www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 11 Nov 2015, Page No. 15095-15106

HIGHLY SECURE REVERSIBLE WATERMARKING MECHANISM USING REVERSIBLE DE-IDENTIFICATON

Fasiha Shereen[1] **B A Sharath Manohar**[2]

Department of ECE (DECS) [1], Associate professor, M.tech.,MIETE[2]

Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh, INDIA

Abstract

In the recent years, despite of massive work made in protecting and securing the image using watermarking schemes, there still exists some flaws. One way to get rid of this is usage of well built reversible watermarking scheme as proposed. This work intends to make the process: reversible ,invisible and highly secure. De-Identification is a process which can be used to ensure privacy by concealing the identity of individuals captured. This additional de-identification process aids in achieving high security. One important challenge is to make the obfuscation process reversible so that the original image can be recovered by persons in possession of the right security credentials. This work presents a novel Reversible De-Identification method that can be used in conjunction with any obfuscation process, here the obfuscation process used is k-same obfuscation process . The residual information needed to reverse the obfuscation process is compressed, authenticated, encrypted and embedded within the obfuscated image using a two-level Reversible Watermarking scheme. The proposed method ensures an overall single-pass embedding capacity of 1.25 bpp, where 99.8% of the images considered required less than 0.8 bpp while none of them required more than 1.1 bpp. Experimental results further demonstrate that the proposed method managed to recover and authenticate all images considered.

KEYWORDS: De-Identification, embedding capacity, obfuscated image, Reversible watermarking, PSNR, MSE,SSIM

1. INTRODUCTION

Due to increase in numerous crimes in present era, led to usage of digital multimedia almost everywhere, this in turn raises the issue of protecting personal information and has emerged as a much more important topic. Multitudinous researches and studies have been made to focus on safe guarding the images, such as facial images, where secrecy and privacy protection is needed. However, changes in the life environment caused by the rapid development of digital media techniques and communication media have changed users' emotional and sensual feelings. Consequently led to, the study on human senses and sensibility, in earnest to satisfy user needs through the convergence of many different fields. It is required to develop image obfuscation techniques which can minimize the infringement of user sensibility and simultaneously protect private life. Therefore, it is necessary to develop techniques to minimize the infringement of user sensibility and protect their privacy.

The goals of the reversible watermarking are to protect the copyrights and recover the original image. The robustness, imperceptibility, capacity, effectiveness, visual quality and the security are the basic criterion of the reversible watermarking. The reversible watermarking is especially suitable for the applications that require high quality images such as medical and military images. Reversible watermarking is also useful in remote sensing, multimedia archive management, law enforcement etc. It is a novel category of watermarking schemes. Reversible watermarking schemes have to be robust against the intentional or the unintentional attacks, and should be imperceptible to avoid the attraction of hackers. The robustness of the watermarked images can be verified on the parameters of PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error) and SSIM (Structural SIMilarity index) which show that the resulting quality of combination watermarking method is good.

Reversible De-Identification is a process which, while still concealing the identity of individuals, enables persons in possession of high security credentials to recover the original multimedia content containing private information. The authors in [4] encode the region of interest (ROI) and background in separate data layers using JPEG2000. On the other hand, the authors in [5,6] employ encryption strategies directly on the pixel intensities of the ROI. However, these methods completely destroy the naturalness of the content.

A ROI transform-domain scrambling technique is driven by a Pseudo Random Number Generator (PRNG) which is initialized by a seed value. The seed is encrypted, e.g. using public key encryption, and embedded in the compressed stream as private date. The method is fully reversible. Namely, authorized users, in possession of the secret encryption key, can reverse the scrambling process and recover the truthful image. The scrambling is confined to ROI, whereas the background remains unaltered. There have been several image scrambling schemes for protecting confidentiality of sensitive images basically through cryptographic and Steganographic techniques. An image scrambling scheme basically transforms an image into another unintelligible image. In spite of these efforts, analysis indicates that security level is still not strong for images and multimedia data in general. Also these techniques barely consider the significant intrinsic properties of images. This indicates content based schemes which are simpler yet stronger for shielding confidentiality of digital images. But the scrambling maintains the process merely maintains the naturalness of image.

Non-reversible watermarking was adopted[3] to solve the latter issue and embed the information which is needed to recover the De-Identified region. Here both these schemes are irreversible and the noise introduced by the watermark embedding process remains permanent, overall compression efficiency is reduced. This author [11] employs reversible watermarking to solve the former issue. This method induces significant distortions within the obfuscated image themselves.

This work presents a Reversible De-Identification method for lossless images which is used to provide additional security. And the approach adopts Reversible Watermarking to make the system reversible. The proposed solution employs the k-Same obfuscation process, which ensures k-anonymity, to obfuscate the face of frontal images. The difference between the original and obfuscated image is compressed, authenticated, encrypted and embedded within the obfuscated image itself. This method keeps the naturalness of the obfuscated images while the original image can only be recovered by individuals having the proper encryption key.

The Reversible Watermarking scheme adopted in this work was found to outperform existing state-of-the-art schemes. Furthermore, experimental results demonstrate that the proposed scheme can recover and authenticate all obfuscated images considered.

2. SYSTEM OVERVIEW

Figure 1: Illustrates the schematic diagram of the Forward Reversible De-Identification **Figure 2:** Schematic diagram of the Inverse Reversible De-Identifications Process.

