Digital Watermarking on Micro Image using Reversible Optical Reproduction

Shraddha Raut, Antra Bhattacharya & S. P. Khandait

G.H.R.I.E.T.W , G.H.R.I.E.T.W, K.D.K.C.E
E-mail : rautshraddha@yahoo.in, antarab@rediffmail.com, prapti_khandait@yahoo.com

***Abstract* – Digital watermarking is becoming increasingly important in a large number of applications such as copyright protection, content authentication, and document annotation. Reversible image watermarking, a kind of digital watermarking, is favored in fields sensitive to image quality like military and medical imaging. This paper studies the problem of achieving reversible visible image watermarking. I propose a lossless visible watermarking scheme that adaptively varies the watermark strength to be embedded in different areas of the host image, depending on the underlying image content and human visual system characteristics.**

Index Terms— Digital watermarking, semi fragile, reversible visible, recovery packet.

I. INTRODUCTION

Owing to the popularity of the Internet and the rapid growth of multimedia technology, users have more and more chances to use multimedia data. Consequently, the protection of the intellectual property rights of digital media has become an urgent issue.

Visible watermarking is the study of techniques that insert copyright information perceptibly into the contents of cover digital multimedia so as to identify the ownership in a displayable manner and to prevent the viewers from making unauthorized use. In most conventional visible watermarking schemes [1]–[2], a visible watermark is usually designed to be irremovable in order to effectively resist unintended editing and malicious attacks [3]–[4]. However, in some potential applications, a visible watermark is required to be removable.

Removable visible watermarking can be classified into the following two categories: irreversible and reversible. We basically describe the reversible visible watermarking.

In the past, various reversible schemes have been developed using the techniques, these methods are applicable only to invisible watermarking.

The necessity for invertible visible watermarking is apparent. But unfortunately, this type of watermarking techniques has not been sufficiently investigated up to now. In the literature, to the best of our knowledge, there are only three works concentrating on distortion-free visible watermarking [6] – [8]. Hu et al. [6] first proposed a reversible visible watermarking scheme by modifying one significant bit plane of the pixels of the host image. They achieved reversibility via losslessly hiding the compressed version of the altered bit plane into the non-watermarked image region. However, the embedded visible watermark with this method appears to be somewhat blurred, and the visual quality of the original image is significantly distorted. Yip et al. [7] presented two lossless visible watermarking methods based on pixel value matching and pixel position shift, respectively.

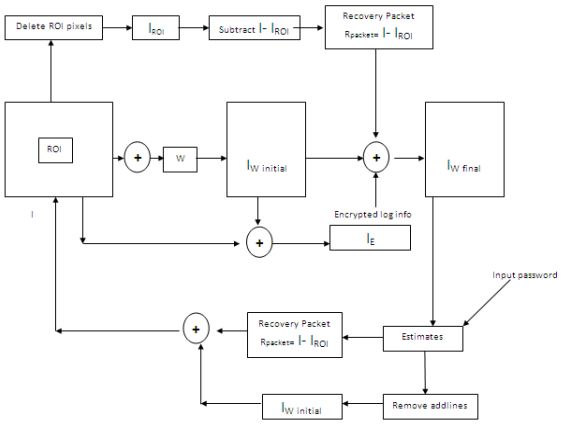

Fig1 : Working Diagram of watermarking

Tsai et al. [8] mapped the pixel values of the host image underlying the watermark into a small range for showing the watermark and then reversibly inserted a reconstruction packet into the watermarked image for perfect restoration.

By seeing the merit of [7] and [8], they need the original watermark for original image recovery, making them unsuitable for most applications in which the original watermark is unavailable at the recovery stage. Moreover, all the existing three methods do not consider human visual system (HVS) characteristics in the visible watermark embedding process. As a result, they are less visually satisfactory and more intrusive.

To addressing the issues of the aforementioned methods and maintaining applicability, we propose a lossless visible watermarking scheme that adaptively varies the watermark strength to be embedded in different areas of the host image, depending on the underlying image content and HVS characteristics. For reversibility, a recovery packet is embedded into the image itself. We develop a simple pixel prediction technique, and also exploit data compression, in order to alleviate the packet overhead and to improve embedding capacity. In addition, the proposed method adopts a unique encoding scheme for the recovery packet. This ensures that the original watermark pattern is not necessarily required when recovering the original host image.

We also provides the techniques for obtaining reconstruction data packet of reduced size and the possibility of devising a general metric for evaluating the visibility of visible watermark.

II. WATERMARKING EMBEDDING & DATA HIDING

A. Visible Watermark Embedding

In this process a secondary image (the watermark in different regions) is inserted into a primary (host) image so that the watermark is visible to the human eye.

