International Journal of Electrical, Electronics and Computer Engineering **4**(2): 75-86(2015)

ISSN No. (Online): 2277-2626

Implementation of Technique for Image Authentication using Slepian Wolf Coding

Soheb Munir, A.S. Zadgaonkar and Manish Shrivastava
[*]*Department of Electronics and Communication Engineering,*
AISECT University, Bhopal, (MP), India,

(Corresponding author: Soheb Munir)

(Received 13 September, 2015 Accepted 29 October, 2015)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: **This paper investigates the performance and proposes modifications to earlier methods for image authentication using distributed source coding and LDPC. New ways of verification are being developed daily to daily life of humans. Biometrics and other methods keep getting formulated and incorporated into the information technology industry. One interesting biometric authentication mechanism developed by a leading Japanese biometric company has found a way to get your DNA. You sign a document and it is digitally scanned, this document is then can be scanned in the future to verify its authenticity. This approach works well on images that have undergone affine geometric transformations such as rotation and resizing and intensity transformations such as contrast and brightness adjustment. The results show that the improvements proposed here can be used to make the original scheme for image authentication robust to affine geometric and intensity transformations. The modifications are of much lesser computational complexity when compared with other schemes for estimation of channel parameters.**

Keywords: Multimedia authentication, image authentication, distributed source coding, LDPC, biometric authentication.

I. INTRODUCTION

Image authentication techniques have recently gained great attention due to their importance for a large number of multimedia applications. Digital images are increasingly transmitted over non-secure channels such as the Internet. Therefore, military, medical and quality control images must be protected against attempts to manipulate them; such manipulations could tamper the decisions based on these images. The risks for security are further exacerbated by the improved possibilities of tampering with media contents such as photos, an ability that would have traditionally required many hours of cumbersome work in a darkroom and that has become now a simple practice using a computer and some commercial software tools. Relying on digital media such in law enforcement and security makes robust techniques for media authentication a must. These techniques are also vital in content delivery via untrusted intermediaries such as video streaming.

Watermarks and media hashes have been used in past for media authentication. Depending upon the application scenario, “robust,” “semi-fragile” or “fragile” watermarks can be embedded into the authentic media such as digital image, audio or video. Authenticity of the media content can be verified by extracting the watermarks [1, 2]. Media hashes [3] are designed in such a way that they remain the same for different encodings of the same media as long as the encoded versions are perceptually the same. On the request of user, the media hash is sent by a server. The user checks the authenticity of his content by matching the hash value generated from his content with the authentication data.

In [4], a method for backward-compatible image authentication based on distributed source coding is presented. This method provides a Slepian-Wolf encoded [5] quantized image projection as the authentication data which can be successfully decoded only by using an authentic image as side information. Distributed source coding helps in achieving false acceptance rates close to zero for very small authentication data size. This scheme has been extended for tampering localization in [6]. The fixed decoder used in [4, 6] can do successful image authentication for JPEG compressed images but image authentication is not possible using fixed decoder if the channel applies contrast and brightness adjustment in addition to JPEG compression. One solution to this problem is the use of the expectation-maximization (EM) algorithm for estimating parameters [7, 8].

This paper proposes some improvements in the scheme described in [4] to make it robust to affine transformations. In rest of the paper, we first describe the proposed improvements to the existing scheme and then present detailed experiment results showing the efficacy of the proposed scheme. This is followed by a brief conclusion.

II. LITERATURE REVIEW

Image authentication using distributed source coding was first proposed by Lin *et. al.* in [4] and was later extended for tampering localization [6]. The methods presented in these papers work well in differentiating legitimate JPEG/JPEG2000 compressed images from illegitimate versions with a small banner inserted in the image using a very small amount of authentication data. But these methods are not robust to non-malicious operations such as global contrast and brightness enhancement, and rotation. In [7, 8], these methods are modified to include the use of the EM algorithm in the Slepian-Wolf decoder for learning parameters of the global affine contrast and brightness operation.

A. Lossless Distributed Source Coding

The problem of compressing features *X* of the original image relative to features *Y* of the target image is a distributed source coding problem as shown in Fig. 1. Source *X* is available at the encoder, but the side information *Y* is available at the decoder only. Slepian and Wolf proved that *X* can be compressed to a rate $R \geq H\ (X|Y)$ and still be decoded without loss in the presence of Y. Conversely, when *R* is less than $H(X|Y)$, the probability of decoding error will be bounded away from zero. State-of-the-art practical Slepian-Wolf coding often employs low-density parity-check (LDPC) codes. The work reported in this paper likewise uses LDPC codes and employs them to efficiently encode random projections of images.