Fig 1 receives the original image **I** and conceals the face of the person using the Face Obfuscation process to generate an obfuscated image $\mathbf{I}_{\theta}$. This work considers color images using the YCbCr color space. The coordinates of the top left corner and bottom right corner of the De-identified region is enclosed within the bounding box $\boldsymbol{\beta}$, which is passed to both ROI Extraction processes to extract the face image **F** and the obfuscated face image $\mathbf{F}_{\theta}$. The face images are then subtracted to derive the difference face image **D**. The Payload Generator process is then used to convert the difference face image **D** and bounding box $\boldsymbol{\beta}$ into a packet $\mathbf{p}_a^e$which is authenticated and encrypted. The packet $\mathbf{p}_a^e$ is then embedded within the obfuscated image $\mathbf{I}_{\theta}$ using the Integer Wavelet Transform (IWT) Reversible Watermark Embedding process (1st level) which generates the embedded image $\mathbf{I}_{\theta}^{W}$, the auxiliary information A and the residual bitstream e. This method provides a good compromise between capacity and distortion. However, additional information might be needed at the receiver to resolve overflow and underflow issues. The Reversible Contrast Mapping Embedding process (2nd level) is therefore used to embed this information (A and e) within the embedded image $\mathbf{I}_{\theta}^{W}$and, which usually corresponds to few bits, and generates the second level embedded obfuscated image $\hat{I}_{\theta}^{W}$and. This method is ideal since it does not need additional information to resolve overflow/underflow issues. Moreover, the distortions introduced at low bitrates are generally negligible. However, its performance significantly degrades at higher bitrates and is therefore not suitable to embed large payloads.

Fig. 2 depicts the schematic diagram of the Inverse Reversible De-Identification process. The second order embedded obfuscated image $\hat{I}_{\theta}^{W}$and is inputted to the Reversible Contrast Mapping Extraction process which extracts the first level embedded obfuscated image $I_{\theta}^{W}$and together with the auxiliary information A and the residual bit stream e. The IWT Reversible Watermark Extraction process is then used to extract the original payload $\mathbf{p}_a^e$ and original obfuscated image $\mathbf{I}_{\theta}$. The Inverse Payload Generator reverses the process of the Payload Generator and recovers the difference image **D** and the bounding box $\boldsymbol{\beta}$, which is used by the ROI.

3. FORWARD REVERSIBLE DE-IDENTIFICATION

3.1 Face Obfuscation

The Face Obfuscation process receives the original image ***I*** and detects the face region and eye locations using the ground truth information available in the color FERET dataset. However, the main contribution of this work is to present a Reversible

De- Identification method. The upper left and bottom right coordinates of the face region are included in the bounding box **β** and used to extract the face **F** which is aligned using affine transformations. The aligned face image **F** is then concealed using the k-same obfuscation process, which computes the average face derived over the k closest aligned faces in Eigen-space, to generate the obfuscated aligned face image $\mathbf{F_\theta}$. The obfuscated face image $\mathbf{F_\theta}$ is then realigned to match the orientation of the original face image **F** using affine transformations and then overwrites the face region in the original image **I** to derive the obfuscated image $\mathbf{I_\theta}$.

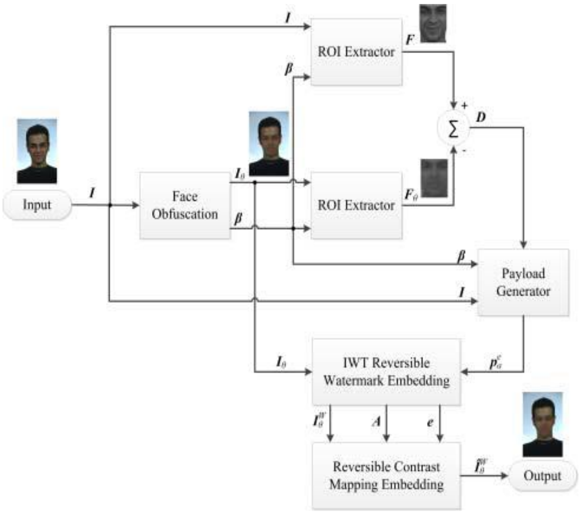


Figure 1: Block diagram of Forward Reversible De Identification Process

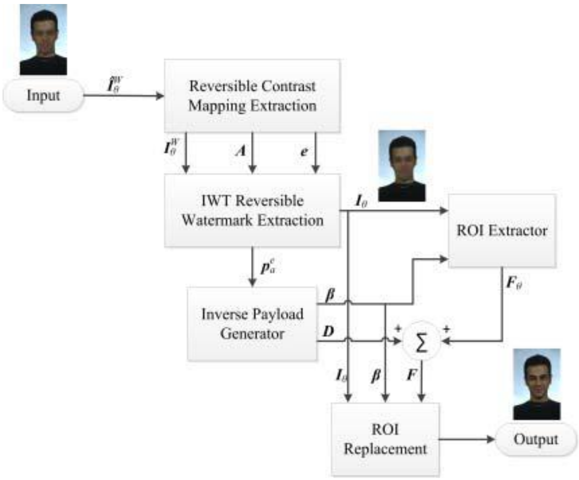


Fig.2 Block diagram of Inverse Reversible De Identification Process

3.2 ROI Extraction

The ROI Extraction process is a simple process which employs the bounding box coordinates **β** to identify the region to be cropped from the input image ***I*** (or $\mathbf{I_\theta}$). The cropped sub-image is then stored in the face image ***F*** (or $\mathbf{F_\theta}$ obfuscated face image).

3.3 Payload Generator

The Payload Generator Process receives the difference image ***D*** which is compressed using the predictive coding method presented in [15] . The image is authenticated using SHA-1 which generates a 20-Byte Hash. The Hash will be used by the Inverse Reversible De- Identification process to ensure that it recovers the original image **I**, and is thus appended to the Payload. The bounding box coordinates **β** are also required at the receiver to identify the face region and are therefore included as information within the header. The resulting packet $\mathbf{p_a}$ illustrated in Fig. 3, was then encrypted using AES-128 to generate the encrypted packet $\mathbf{p_a^e}$. **Figure 3:** The authenticated packet $\mathbf{p_a}$.

