Multiuser Watermarking for Secured Social Networking

Komal Toshniwal,Prof. Santosh Chobe. DYPIET, PUNE University

Abstract
Watermarking is a widely used technique to protect the copyright of digital media such as image, text, music and movie. In this study, we are proposing a robust watermarking scheme which can be used by all social networking users to secure their images while sharing it on internet. The proposed scheme makes use of the steganography techniques, transform domain technique, chaos technique, noise reduction technique and error correcting code technique to achieve the desired results. The steganography technique provides an ability to protect all the images or even images which has multiple owners. Rest of the techniques are applied to enhance the robustness of the scheme. The integration with this utility will help many internet users to embed their registered watermarks in their photos before uploading them to common social networking sites like Facebook or Twitter or Flicker to avoid misuse of their uploaded photos. And users will have the ability to prove the morphed images does not belong to them.

Key terms
Stegnography,Permutation,Visual Cryptography

1. Introduction

Faster adoption of the internet and everyday use of the social meia has made sharing of digital media on social networking extremely popular. Since this data can be obtain and shared easily, protecting the copyright protection of digital media is becoming very important to avoid the mis-use. We will focus on a watermarking schemes for digital media and images using steganography in this paper which will help users to protect their images before sharing them on social networking sites.

Watermarking is a technique to protect the copyright of digital media such as image, text, music and movie. For example, if paper bank notes or stock certificates could be easily copied and used, trust in their authenticity would greatly be reduced, resulting in a big loss. To prevent this, currencies and stock certificates contain watermarks. These watermarks are one of the methods for preventing counterfeit and illegal use. Watermarking schemes for digital images suffer a lot of attacks that aim at severing the relationship between the watermarked image and the watermark, such as compression attack, blurring attack, sharpening attack, scaling attack, cropping attack, distortion attack and noise attack.

A watermarking scheme should have ability to combine cover images with a watermark is such a way that it is very difficult to detect and significantly harder to be removed. It should also provide the owner of the image an ability to prove his copyright by extracting his watermark from the secured image. The features like imperceptibility, robustness, security and blindness should help watermarking schemes to become strong. To achieve the desired results we have decided to use the Steganography for encryption. The main advantages of steganography based watermarking like having large embedding capacity, provides high security, and most important it can share secret image between multiple users. In order to enhance the robustness of the steganograhpy based watermarking schemes, the transform domain technique, chaos technique, noise reduction technique and error correcting code technique are applied. For most VC-based watermarking schemes in the literature, each cover image corresponds to a secret image that is registered to a trust authority (TA) (the arbitrator). When the number of cover images is large, it will be a heavy burden for the TA to store all the secret images. To overcome on this challenge we have come up with a scheme where user needs to register only one secret image/watermark image for all the images he owns. This will help us reduce the burden on TA significantly. The major technical requirements for this application are as follows: The watermark does not incur visible (or audible) artifacts to the ordinary users.

- The watermark is not visible to the ordinary users
- The watermark is independent of the image types
- The information carried by the watermark is robust to content manipulations

- The watermark can be detected without the original image
- The watermark can be identified by unique passwords that are used to identify individual user uniquely

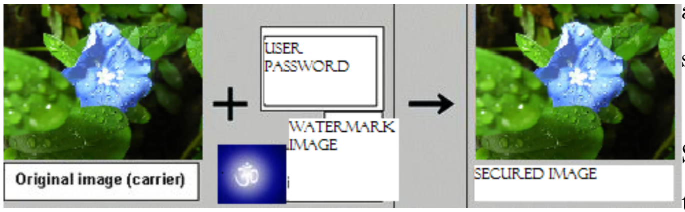


Figure 1: Structure of Digital Watermark

The image that contains a digital watermark is also called a carrier. While sharing the images the digital watermark is not provided as a separate file or a link. The watermark information is directly embedded in the carrier file. Because of this embedding, the digital watermark cannot be identified by looking at the carrier image containing it. Special software processing is required to extract or detect such hidden digital watermarks. Ideally both images and audio data can carry watermarks. A digital watermark can be detected as shown in the following illustration