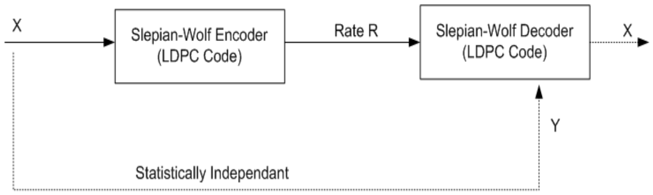

X
Slepian-Wolf Encoder (LDPC Code)
Rate R
Slepian-Wolf Decoder (LDPC Code)
X
Y
Statistically Independant

Fig. 1. The source X and side information Y are statistically dependent, but Y is available only at the decoder.

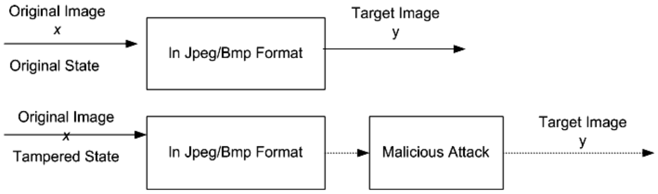


Fig. 2. The target image y is modeled as an output of a two-state channel. In the original state, the channel consists of JPEG and BMP; in the tampered state, the channel further applies a malicious attack.

B. Slepian-Wolf Coding

The Slepian-Wolf theorem deals with the lossless compression of two or more correlated data streams (Slepian and Wolf, 1973). In the best-known variation, each of the correlated streams is encoded separately and the compressed data from all these encoders are jointly decoded by a single decoder as for two correlated streams. Such systems are said to employ Slepian-Wolf coding, which is a form of distributed source coding. Lossless compression means that the source outputs can be constructed from the compression version with arbitrary small error probability by suitable choice of a parameter in the compression scheme.

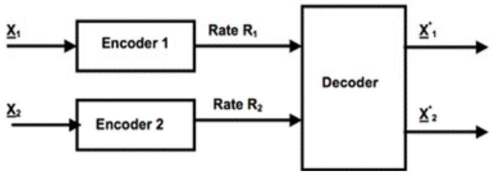


Fig. 3. Block diagram for Slepian-Wolf coding.

An important aspect of the Slepian-Wolf theorem is that, compared to an encoder that assumes the data streams are independent; the separate encoders can achieve better compression rates by exploiting the fact that the data streams are correlated. The surprising result is that Slepian-Wolf coding can in fact achieve the same compression rate as the optimal single encoder that has all correlated data streams as inputs. The Slepian-Wolf theorem has practical consequences for systems where the correlated data streams are physically separated or where the encoder has limited computational ability. It can be applied to sensor networks such as those for monitoring temperature or seismic activity where wireless transmitters, distributed over some environment, collect data and transmit it to central location. Two transmitters that are near each other sense similar values and thus produce correlated outputs. Because transmitter resources such as battery power are limited, transmitting at higher compression rates improves the system's performance. The Slepian-Wolf theorem has practical application even when the encoder has access to the multiple correlated data streams. For example, to reduce complexity for image and video compression for mobile telephones, the streams may be encoded separately without reducing the compression rate.

C. Research Finding

After studying literatures, we find that the work that does not came into the knowledge of researchers i.e. key points that are further work for research. Image authentication based on distributed source coding. In the distributed source coding we will focus on the Slepian Wolf coding and on the basis distributed source coding we will try to find out that whether image is tampered or not.

III. AUTHENTICATION ALGORITHM

As in the previous work that is in the base paper a Slepian-Wolf encoded quantized image projection is used for authentication of the image. The findings are:-

-The previously proposed method is not a capacity approaching.

-The previously proposed method cannot be used where highly reliable information is to be sending over a medium.

-The same noise is used for different images.

-The compression of the image is done which increases the probability of distortion of the image.

-Gray scale image is used for the image authentication.

A. Slepian-Wolf Theorem

The efficiency of the system is measured by the rates in encoded bits per source symbol of the compressed data streams that are output by the encoders.

The Slepian-Wolf Theorem specifies the set of rates that allow the decoder to reconstruct these correlated data streams with arbitrarily small error probability.