β	Payload	Hash

Figure 3: The authenticated packet p_a

3.4 IWT Reversible Watermarking Embedding

The IWT Reversible Watermarking Embedding process first derives the number of decompositions $\mathbf{N_{dec}}$ needed to embed the packet $\mathbf{p_a^e}$ within $\mathbf{I_\theta}$ using

$$\mathbf{N_{dec}} = \begin{cases} \left[-\frac{1}{2}\log_2(1-\mathrm{C})\right] & \text{if } \mathrm{C} < 1 \\ \mathrm{M} & \text{otherwise} \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

where **M** indicates the maximum number of decomposition allowed and **C** represents the capacity needed to embed $\mathbf{p_a^e}$ bits and is computed using

$$\boldsymbol{C} = \frac{|\mathbf{p_a^e}|}{\boldsymbol{Ch \times W \times H}} \quad \ldots\ldots\ldots\ldots\ldots\mathbf{(2)}$$

where| | represents the cardinality of the set, **W** and **H** represent the number of columns and rows in the image and **Ch** represents the number of color channels (in our case 3). This process then adopts the CDF(2,2) integer wavelet transform specified in [17] to decompose the image

3.4.1 Threshold Selection

The proposed Threshold Selection method is based on the observation that different sub-bands provide different levels of distortions [18]. However, in order to reduce the complexity of the optimization function, the following assumptions were made

- The chrominance sub-bands have similar properties and thus share the same threshold.
- The **HL** and **LH** sub-bands within the same color channel (luminance or chrominance) are assumed to have similar characteristics and therefore have the same threshold.

This work employs Differential Evolution (DE), which is a population based optimization algorithm, to derive the set of threshold which minimize a distortion criterion while ensuring that the capacity of the proposed system is sufficient to embed the message s.

3.4.2 Forward Integer Wavelet Expansion

The Forward Integer Wavelet Expansion process receives the set of thresholds **T** which are derived by the Threshold Selection process and encapsulates the packet $\mathbf{p_a^e}$ shown in Fig. 3 to generate the packet **s** to be Embedded in fig 4

N_{dec}	T	L	$\boldsymbol{p_a^e}$

Figure 4: The actual bit stream to be embedded *s*

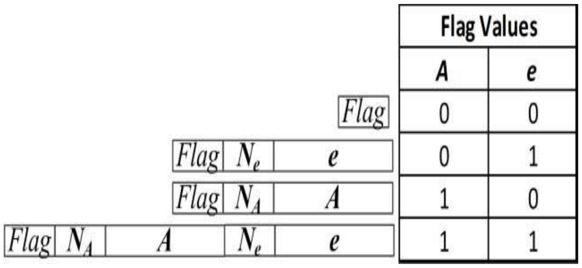
A wavelet coefficient $\mathbf{w_{\mu,j}}$ is considered for embedding if its magnitude is smaller than the threshold responsible of sub-band μ. A bit b is embedded using

$$\mathbf{\hat{w}_{\mu,j}} = \mathbf{2w_{\mu,j}} + \boldsymbol{b} \ \ldots\ldots\ldots\ldots\mathbf{(3)}$$

This process is terminated once all the bits in ***s*** are embedded. In the rare event where this process fails to embed all bits, these bits are stored in a bit-sequence ***e***.

3.5 Reversible Contrast Mapping

The only problem with the proposed Forward Integer Wavelet Expansion process is that sometimes **A** and **e** are not empty. This work adopts the syntax shown in Fig. 5 to represent this information **r** to be embedded. The Flag is a 2-bit field which indicates whether **A** and **e** are empty or not. The main advantage of using RCM is that it embeds all information within the image without any ambiguities and provides an additional capacity of 0.5 bpp. However, the packet size ***r*** is expected to be very low (generally 2-bits).



Packet	Flag Values A	Flag Values e
Flag	0	0
Flag N_e e	0	1
Flag N_A A	1	0
Flag N_A A N_e e	1	1

Figure 5: the packet to be embedded within $\mathbf{I}_{\boldsymbol{\theta}}^{\mathbf{W}}$

The forward RCM transforms two neighboring pixel pairs $(\boldsymbol{x}, \boldsymbol{y})$ into $(\boldsymbol{x}', \boldsymbol{y}')$ using

$$\mathbf{x}' = \mathbf{2x} - \mathbf{y}, \mathbf{y}' = \mathbf{2y} - \mathbf{x} \quad \textbf{…….(4)}$$

To prevent overflow and underflow, the transform is restricted to a sub-domain defined by the pixels which satisfy the conditions

$$\mathbf{0} \leq \mathbf{2x} - \mathbf{y} \leq \mathbf{255}, \mathbf{0} \leq \mathbf{2y} - \mathbf{x} \leq \mathbf{255} \quad \textbf{……(5)}$$

The RCM scheme[21] replaces the least significant bits (LSBs) of the transformed pairs **(*x'*, *y'*).**

The LSB of **x'** is used to indicate whether information is embedded within **y'** or not**.**

4. INVERSE REVERSIBLE DE-IDENTIFICATION

4.1 Reversible Contrast Mapping Extraction

The Reversible Contrast Mapping Extraction process receives the image $\hat{\boldsymbol{I}}_{\boldsymbol{\theta}}^{\boldsymbol{W}}$and recovers $\boldsymbol{I}_{\boldsymbol{\theta}}^{\boldsymbol{W}}$and ***r***. The information bit can be extracted from the LSB of $x^{\acute{}}$when the LSB of $\acute{y}$ is '1'. However, in the event when the LSB of $x^{\acute{}}$ is '0', both LSBs of $x^{\acute{}}$and y' are forced to be odd and condition (5) is checked. If the condition is satisfied it then represents an odd pixel pair while if it does not it indicates that $y = \acute{y}$ and the original LSB value of *x* is extracted from the bit stream. More information about this is available in [21]. The auxiliary information ***A*** and residual bit stream ***e*** are then extracted from the packet ***r***. The original pixel values are recovered using

$$\boldsymbol{x} = \left[\frac{\mathbf{2}}{\mathbf{3}}\,\mathbf{x}' + \frac{\mathbf{1}}{\mathbf{3}}\mathbf{y}'\right], \boldsymbol{y} = [\frac{\mathbf{1}}{\mathbf{3}}\mathbf{x}' + \frac{\mathbf{2}}{\mathbf{3}}\mathbf{y}'] \quad \textbf{…..}(6)$$