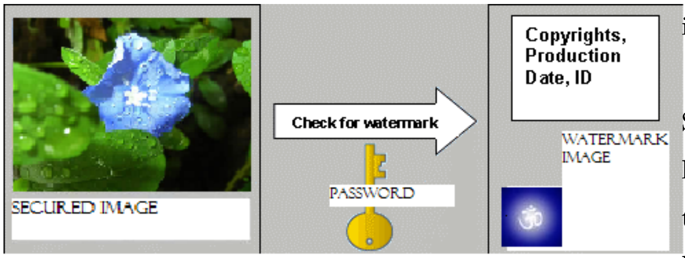


Figure 2: Detecting Digital Watermark

2. Related Work

1. Visual Cryptography

VC is a kind of technique which is commonly used to share secret images, and it helps protect user's copyright for their work. The basic principle of visual cryptography scheme (VCS) was first formally introduced by Naor and Shamir Generally speaking, a (k, n)-VCS takes a secret image as input, and outputs n share images that satisfy two conditions: first, any k out of n shares can recover the secret image; second, any less than k shares cannot get any information about the secret image (unconditionally secure).

The underlying operation of the Visual Cryptography Scheme (VCS) can be OR and XOR (exclusive-OR). In this paper, we make use of the VCS with underlying operation XOR. An XOR-based VCS usually has better performance and results in terms of the visual quality of the recovered secret image. An example of (2, 2)-VCS with underlying operation XOR is shown in below figure which denotes the XOR operation, we have (d) XOR (b) = (c) in figure.

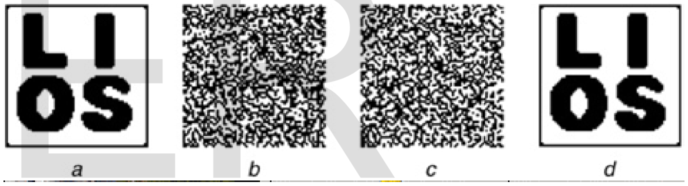


Figure 3: VCS with the underlying operation XOR

(a - Original image, b and c - Shares, d - Recovered image from b and c)

2. Steganography

Steganography is generally called as the art/science of hiding the messages in such a way that no one else except the sender and the recipient, suspects the existence of the hidden message. This is a form of security through obscurity. The advantage of steganography over cryptography is that messages do not attract obvious attention in case of encryption. Directly visible encrypted messages even if they are unbreakable wil always arouse suspicion. So cryptography can help to protects the contents of a message however steganography can be said to protect both messages and the participants.

Steganography includes the hiding of the information within computer files. Using digital steganography in electronic communications we may include steganographic coding inside of a transport layer, such as a document file or image file. Media files are good candidatel for steganographic transmission because of their large size. This can be as simple example as, a sender might start with a simple image file and adjust the color of some regular pixel to correspond to a letter in the alphabet. A change can be so subtle that someone not specifically looking for it is may not notice it at all.

3. Chaos technique

The chaos technique is widely used in the study of watermarking. The most commonly used and simple chaos technique is called torus automorphism.The torus automorphism is used to convert an image into an unrecognized form. The transformation in this technique is defined by following formula. In the formula (xi, yi) and (xi+1, yi+1) are the coordinates of pixels in an image, i is the number of rounds of the torus automorphism, and N * N is the size of the image. The pixel at position (x1, y1) is moved to the position (xi, yi) after i rounds of the torus automorphism. The inversion of torus automorphism exists which helps to extract the image back.

$$\begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ k & k+1 \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} \bmod N$$

Figure 4: Chaos Transformation Formula

The torus automorphism is widely used to change a regular image to a completely chaotic image.This technique is generally applied to square images, and it can be applied on any size images. The genral model of watermarking can be viewed as an information transmission medium, where the cover image is the channel and the watermark is the message. The attack on the watermarking scheme can add error pixels or noise to the watermark. For some attacks like cropping attack, the error pixels may be aggregated. The torus automorphism can help scatter the error pixels uniformly to the entire image. Take the cropping attack as an example where 20 percent of the cover image being cropped. If we apply the torus automorphism process, there is only one error pixel in every 5 pixels on an average. This can enable the possibility of correcting the error pixel by using the information from remaining pixels.