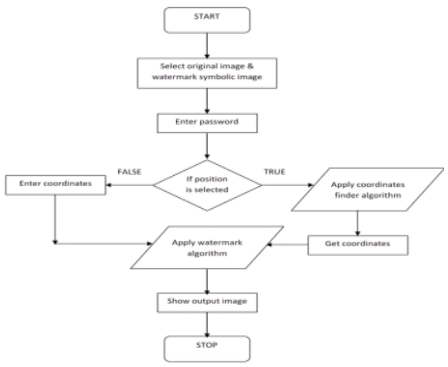

Fig2 : Working flow chart of watermark

III. ALGORITHM FOR REVERSIBLE VISIBLE WATERMARKING

1. Initialize image as original image.
2. Get the height and width of original image and denote it as originalimage. height and originalimage width.
3. Initialize watermark symbol as watsym.
4. Get the height and width of watsym, and denote it as watsym.height and watsym.width
5. Initialize password as pass.
6. Initialize the location points where we want to insert watermark and denote them as x, y
7. Now convert watsym.height and watsym.width to their ascii values and find the length of them and denote them as watsym.height.length and watsym.width.length. Same for x & y coordinate and denote them as x.length & y.length
8. Now finally convert the password into their ascii values and get the length of it and denote them as pass.length
9. Now find the lines require to store all the information related to password security and recovery packet by using given formula:

Int length = 3+ pass.length+5+ watsym.width.length + 5 + watsym.height.length + 5 + watsym.width * watsym.height + 5 + x.length + 5 + y.length + 5.

```
If(length  % originalimage.height>0) then
 addlines =length/originalimage.height + 1;
 else
 addlines =length/originalimage.height;
```

10. Once we get the addlines we create a new blank image using the new parameters which is given below
    Bmpnew=new bitmap (piccontainer.width + addlines, piccontainer.height);

11. Now we create every pixels using one grammar as given below which is basically designed for to stop hacking on an image.

IV. PASSWORD SECURING ALGORITHM

1. Start

2. Input password

3. Initialize

Int[]passarray = new int [password.length];

4. int i=0;

5. if (i<password.length) then go to step 6

Else go to step 9

6. passarray[i]convert.toint32(password[i]);

7. i++

8. jump to step 5

9. stop

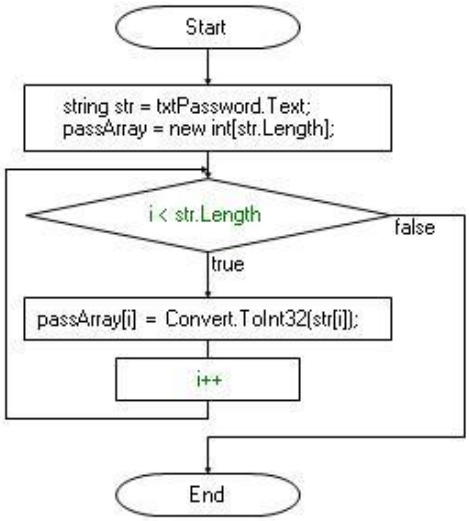

Fig 3: Password encryption flowchart diagram

V. LOG GRAMMAR IS GIVEN BELOW

A. First three pixels for to store some information for checking whether the image is watermarked or not.

Pixel (0,0)=(255,255,255)

Pixel (0,1)=(0,0,0)

Pixel (0,0)=(255,255,255)

B.Now add password into the pixels as

Pixel (x,y)=(password,0,0)

C. Now add five blank pixels for closing sign of password as Pixel (x,y)=(0,0,0)

D. Now add watSym.Width into the pixels as

Pixel (x,y)=(width,0,0)

E. Now add five blank pixels for closing sign of width as

Pixel (x,y)=(0,0,0)

F. Now add watSym.Height into the pixels as

Pixel (x,y)=(Height,0,0)

G. Now add five blank pixels for closing sign of Height as

Pixel (x,y)=(0,0,0)

H.Now add recovery packet pixels as it is,

pixels as Pixel (x,y) =RecPixel(I,j) I. Now add five blank pixels for closing sign of Height as

Pixel (x,y)=(0,0,0)

Finally add the remaining original image with added watermark symbol.