4.2 IWT Reversible Watermarking Extraction

The IWT Reversible Watermarking Extraction reverses the IWT Reversible Watermarking Embedding process and extracts the payload information $\boldsymbol{p}_{\boldsymbol{a}}^{\boldsymbol{e}}$ and the original obfuscated image $\boldsymbol{I}_{\boldsymbol{\theta}}$. It must be noted here that initially, the decoder has no knowledge about the number of decompositions employed $\boldsymbol{N}_{\boldsymbol{dec}}$and the threshold values ***T***. The decoder thus assumes that a single decomposition is employed and that the threshold values are set to zero. These values are then updated once they are extracted from the header information of ***s***. It is important that the

encoder does the same thing during embedding in order to ensure synchronization between the encoder and decoder.

The watermarked obfuscated image $\boldsymbol{I}_{\boldsymbol{\theta}}^{\boldsymbol{W}}$ is then decomposed into N_{dec} levels using the CDF(2,2) integer wavelet transform and the wavelet coefficients of the high frequency sub-bands are scanned using the random permutation index generated using the encryption key. The bits of the packet are extracted from coefficients which satisfy the condition $-2T_{\mu,j} + 1 < \widehat{w}_{\mu,j} < 2T_{\mu,j}$ and are extracted from the LSB of $\widehat{w}_{\mu,j}$ while the original wavelet coefficient is recovered using

$$\mathbf{w}_{\boldsymbol{\mu},\mathbf{j}} = \left[\frac{\widehat{\mathbf{w}}_{\boldsymbol{\mu},\mathbf{j}}}{\mathbf{2}}\right] \quad \textbf{…………..(7)}$$

The remaining coefficients are not used for embedding and are recovered using

$$\mathbf{w}_{\mathbf{\mu,j}} = \begin{cases} \widehat{w}_{\mu,j} - T_{\mu} & if \;\; \mathbf{w}_{\mathbf{\mu,j}} \geq 0 \\ \overrightarrow{\mathbf{w}}_{\mathbf{\mu,j}} + T_{\mu} - 1 & otherwise \end{cases} \;\; \text{.......}\mathbf{(8)}$$

This method terminates once all the *L* bytes are extracted. The resulting image is then inverse CDF(2,2) transformed to recover the original obfuscated image I_{θ} . The auxiliary information (if any) is then used to recover ambiguous pixel values while the residual bit stream ***e*** (if any) is appended to the recovered payload.

4.3 ROI Replacement

The ROI Replacement process replaces the region marked by the bounding box **β** with the recovered face image ***F***. The image $\mathbf{I_{rec}}$ can be authenticated by comparing the hash derived by computing the *SHA-1* on $\mathbf{I_{rec}}$ to the *Hash* value present in the tail of the packet p_a.

5. SIMULATION RESULTS

The results presented in this section consider different sets of images. All images considered in this work were converted in the YCbCr color space and the outputs are shown in accordance to the proposed method. Further, the images are also converted into CIEL*a*b and compared with YCbCr color space in terms of parameters like MSE,PSNR,SSIM to prove its effectiveness. Later, the performance analysis is done on different standard images.

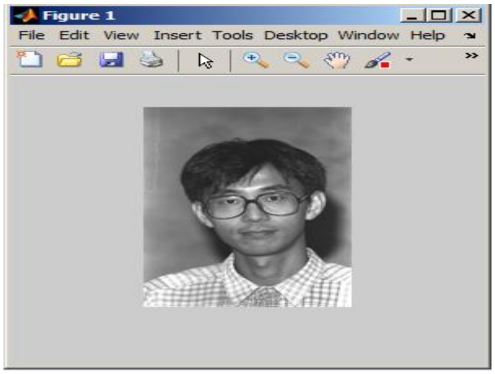


Figure 6: Input image

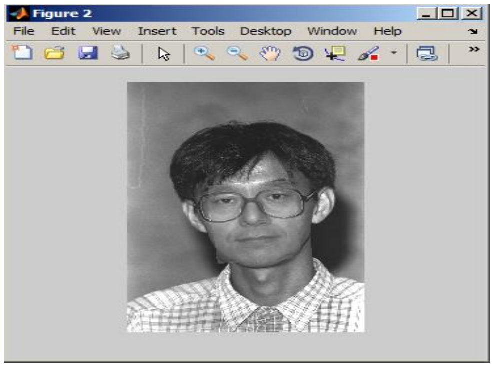
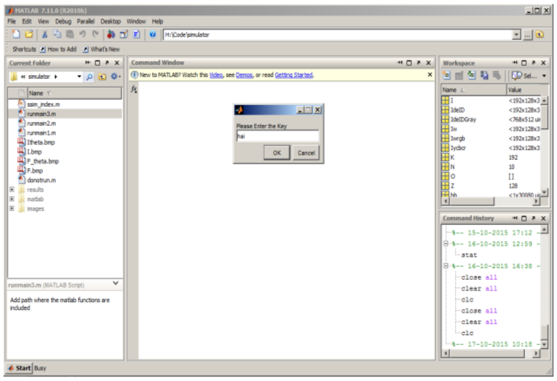


Figure 6: Obfuscated image



H:\Code\simulator

« simulator ›

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

Please Enter the Key

hai

Name
ssim_index.m
runmain3.m
runmain2.m
runmain1.m
Itheta.bmp
I.bmp
F_theta.bmp
F.bmp
donotrun.m
results
matlab
images

runmain3.m (MATLAB Script)