4. Transform Technique - DCT

There are many transformation techniques. However most of these techniques are very slow. This is important aspect to consider as image encoding demands a faster response time for encoding and decoding. The JPEG committee took suggestions and empirically studied the use of several different transforms. Of the transforms studied, DCT (Discrete Cosine Transform) proved superior. In JPEG, DCT operates on one block at a time. Because there are 64 elements in an 8x8 block, this is called the 64-element or 64-coefficient DCT. The DCT transform operates on this block in a left-to- right, top-to-bottom manner. The image below gives the formula for DCT.

$$S_{ij} = \frac{1}{4} C_j C_i \sum_{x=0}^{7} \sum_{y=0}^{7} P_{xy} \cos\left[(2x+1)j\pi/16\right] \cos\left[(2y+1)i\pi/16\right]$$

$$C_i, C_j = 1/\sqrt{2} \text{ when } i, j = 0$$

$$C_i, C_j = 1 \text{ otherwise}$$

Figure 5: DCT Formula

The transformation of a 64 bit DCT results into 1 DC coefficient and 63 AC coefficients. The DC coefficient represents the average color of the 8 * 8 pixel block and the 63 AC coefficients represent color difference across the block. Low-numbered coefficients represent low-frequency color change, or gradual color change across the region. High-numbered coefficients represent high-frequency color change, or color which changes rapidly from one pixel to another within the block. These 64 results are written in a zig-zag order as follows, with the DC coefficient followed by AC coefficients of increasing frequency.

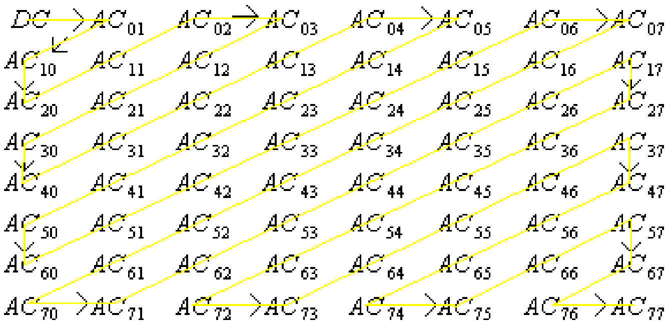

DC AC01 AC02 AC03 AC04 AC05 AC06 AC07
AC10 AC11 AC12 AC13 AC14 AC15 AC16 AC17
AC20 AC21 AC22 AC23 AC24 AC25 AC26 AC27
AC30 AC31 AC32 AC33 AC34 AC35 AC36 AC37
AC40 AC41 AC42 AC43 AC44 AC45 AC46 AC47
AC50 AC51 AC52 AC53 AC54 AC55 AC56 AC57
AC60 AC61 AC62 AC63 AC64 AC65 AC66 AC67
AC70 AC71 AC72 AC73 AC74 AC75 AC76 AC77

Figure 6: ZigZag Sequencing for DCT

Please consider that each diagonal in this zig-zag sequence contains AC coefficients whose sum is constant. For example, the coefficients 30, 21, 12, 03 all add to 3. The ordering of coefficient is really important. If we consider a block of 8x8 pixels out of a coherent image, the pixels are likely to be very similar. If you run DCT on 64 pixels which are very similar, you will get a DC coefficient and some values for the low-frequency AC coefficients; the remaining coefficients will likely be at or near zero. Try to imagine creating an image out of pixels that wildly vary from their neighbors. The resulting image will more than likely not make much sense, it will just be a mess of dots.