VI. WATERMARK REMOVAL AND ORIGINAL IMAGE RECOVERY

During recovery, we do not need the original watermark pattern, but some side information to help the decoder. The side information includes the secret key, the spatial position of the ROI in the host image, watermark pattern size. First, we extract the embedded binary sequence from the area outside the ROI of the image to obtain the watermarked image Iw. Second, using the secret key, we produce the same {0, 1} sequence in the embedding process and perform bitwise Exclusive-OR operation on the key-controlled binary sequence and the extracted binary sequence. Third, the encoded payload De is attained after applying decompression to the decrypted data, and furthermore, by decoding De **we** reconstruct the original reconstruction packet and the original watermark pattern W (see Section II-C). Fourth, we apply the pixel prediction technique as utilized during embedding to the watermarked image Iw so as to generate a roughly estimated version of the original host image I. Note that in the estimation process, the watermark W constructed earlier is required to indicate which pixels in the marked image have been watermarked. This information helps the decoder know which pixels need to be estimated. Fifth, according to the estimated image, we calculate the estimated scaling factor, and furthermore, obtain the approximate version Ia of the original image I after plugging αn into and to remove W from Iw. Based on the original pixels within ROI of the host image I can be recovered via pixel-to-pixel addition of the reconstruction packet D and the ROI of the approximate image I a: that is, IROI = Ia ROI + D. Finally, we losslessly retrieve the host image I by replacing the pixels in ROI of the watermarked image Iw with corresponding values in IROI.

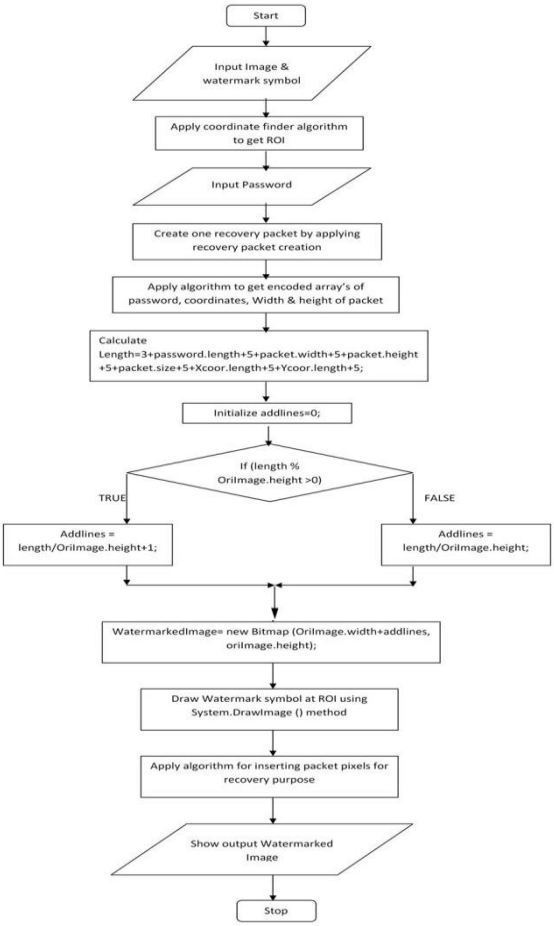

Fig 4: Flowchart

VII. ACTUAL WORKING OF MODULES

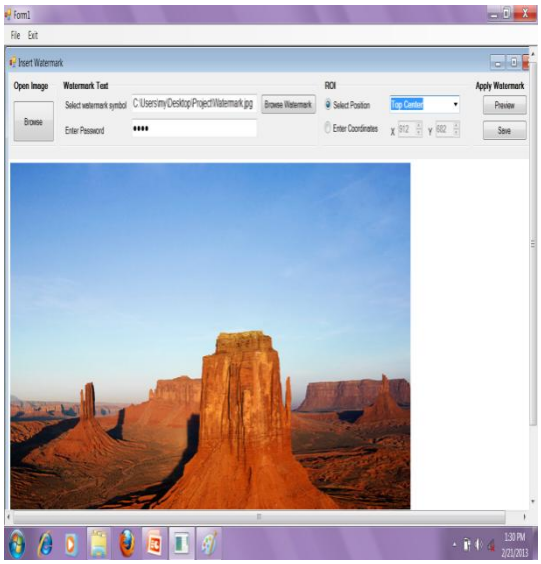

Fig 5: Original Image

This is the original image which is selected to apply a watermark symbol to provide ownership. Here, image is selected in bmp format because this bmp (bitmap) image does not make a large distortion in an original image.

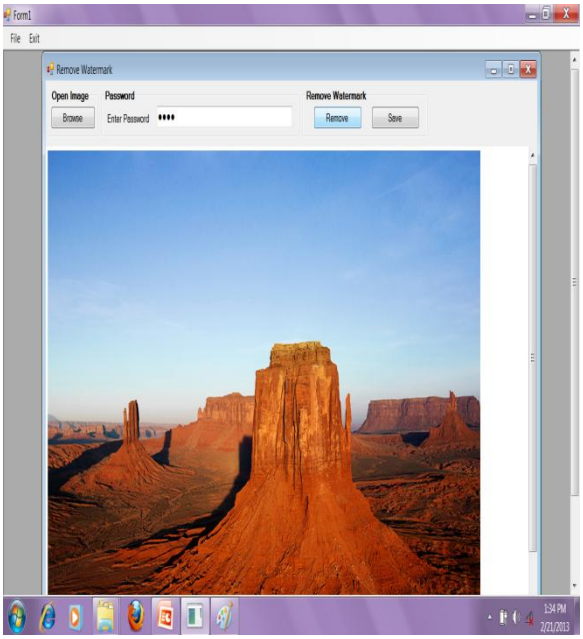

Fig 6: Authentication

This image show the authentication applying process to provide a security. In this paper provide a full authority to owner for accessing image.