Although the theorem holds for much more general classes of inputs, the special case stated below is for two correlated data streams, X—1 , and X—2 , which are formed from making n independent drawings from a joint probability distribution $P(X1=x1,X2=x2)$ .As shown in Fig. 4, encoder 1, observes X—1 and then sends a message to the decoder which is a number from the set $\{1,2,\cdots,2nR1\}$. Similarly, encoder 2, observes X—2 and then sends a message to the decoder which is a number from the set $\{1,2,\cdots,2nR2\}$. The outputs from the two encoders are the inputs to the single decoder. The decoder, upon receiving these two inputs, outputs two n-vectors X—∗1 and X—∗2 which are estimates of X—1 and X—2 , respectively.

The systems of interest are those for which the probability that X—∗1 does not equal X—1 or X—∗2 does not equal X—2 can be made as small as desired by choosing n sufficiently large. Such a system is said to be an admissible system and the rate pair $(R1,R2)$ for an admissible system is said to be an admissible rate pair. The admissible rate region is the closure of the set of all admissible rate pairs.

The following entropies can be calculated for the pair of random variables $X1$ and $X2$ with joint probability distribution $P(X1=x1,X2=x2)$:

$H(X1,X2),H(X1|X2),H(X2|X1),H(X1)$

And $H(X2)$ when calculating entropies, all logarithms will be taken to the base 2.

B. Distributed Source Coding Using Binary LDPC Codes

Low density parity check codes can be used for applications involving compression of two correlated sources using the syndrome concept. The compressed sequence of the source output bits is the syndrome, which is determined using the parity check matrix H. It has been shown [2] that LDPC codes can be employed when viewing the problem using an equivalent channel and applying the syndrome approach for the case where one of the two correlated sources is available lossless at the joint decoder. This can be viewed as application of LDPC codes to a compression problem with side information. It is based on modifying the conventional message passing LDPC decoder to take into account the syndrome information.

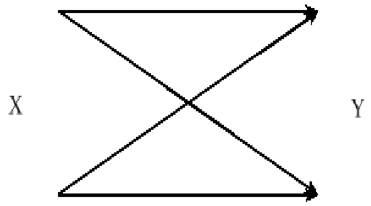


Fig. 4. Modeling Correlation using BSC.

Also all LDPC code design techniques can be applied to distributed source coding producing simulation results better than any turbo coding scheme. The system considered consists of two equiprobable binary random sources X=[X$_1$, X$_2$, ..., X$_n$] and Y=[Y$_1$, Y$_2$, ..., Y$_n$]. X_i and Y_i are correlated with $Pr[X_i \neq Y_i]=p<0.5$. The correlation between X_i and Y_i is modeled using a binary symmetric channel with crossover probability p. X_i being the input to the channel and Y_i being the output (figure 4.1), with the compressed version of X looking like a codeword to the channel. The source Y is available losslessly at the joint decoder and source X is compressed. The rate used for Y is its entropy $NR_2=NH(Y_i)=N$ bits, therefore the theoretical limit for lossless compression of X is [1] $nR_1 \geq nH(X_i/Y_i)=nH(p)=n(-p \log_2 p-(1-p)\log_2 (1-p))$. The compression results in mapping a sequence of N input bits into N-K syndrome bits resulting in a compression ratio of N: (N-K), known as the "Wyner's scheme" [13]. The all zero syndrome of the linear block code is considered to be the original code for distributed coding.

C. Encoding and Decoding with Binary LDPC Codes

Encoding: Encoding is done by multiplying the source X (length N) by the parity check matrix H, resulting in the compressed X which is the syndrome S (length (N-K)).

$$S=X.H^T$$

In the bipartite graph encoding can be viewed as the binary addition of all the variable nodes connected to the same check node. Considering an example of H matrix as shown below.

$$X_0 \;\; X_3 \;\; X_4 \;\; X_5 \;\; X_6 = S_0$$
$$X_1 \;\; X_3 \;\; X_5 = S_1$$
$$X_1 \;\; X_2 = S_2$$
$$X_0 \;\; X_2 \;\; X_4 \;\; X_6 = S_3$$
$$S=[S_1 \;\; S_2]^T$$

Here the vector is the compressed version of the source sequence ***X***. As the vector ***S*** is binary it consists of ones and zeros. When the value of S is zero the parity check equation obtained by multiplying the source ***X*** (length N) by the parity check matrix H is same as the parity check equation of low density parity check codes, which is obtained by multiplying the received vector with the H matrix for channel coding. When the value of S is one the parity check equation obtained by multiplying the source ***X*** (length N) has to satisfy value of S. This is taken care of in the horizontal step of the decoding algorithm.