Add path where the matlab functions are included

Name	Value
I	<192x128x3
IdeID	<192x128x3
IdeIDGray	<768x512 ui
Iw	<192x128x3
Iwrgb	<192x128x3
Iycbcr	<192x128x3
K	192
N	10
O	[]
Z	128
hh	<1x30080 ui

```
%-- 15-10-2015 17:12 -
%-- 16-10-2015 12:59 -
  stat
%-- 16-10-2015 16:38 -
  close all
  clear all
  clc
  close all
  clear all
  clc
%-- 17-10-2015 10:18 -
```

Busy

Figure 7: Key image

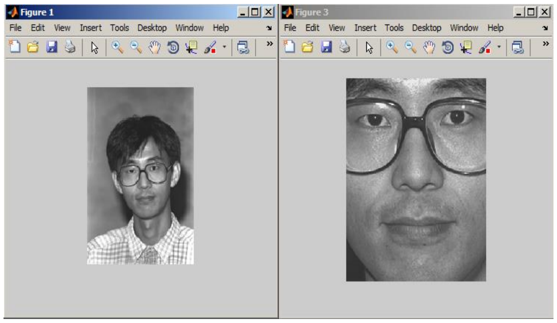


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 3
File Edit View Insert Tools Desktop Window Help

Figure 8: Extraction input

DOI: 10.18535/Ijecs/v4i11.39

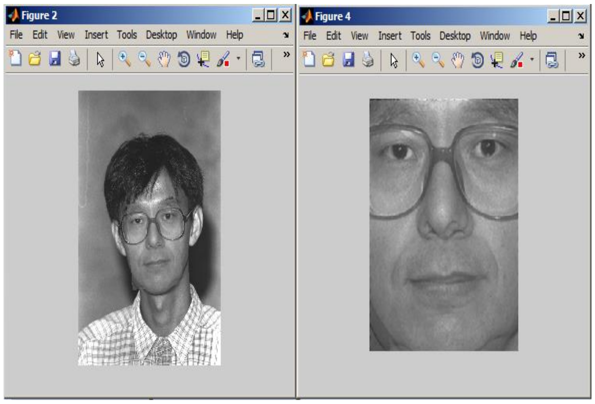
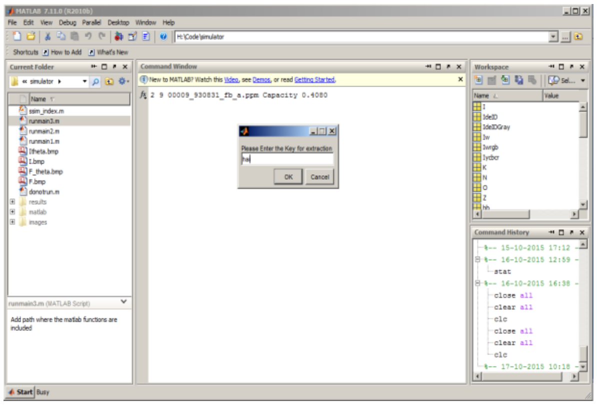


Figure 2
File Edit View Insert Tools Desktop Window Help
Figure 4
File Edit View Insert Tools Desktop Window Help

Figure 9: Obfuscation extraction



Add path where the matlab functions are included

```
2 9 00009_930831_fb_a.ppm Capacity 0.4080
```

Please Enter the Key for extraction

hai

```
%-- 15-10-2015 17:12 --%
%-- 16-10-2015 12:59 --%
stat
%-- 16-10-2015 16:38 --%
close all
clear all
clc
close all
clear all
clc
%-- 17-10-2015 10:18 --%
```

Figure 10: Extraction key

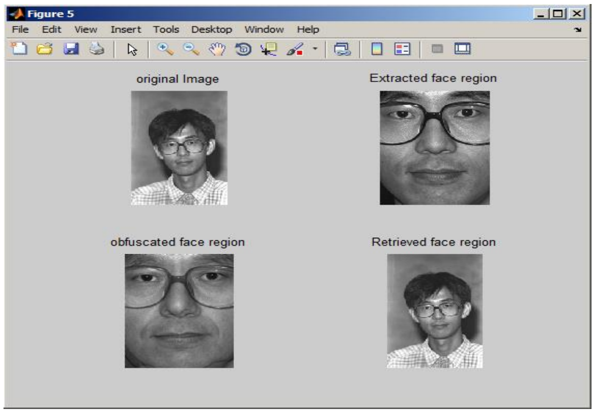


Figure 11: Retrieved face image

INPUT IMAGES	YCbCr				CIE L*a*b			
	Capacity (bpp)	MSE	PSNR (db)	SSIM	Capacity (bpp)	MSE	PSNR (db)	SSIM
IMAGE 1	0.4497	4.7623	41.3527	0.9995	0.4497	8.0566	39.0693	0.9993
IMAGE 2	0.3507	4.4434	41.6537	0.9996	0.3507	8.1761	39.0053	0.9993
IMAGE 3	0.3924	6.4481	40.0365	0.9991	0.3924	16.6258	35.9230	0.9977
IMAGE 4	0.3993	4.2987	41.7975	0.9996	0.3993	19.3274	35.2691	0.9986
IMAGE 5	0.3976	4.2043	41.8939	0.9996	0.3976	19.2334	38.4772	0.9992
IMAGE 6	0.4236	6.8553	39.7706	0.9995	0.4236	26.0768	33.9683	0.9983
IMAGE 7	0.5417	8.2381	38.9726	0.9990	0.5417	16.8707	35.8595	0.9981
IMAGE 8	0.4184	6.2330	40.1838	0.9993	0.4184	11.7999	37.4120	0.9987
IMAGE 9	0.4080	8.3256	38.9266	0.9988	0.4080	24.7326	34.1981	0.9972
IMAGE 10	0.3281	5.6478	40.6120	0.9993	0.3281	11.4357	37.5482	0.9985