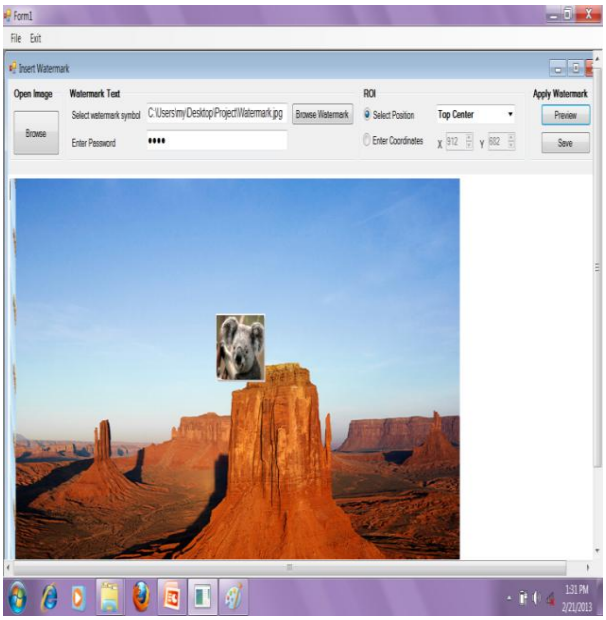

Fig 7: Watermark Image

This is the watermark image .here, in this image at very first pixels are used to store a recovery packet & user password.

VIII. CONCLUSION

This paper studies watermark embedding to achieve semi fragile multimedia authentication through a composite hypothesis testing approach. Our results show that the quantization- based embedding method outperforms spread spectrum in the tradeoff between algorithm robustness and fragility. Based on the hypothesis testing model, we also analyze certain common image-processing distortions, such as JPEG compression and filtering, and demonstrate how our approach can distinguish effectively minor changes from severe ones in quantization-based authentication watermarking. The results in this paper show that the hypothesis testing model provides insights for authentication watermarking and allows better control of robustness and fragility in specific applications

IX. REFERENCES

[1] M. S. Kankanhalli, Rajmohan, and K. R. Ramakrishnan, "Adaptive visible watermarking of images," in Proc. IEEE Int. Conf. Multimedia Comput. Syst., vol. 1. Florence, SC, Jul. 1999, pp. 568–573.

[2] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in Proc. 27th Ann. Conf. Comput. Graph. Interactive Technol. New Orleans, LA, 2000, pp. 417–424.

[3] R. Lukac and K. N. Plataniotis, "Secure single-sensor digital camera," Electron. Lett., vol. 42, no. 11, pp. 627–629, May 2006.

[4] S. C. Pei and Y. C. Zeng, "A novel image recovery algorithm for visible watermarked images," IEEE Trans. Inf. Forens. Security, vol. 1, no. 4, pp. 543–550, Dec. 2006.

[5] Y. J. Hu and B. Jeon, "Reversible visible watermarking and lossless recovery of original images," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 11, pp. 1423–1429, Nov. 2006.

[6] H. M. Tsai and L. W. Chang, "A high secure reversible visible watermarking scheme," in Proc. IEEE Int. Conf. Multimedia Expo, Beijing, China, 2007, pp. 2106–2109.

[7] Y. Yang, X. Sun, H. Yang, and C.-T. Li, "Removable visible image watermarking algorithm in the discrete cosine transform domain," J. Electron. Imaging, vol. 17, no. 3, pp. 033008-1–033008-11 Jul.–Sep. 2008.

[8] Vasiliy Sachnev, Hyoung Joong Kim, "Reversible Watermarking Algorithm Using Sorting and Prediction," in Proc. IEEE Int. Conf. on circuits & systems for video technology, 2009, pp. 989–999.

[9] Chuhong Fei, Raymond H. Kwong, and Deepa Kundur," A Hypothesis Testing Approach to Semi fragile Watermark-Based Authentication," in Proc. IEEE Int. Conf. on information forensic and security, 2009, pp. 179–192.

[10] Chun-Hsien Chou and Kuo-Cheng Liu," A Perceptually Tuned Watermarking Scheme for Color Images," in Proc. IEEE Int. Conf. on image processing, 2010, pp. 2966–2982

◈◈◈