5. Noise Reduction technique

Noise reduction helps to remove noise from the signal. In the case of images the noise gets introduced due to the grain structure of the image. Noise can be random or white noise with no coherence, or coherent noise introduced by the device's mechanism or processing or by external attacks. The attacks on the watermarking techniques are just like adding noise to the watermark. Applying the chaos technique in the watermarking scheme helps us to spread the noise uniformly in the extracted image. We can use some noise reduction techniques to improve the qaulity of extracted digital image.

3. Programmer's design

To depict the embedding and extraction algorithm and their results, we have build a web based application. The application gives you a basic capability to browse through the current digital images and users which should be used to encrypt the images. Once you chose the image and users, it picks up the watermarks and embed them in the cover images. These secured cover images are saved in a specific folder. Users can use thse secured images while sharing their content on social networking sites. To enable users to have a look at all encrypted image browsing, we have also implemented a simple web based interface. For users who want to extrac their watermark to confirm the identity, there is another interface where you can chose the attacked or secured image, provide the user password and extract the watermark from the image.

Both of these interfaces in the background call the Embedding and Extraction Algorithms explained below in detail. The algorith implementation and UI layer gives user capability to validate the implementation of our scheme.

3.1. The Algorithm

The proposed watermarking scheme at high levle consists of two main algorithms:

- The embedding algorithm
- The extraction algorithm

We have implemented slight variation of some of the following algorithms as part of these these two algorithms

- DCT Algorithm

- JPEG Encoder Algorithm

The embedding algorithm helps users to encrypte their cover images using the password and watermark registered with Trusted Authority (TA). The algorithm has capability to support multiple file types as well as multiple owners. The extraction algorithm helps users to extract the watermark from the attacked image with secret password. If the image is attacked the extraction algorithm has an ability to detect the attack and warn the user that this is an morphed/attacked image.

1. Embedding Algorithm

- Check if the cover image and watermark images are valid
- Create new output image file
- Apply DCT Algorithm
  - Initialize AANscaleFactor array
  - Initialize the cosine transform matrix
  - Define the quality percentage for luminance and chrominance
  - Create luminance and chrominance matrix
- Apply Huffman encoding Algorithm
  - Define DC Tables and AC Tables
  - Find out the n bit
  - Update the bits based on DC Table and AC Table coefficients
  - Create luminance and chrominance matrix
- Starting from upper left corner of the image, compress a block of 8 by 8 data until the full image is compressed
- Start the DCT Quantization
- Collect all DCT coefficient in an array
- Validate the expected capacity needed for embedding is available or not
- Embed the secret data in permuted sequence
- Generate random number based on password
- Shuffle the embedding of bits based on generated random number
- Define the âĂIJnâĂİ for algorithm
- Add psuedo random number bit string to hide the distribution
- Start embedding the bits
  - Embed status word
  - Skip DC coefficients
  - Skip zero bits
  - Decrease absolute values
  - Move to next bit, if byte of embedded text is empty get a new one
  - repeat the steps 2 to 6 untill all the bytes are embedded
- Save the encrypted file to a common shared directory

2. Extraction Algorithm

- Check if input image is valid and take the decryption password too
- Initialize Huffman Decode related tables
  - Parse marker and header information
  - Read Huffman table
  - Read Quantization table
- Run Huffman decode algorithm
  - Calculate the number of encoded blocks
  - Process byte by byte
  - Get AC coefficients

  - Decode non zero AC coefficients
- Get the random number based on the password
- Recreate shuffle sequences based on random number
- Based on shuffled index start extraction byte by byte
  - Skip DC Coefficients and Zeros
  - Extract bits

3.2. Data independence and Data Flow architecture

The Process Flow for the above algorithms is as shown below

3.3. Multiple users Logic

The proposed algorithm also has a capability to support the multiple user ownership for single digital content. The algorithm can embed the joint watermark using their individual passwords. To decrypt the watermark image, it is required to provide both the passwords. If only one password is provided the watermark image can not be extracted. This capability is really important considering the fact that more and more digital content is work of more than one person these days.