Figure 11: Comparison between color spaces on different parameters

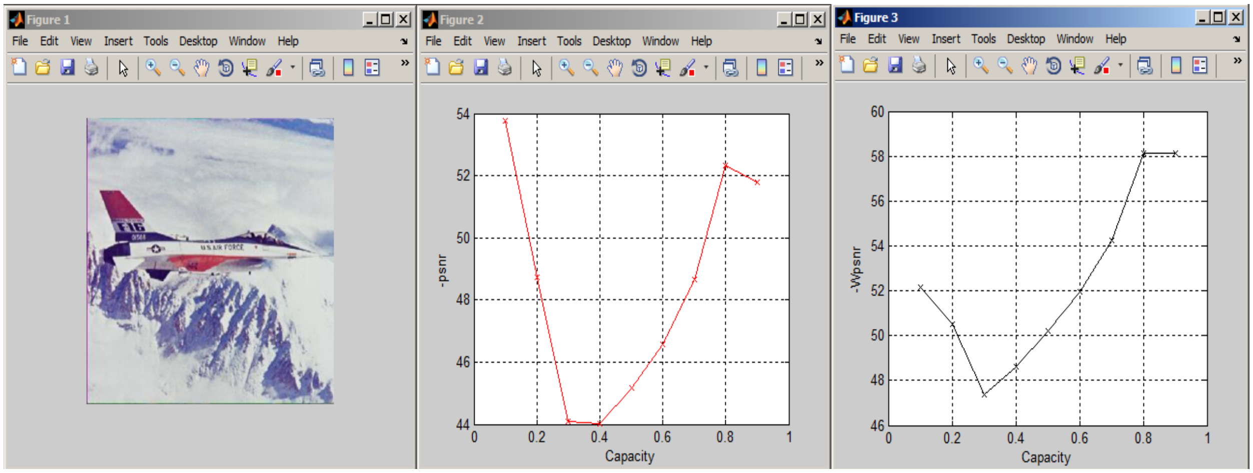


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 2
File Edit View Insert Tools Desktop Window Help
54
52
50
48
46
44
-psnr
0
0.2
0.4
0.6
0.8
1
Capacity
Figure 3
File Edit View Insert Tools Desktop Window Help
60
58
56
54
52
50
48
46
-Wpsnr
0
0.2
0.4
0.6
0.8
1
Capacity

Figure 12: Plane image and its performance analysis

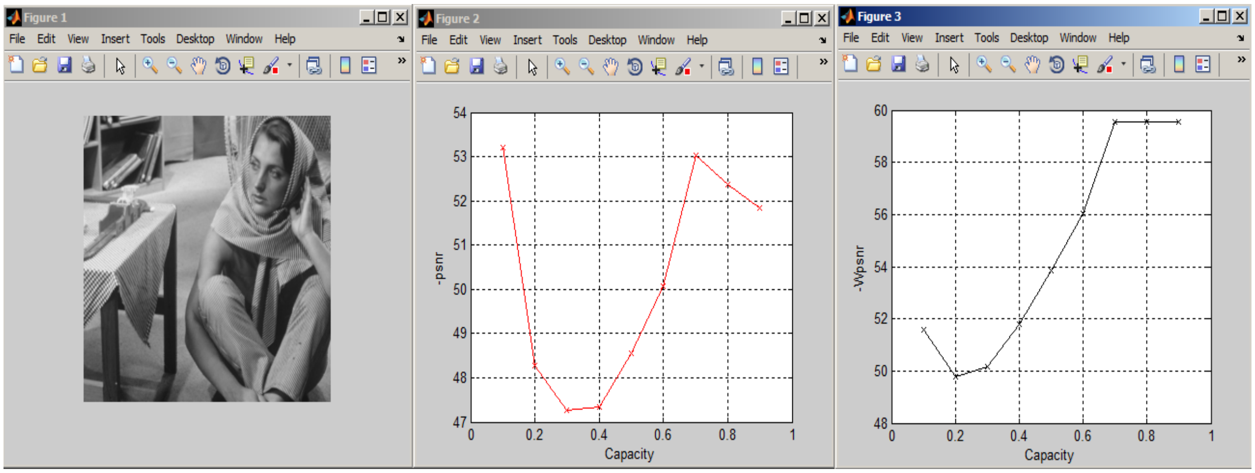


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 2
File Edit View Insert Tools Desktop Window Help
54
53
52
51
50
49
48
47
-psnr
0
0.2
0.4
0.6
0.8
1
Capacity
Figure 3
File Edit View Insert Tools Desktop Window Help
60
58
56
54
52
50
48
-Wpsnr
0
0.2
0.4
0.6
0.8
1
Capacity