D. Decoding

Decoding involves the estimation of the N-length sequence ***X*** from the (N-K) length syndrome S and the N-length sequence ***Y***. The decoding algorithm is similar to LDPC decoding used for channel coding expect for the inclusion of the syndrome bits in the horizontal step of the algorithm. The set of noise bits n that participate in check m are denoted by $N(m)=\{n{:}H_{mn}=1\}$. We also define the set of checks in which noise bit n participates, $M(n)=\{m{:}H_{mn}=1\}$.$N(m)\backslash n$ denotes the set of noise bits excluding the noise bit n. There are two quantities q_{mn} and r_{mn} associated with each non-zero element in H matrix that is alternatively updated iteratively. is the probability that noise bit n of ***X*** has the value a, given information obtained via checks other than the check m. is the probability of check m being satisfied if bit n of ***X*** is considered fixed at a with the other bits having separable distribution given by $\{q_{mn} : nN(m)\backslash n\}$.

E. Initialization

Considering a binary symmetric channel (BSC) with crossover probability p. We model the correlation between the two sources using the binary symmetric channel with X_i as the input to the channel and Y_i as the output. The compressed version of X is the syndrome S which is be made to look like a codeword of the channel.

$$p(y_j/x_j=0)=p^{yj}(1-p)^{1-yj}$$
$$p(y_j/x_j=1)=p^{1-yj}(1-p)^{yj}$$

F. Horizontal Step

The horizontal step involves the computation of two probabilities and , the probability of the observed values of S_m when $x_n=0$ and $x_n=1$ respectively, given the other bits $\{x_n{:}\; n \neq n\}$have a separable distribution given by the probabilities {, }, which is obtained by running through the checks m for each $nN(m)$. The syndrome information is included in this step to modify the message passing LDPC decoder for distributed source coding. Where is the probability of the observed value of S_m arising when x_n=0, arising when x_n=1, defined by the conditional probabilities in the summations are either zero or one, depending on whether the observed z_m matches the hypothesized values of x_n and the $\{x_n\}$. A convenient way to implement these probabilities is using the forward and backward passes with the products of the differences computed. Also , and hence and the syndrome which is the compressed information is included in the horizontal step in the calculation .

G. Vertical Step

In the vertical step the values and are used to update the probabilities and α_{mn} is chosen such that . These products can be efficiently computed in a downward pass and an upward pass. Also the pseudo posterior probabilities and at this iteration are computed using these quantities are used in the estimation of, which is the N-length estimate of X.

The horizontal and vertical steps are repeated for a given number of iterations after which the estimate , of the N-length message vector X is determined. The estimated is then compared with X and the bit error rate is calculated.

IV. RESULTS AND ANALYSIS

As according to the algorithm the complete system will work in two phases they are as follows:
-At Server Side
-At Client Side
For this first we are taking the same image or the unaffected image and in the next section we will take the image which is being affected.

A. Graphical User Interface for Unaffected Image

At the Server Side. With the help of GUI the working of LDPC Intrusion identification system has been shown in the above interfaces. The working of the system starts with a interface which will be having the two side authentication log in system. That is called as the server side and the client side the communication medium between the two will be a two state channel.

The working of the system starts with the authentication of the digital image. The authentication of the digital image is done on the server end firstly and on the other end its authenticity is check. The implementation of this work is done using MATLAB which is a high-performance language. As shown in the figure at the server end the user has to log in through a interface with the help of his user name and password respectively. The user name and password are again will the unique one for this system and if there is some security issues relating to the user name and password the administrator can change both the user name and password of the system. When the user log in successfully into the system another graphical user interface will pop up this will ask to browse the digital image from any database of any size, quality, format etc as revealed in the figure.