3.4. Turing Machine

The state transition diagram for embedding algorithm is as follows

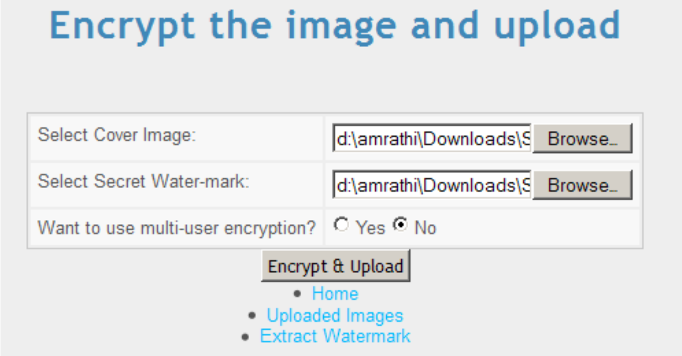
The state transition diagram for extraction algorithm is as follows

4. Results and Discussion

As part of the simulation testing we have ensured that algorithm is meeting the desired criteria of imperceptibility, security, blindness and fragility. Following are some simulation tests that we have performed while validating this algorithm.

- Validate encryption/decryption algorithm supports multiuser inputs
- Validate encryption/decryption algorithm supports multiple file types
- Validate that there is no visual difference between the cover image and encrypted image
- Validate that a valid user is able to decrypt the cover image to prove the ownership
- Validate that an if the image is attacked
- Validate that a watermark is not getting embedded in only one section of image
- Validate that a ownership can be proven even if the image has multiple owners
- Validate that users with wrong passwords can not extract the watermark images

These tests helped us to ensure that we are serving the basic needs of the social networking users while sharing their images on the internet.



Encrypt the image and upload

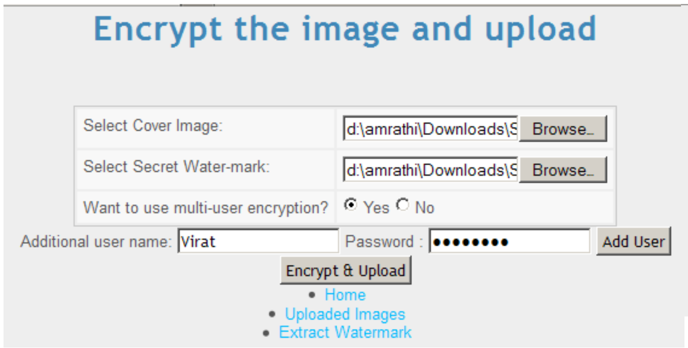
Select Cover Image: d:\amrathi\Downloads\S

Select Secret Water-mark: d:\amrathi\Downloads\S

Want to use multi-user encryption? ○ Yes ● No

Figure 7: Embedding the watermark image - Single User

As part of future work even same algorithm can be implemented on the camera devices or scanner devices so that the basic copyright information is embedded at the time of creation of digital content itself. This will help us

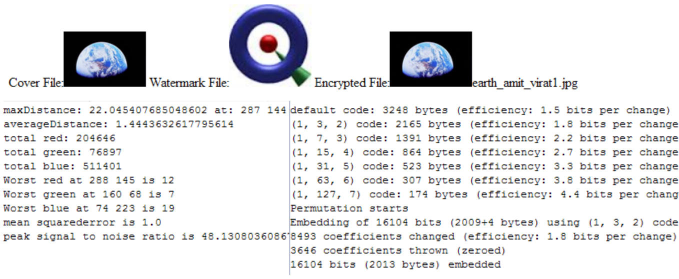


Encrypt the image and upload

Select Cover Image:	d:\amrathi\Downloads\S Browse...
Select Secret Water-mark:	d:\amrathi\Downloads\S Browse...
Want to use multi-user encryption?	● Yes ○ No