Figure 13: Barbara image and its performance analysis

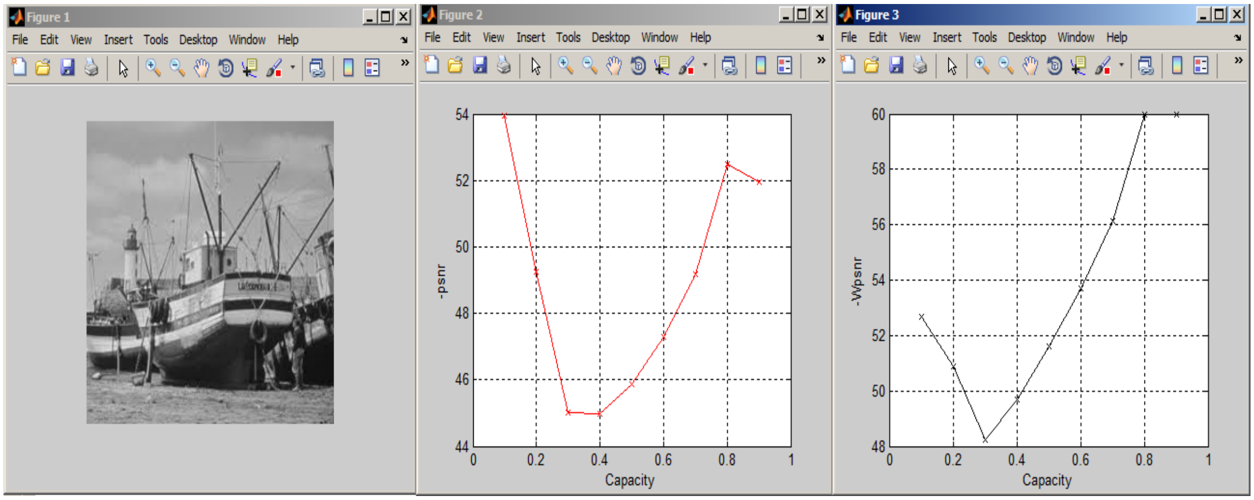


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 2
File Edit View Insert Tools Desktop Window Help
54
52
50
48
46
44
-psnr
0
0.2
0.4
0.6
0.8
1
Capacity
Figure 3
File Edit View Insert Tools Desktop Window Help
60
58
56
54
52
50
48
-Wpsnr
0
0.2
0.4
0.6
0.8
1
Capacity

Figure 14: Boat image and its performance analysis

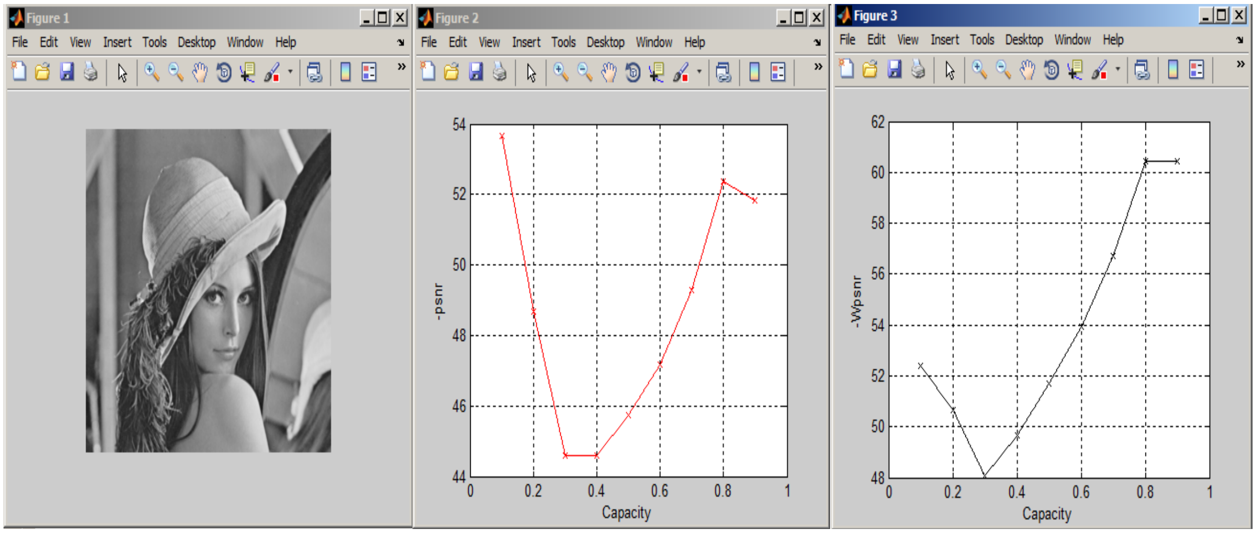


Figure 15: Lena image and its performance analysis

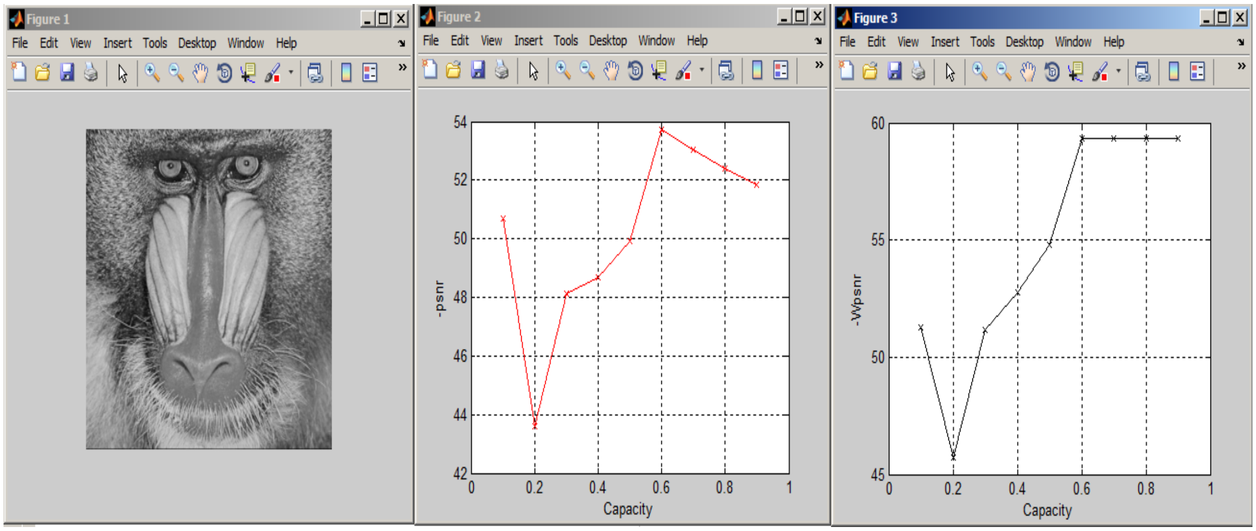


Figure 16: Mandrill image and its performance analysis

6. CONCLUSION

This work presents a novel Reversible De-Identification method for lossless images. The proposed scheme is generic and is employed with obfuscation strategy, k-Same. It also achieves the 3 basic needs: reversibility, invisibility and security. A two level Reversible-Watermarking scheme was adopted, which uses IWT for embedding the residual information and Reversible Contrast Mapping to handle overflow and underflow issues. Simulation results have shown that this method is able to recover the original image if the correct encryption key is employed. It further shows that 0.8 bpp were sufficient to cater for 99.8% of the frontal images considered and none of the image needed more than 1.1bpp. It also calculates various parameters like PSNR,MSE,SSIM which proves, its efficiency is good.