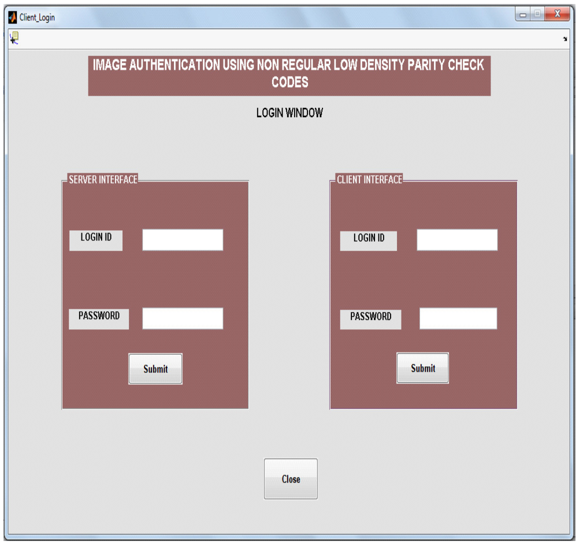
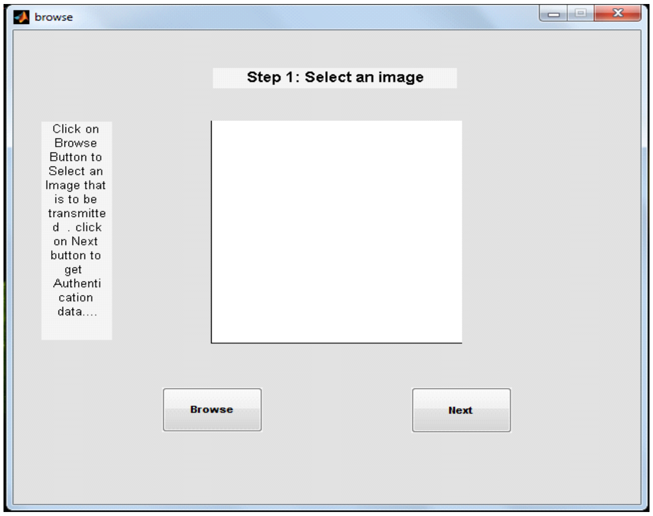


IMAGE AUTHENTICATION USING NON REGULAR LOW DENSITY PARITY CHECK CODES

LOGIN WINDOW

Fig. 5. Login Interface for the system.

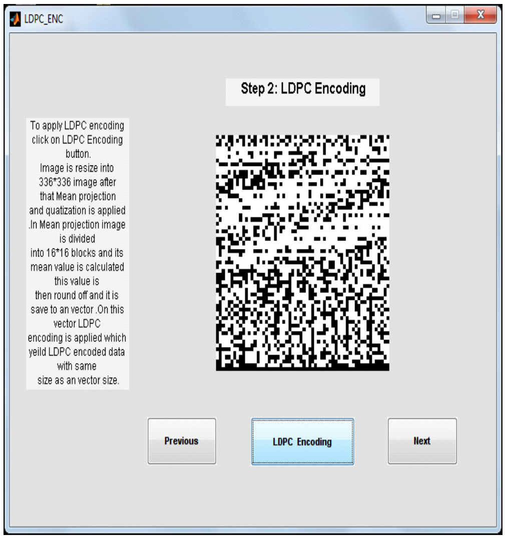


Step 1: Select an image

Click on Browse Button to Select an Image that is to be transmitte d . click on Next button to get Authenti cation data....

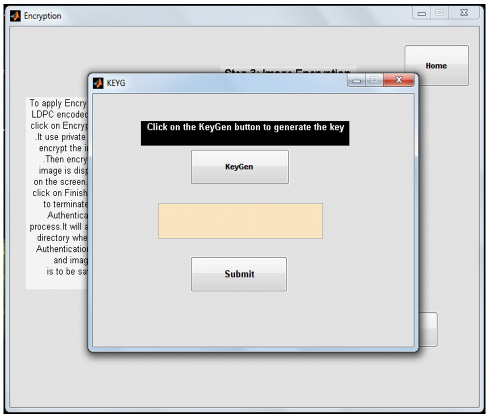
Fig. 6. Image Browsing for the server side.

In figure 5 the system graphical user interface is shown in the interface two login window can be seen one is for the server side and other is for the client side. After the ldpc encoding is done the user will click on the next button to encrypted the digital image depict in the figure 8. To apply Encryption on image the user has to click on Encrypt button. It use private key to encrypt the image.



To apply LDPC encoding click on LDPC Encoding button. Image is resize into 336*336 image after that Mean projection and quatization is applied .In Mean projection image is divided into 16*16 blocks and its mean value is calculated this value is then round off and it is save to an vector .On this vector LDPC encoding is applied which yeild LDPC encoded data with same size as an vector size.