Additional user name: Virat Password : •••••••• Add User

Figure 8: Embedding the watermark image - Multi User



```
maxDistance: 22.045407685048602 at: 287 144
averageDistance: 1.4443632617795614
total red: 204646
total green: 76897
total blue: 511401
Worst red at 288 145 is 12
Worst green at 160 68 is 7
Worst blue at 74 223 is 19
mean squarederror is 1.0
peak signal to noise ratio is 48.1308036086
```

```
default code: 3248 bytes (efficiency: 1.5 bits per change)
(1, 3, 2) code: 2165 bytes (efficiency: 1.8 bits per change
(1, 7, 3) code: 1391 bytes (efficiency: 2.2 bits per change
(1, 15, 4) code: 864 bytes (efficiency: 2.7 bits per change
(1, 31, 5) code: 523 bytes (efficiency: 3.3 bits per change
(1, 63, 6) code: 307 bytes (efficiency: 3.8 bits per change
(1, 127, 7) code: 174 bytes (efficiency: 4.4 bits per chang
Permutation starts
Embedding of 16104 bits (2009+4 bytes) using (1, 3, 2) code
8493 coefficients changed (efficiency: 1.8 bits per change)
3646 coefficients thrown (zeroed)
16104 bits (2013 bytes) embedded
```

Figure 9: Embedding the watermark image - Multi User - PSNR

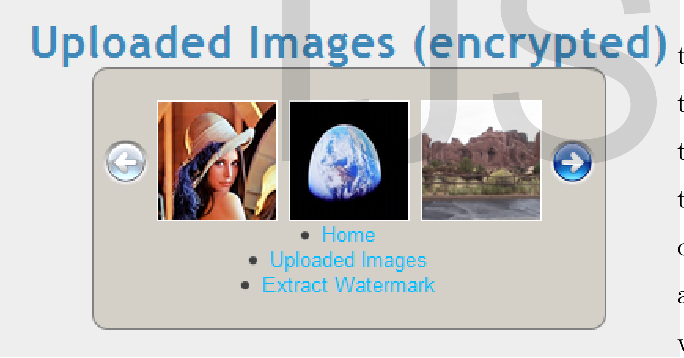
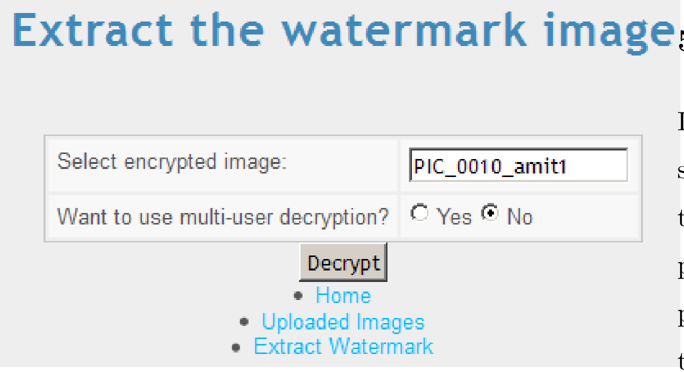


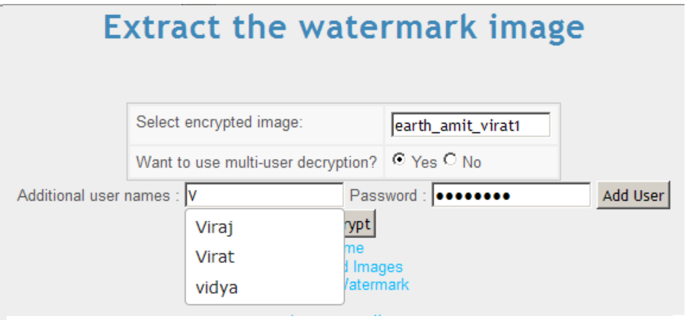
Figure 10: Uploaded images



Extract the watermark image

Select encrypted image:	PIC_0010_amit1
Want to use multi-user decryption?	○ Yes ● No