DOI: 10.18535/Ijecs/v4i11.39

7. FUTURE SCOPE

Future work will focus on the extension of this algorithm for lossy image and video compression standards.

REFERENCES

[1] Farrugia, R.A. Dept. of Commun. & Comput. Eng., Univ. of Malta, Msida, Malta, "Reversible De-Identification for lossless image compression using Reversible Watermarking", *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014*

[2] E.M. Newton, L. Sweeney and B. Malin, "Preserving privacy by de-identifying face images," *IEEE Trans. on Knowl. and Data Eng.*, vol. 17, no. 2, pp. 232-243, Feb. 2005.

[3] W. Zhang, S.S. Cheung and M. Chen, "Hiding privacy information in video surveillance systems," in *IEEE Int. Conf. on Image Processing*, Genoa, Italy, Sep. 2005.

[4] I. Martinez-Ponte, X. Desumont, J. Meessen and J.F. Delaigle, "Robust Human Face Hiding ensuring Privacy," in *Proc. of Int. Workshop on Image Analysis for Multimedia Services*, Montreux, Switzerland, Apr. 2005.

[5] T.E. Boult, "Pico: Privacy throough invertible cryptographic obscuration," in *IEEE Proc. of the Computer Vision for Interactive Intelligent Environment*, Whashington DC, USA, Nov. 2005.

[6] K. Martin and K.N. Plataniotis, "Privacy protected surveillance using secure visual object coding," *IEEE Trans. Circuits and System for Video Technol.*, vol. 18, no. 8, pp. 1152-1162, Aug. 2008.

[7] F. Dufaux, M. Ouaret, Y. Abdeljaoued, A. Navarro, F. Bergnenegre and T. Ebrahimi, "Privacy Enabling Technology for Video Surveillance," in *SPIE Mobile Multimedia/Image Processing for Military and Security Applications*, Orlando, Florida, May 2006.

[8] F. Dufaux and T. Ebrahimi Scrambling for privacy protection in video surveillance systems, "Scrambling for privacy protection in video surveillance systems," in *IEEE Trans on Circuits andSystems for Video Technolgy.*, vol. 18, no. 8, pp. 1168-1178, Aug. 2008.

[9] H. Sohn, W. De Neve and Y-M. Ro, "Privacy protection in video surveillance systems: Analysis of subband-adaptive scrambling in JPEG XR," in *IEEE Trans. Circuits and Systems for Video Technol.*, vol. 21, no. 2, pp. 170-177, Feb. 2011.

[10] J. Meuel, M. Chaumont and W. Puech, "Data hiding in H.264 video for lossless reconstruction of region of interest," in *European Signal Processing Conf.*, Poznan, Poland, Sep. 2007.

[11] S.S. Cheung, J.K. Panichuri and T.P. Nguyen, "Managing privacy data in pervasive camera networks," in *IEEE Int. Conf. on Image Processing*, San Diego, California, USA, Oct. 2008.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Proc. Computer Vision and Pattern Recognition*,Kauai, USA, Dec. 2001.

[13] F. Hahmann, G. Boer and H. Schramm, "Combination of Facial Landmarks for Robust Eye Localization using the Discriminative Generalized Hough Transform," in *Int. Conf. of the Biometrics Special Interest Group*, Germany, Sep. 2013.

[14] R. C. Gonzalez and R.E. Woods, *Digital Image Processing*, Second Edition, Prentice Hall, 2001.

[15] M. Weinberger, G. Seroussi and G. Sapiro, "LOCO-I: A Low Complexity, Context-Based, Lossless Image Compression

Algorithm," in *Proc. IEEE Data Compression Conf.*, Washington DC., USA, Apr. 1996.

[16] P. Deutsch, *DEFLAGE Compressed Data Format Specification version 1.3*, May 1996.

[17] G. Xuan, Y.Q. Shi, P. Chai, X. Cui, Z. Ni and X. Tong, "Optimum Histogram Pair based image Lossless Data Embedding," in *Proc. Int. Workshop on Digital Watermarking*, Berlin, Germany, 2008.

[18] Z. Wang, E.P. Simoncelli and A.C. Bovik, "Multi-Scale Structural Similarity for Image Quality Assessment, in *IEEE Proc .Asilomar Conf. on Signals, Systems and Computers*, USA, Nov. 2003.

[19] R. Storn, K. Price, "Differential Evolution: A simple and efficient heuristic for global optimization over continuous spaces," *J. on Global Optimization*, vol.11, no. 4, pp. 341-359, Dec. 1997.

[20] F. De Simone, M. Ouaret, F. Dufaux, A.G. Tescher and T. Ebrahimi, "A Comparative study of JPEG2000, AVC/H.264 and HD Photo," in *Proc. SPIE Optics and Photonics, Applications of Digital Image*, San Diego, USA, Aug. 2007.

[21] D. Coltuc and J-M. Chassery, "Very Fast Watermarking by reversible Contrast Mapping," *IEEE Sig. Proc. Letters*, vol. 14, no.4, Apr. 2007.

[22] T. Ahonen, A. Hadid and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no.