. 488, pp. 226-247, March 2022.

[2] N. T. B. Abdulla and K. A. Navas, Robust video watermarking resilient to inadvertent attacks, *2020 International Conference on Power Electronics and Renewable Energy Applications (PEREA)*, pp. 1-5, 2020.

[3] S. Kakikura, H. Kang, and K. Iwamura, Collusion resistant watermarking for deep learning models protection, *2022 24th International Conference on Advanced Communication Technology (ICACT)*, pp. 40-43, 2022.

[4] O. Abodena, Robust and high-capacity audio watermarking basedon chirp z-transform, *2021 29th Signal Processing and Communications Applications Conference (SIU)*, pp. 1-4, 2021.

[5] S. Siledar and S. Tamane, A distortion-free watermarking approach for verifying integrity of relational databases, *2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC)*, pp. 192-195, 2020.

[6] R. Huang, H. Liu, and Y. Chen, Multimedia courseware copyright notification and protection based on DCT, *2007 First IEEE International Symposium on Information Technologies and Applications in Education*, pp. 449-452, 2007.

[7] R. -H. Huang, H. -L. Liu, and J. Dai, Copyright notification and protection of multimedia courseware based on dual digital watermark, *2008 Congress on Image and Signal Processing*, pp. 683-687, 2008.

[8] D. Zhong and C. Chen, The study of digital watermarking system for the protection of multimedia courseware, *2011 IEEE 2nd International Conference on Com-puting, Control and Industrial Engineering*, pp. 304-307, 2011.

[9] D. Zhong, The study of digital video watermarking algorithm based on the protection of multimedia courseware, *Proceedings of 2011 Cross Strait Quad-Regional Radio Science and Wireless Technology Conference*, pp. 1297-1300, 2011.

[10] B.Chen and G.W.Wornell, Digital watermarking and information embedding using dither modulation, *IEEE Second Workshop on Multimedia Signal Processing*, pp. 273-278, 1998.

[11] F.-H. Wang, J.-S. Pan, and L. C. Jain, *Innovations in Digital Watermarking Techniques*, Springer, Berlin-Heidelberg, Germany, 2009

[12] S.-C. Chu, J. F. Roddick, Z.-M. Lu, and J.-S. Pan, A digital image watermarking method based on labeled bisecting clustering algorithm, *IEICE Transactions on Fundamentals of Electronics, Communication and Computer Sciences*, vol. E-87-A, no. 1, pp. 282-285 , 2004

[13] F. Gu, Z.-M. Lu, and J.-S. Pan, Multipurpose image watermarking in DCT domain using subsampling, *IEEE International Symposium on Circuits and Systems*, pp. 4417-4420, 2005

[14] J.-S. Pan, W. Li, C.-S. Yang, and L. Yan, Image steganography based on subsampling and compressive sensing, *Multimedia Tools and Applications*, vol. 74, no. 21, pp. 9191-9205, 2005