Figure 11: Extracting the watermark image - Single User

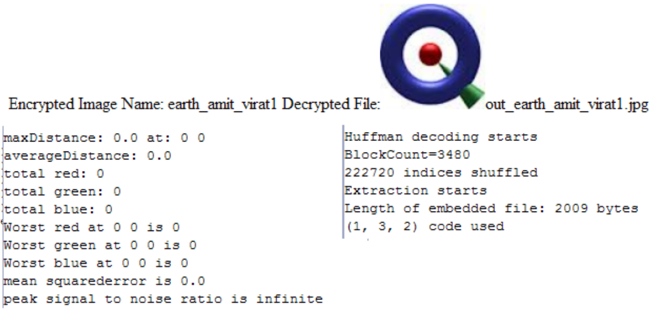


Extract the watermark image

Select encrypted image:	earth_amit_virat1
Want to use multi-user decryption?	● Yes ○ No

Additional user names : V Password : •••••••• Add User

Figure 12: Extracting the watermark image - Multi User



Encrypted Image Name: earth_amit_virat1 Decrypted File:
out_earth_amit_virat1.jpg

```
maxDistance: 0.0 at: 0 0
averageDistance: 0.0
total red: 0
total green: 0
total blue: 0
Worst red at 0 0 is 0
Worst green at 0 0 is 0
Worst blue at 0 0 is 0
mean squarederror is 0.0
peak signal to noise ratio is infinite
```

```
Huffman decoding starts
BlockCount=3480
222720 indices shuffled
Extraction starts
Length of embedded file: 2009 bytes
(1, 3, 2) code used
```

Figure 13: Extracting the watermark image - Multi User - PSNR

to take security and ownership to the digital content to the next level. The current algorithm is more centered towards detecting the attack or helping users to claim that this is an attacked image. We can also enhance the decryption algorithm little further to give the percent attack output on attacked images compared to original watermark.

5. Conclusion

In this paper, we proposed a VC-based watermarking scheme which can be used by the social networking users to protect their images while sharing it on internet. The proposed scheme can deal with images which has multiple owners too and also has reduced the burden on TA to save secret share per image. The implementation has shown strong robustness, perfect imperceptibility and has also satisfied the blindness and security properties. We made some simulations on the robustness, and gave a quantitative comparison on robustness and a qualitative comparison on effectiveness between our scheme and some known VC-based watermarking schemes. The integration of this algorithm with social networking sites like facebook and twitter can greatly increase the confidence in regular users while sharing their images and other digital content in the internet.

References

[1] [1] Chang, C.C., Chuang, J.C.: âĂŸAn image intellectual property protection scheme for gray-level images using visual secret sharing strategyâĂŹ, Pattern Recognit. Lett., 2002, 23, pp. 931âĂŞ941

[2] Wang, F.H., Yen, K.K., Jain, L.C., Pan, J.S.: âĂŸMultiuser-based shadow watermark extraction systemâĂŹ, Inf. Sci., 2007, 177, pp. 2522âĂŞ2532

[3] Lou, D.C., Tso, H.K., Liu, J.L.: âĂŸA copyright protection scheme for digital images using visual cryptography techniqueâĂŹ, omput. Standard Interfaces, 2007

[4] Hsieh, S.L., Hsu, L.Y., Tsai, I.J.: âĂŸA copyright protection scheme for color images using secret sharing and wavelet transformationâĂŹ. Proc. World Academy of Science, Engineering and Technology, 2005

[5] Wang, M.S., Chen, W.C.: âĂŸDigital image copyright protection scheme based on visual cryptography and singular value decompositionâĂŹ, Opt.Eng., 2007

[6] Hsu, C.T., Wu, J.L.:âĂŸHidden digital watermarks in imagesâĂŹ, IEEE Trans. Image Process., 1999

[7] Voyatzis, G., Pitas, I.: âĂŸApplications of toral automorphisms in image watermarkingâĂŹ. Proc. Int. Conf. on Image Processing, 1996, vol. 2, pp. 237âĂŞ240