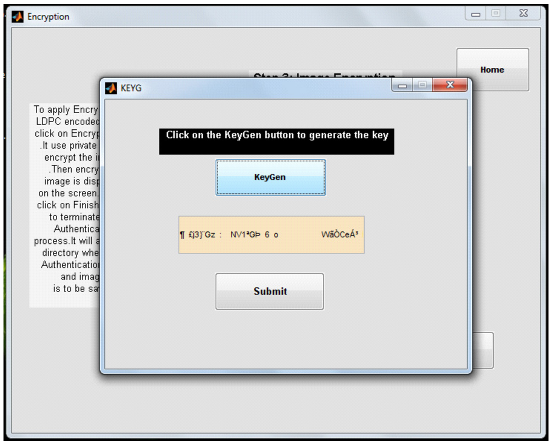
Fig. 7. LDPC Encoded Data generated.



To apply Encry
LDPC encoded
click on Encry
.It use private
encrypt the i
.Then encry
image is disp
on the screen
click on Finish
to terminate
Authentica
process.It will a
directory whe
Authenticatio
and imag
is to be sa

Click on the KeyGen button to generate the key

Fig. 8. Key Generation.



To apply Encry
LDPC encoded
click on Encry
.It use private
encrypt the i
.Then encry
image is disp
on the screen.
click on Finish
to terminate
Authentica
process.It will a
directory whe
Authenticatio
and imag
is to be sa

Click on the KeyGen button to generate the key

¶ £j3}¨Gz : NV1³GÞ 6 o WãÒCeÁ³

Fig. 9. Generated Key.

The key generation is shown in the figure 9. After putting in the key the encrypted image is displayed on the screen. At last click on Finish button to terminate the Authentication process. It will ask for a directory where the Authentication data and image is to be saved and the system will save the entire data in the zip file on the given location.

At the Client Side. At the other end that is at the client end when the digital image authenticity is to be checked the user has to login through the client side window depict in the figure 10. To apply LDPC decoding user will click on LDPC Decoding button than it will ask for inputting LDPC encoded data. On the LDPC encoded data decoding is applied, which give the LDPC decoded data as depict in the figure. On this vector LDPC decoding is applied which yield LDPC decoded data with same size as an vector size as revealed in the figure. To apply Decryption on image the user has to click on Decrypt button. It use private key to decrypt the image. The key generation is shown in the figure 11and 12. The authenticity of the system even increases when the user has to put in the same key as it is used for the encryption of the image. After putting in the key the decrypted image is displayed on the screen shown in the figure 13.

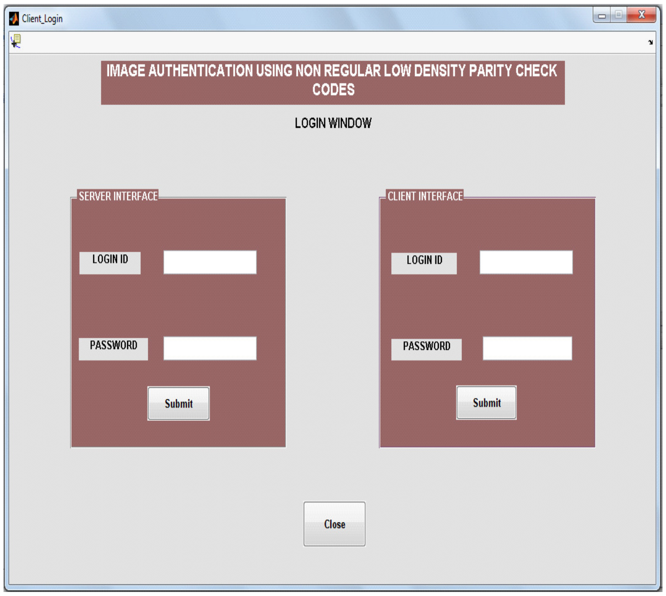
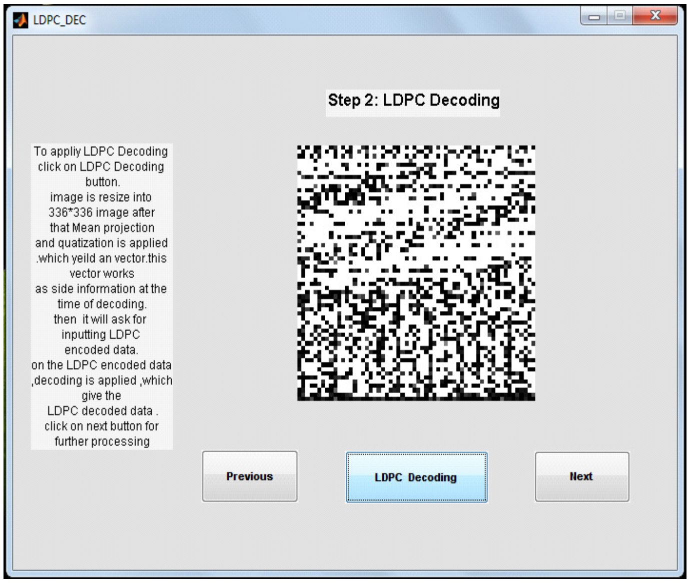
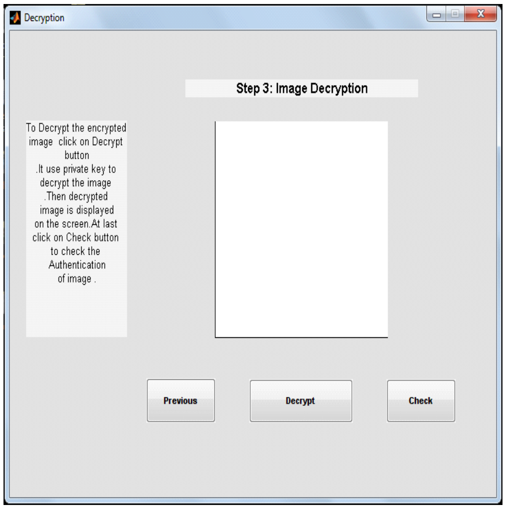


Fig. 10. Interface of the Image Authentication System.



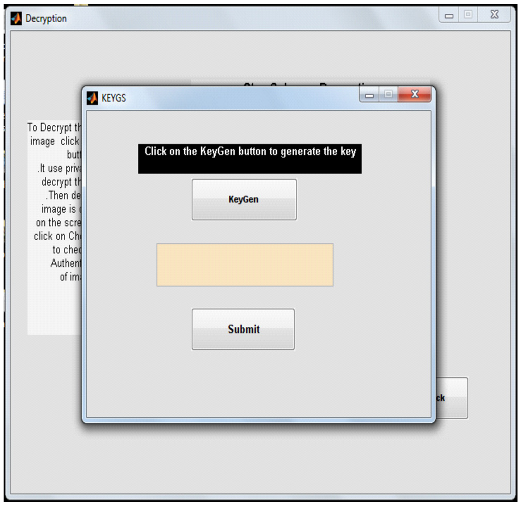
To appliy LDPC Decoding click on LDPC Decoding button.
image is resize into 336*336 image after that Mean projection and quatization is applied .which yeild an vector.this vector works
as side information at the time of decoding.
then it will ask for inputting LDPC encoded data.
on the LDPC encoded data ,decoding is applied ,which give the
LDPC decoded data .
click on next button for further processing

Fig. 11. Decoded Image.



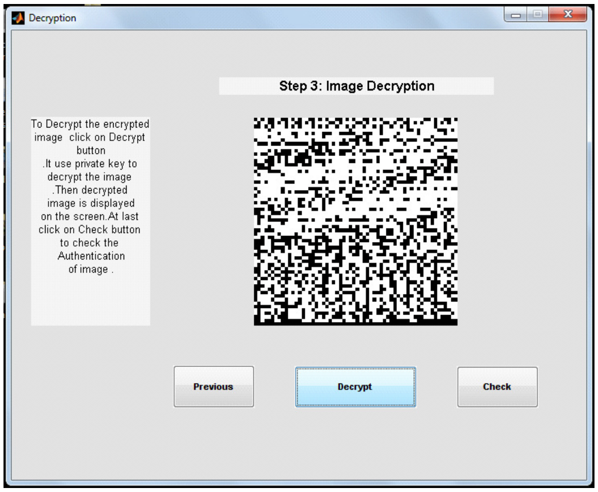
To Decrypt the encrypted image click on Decrypt button .It use private key to decrypt the image .Then decrypted image is displayed on the screen.At last click on Check button to check the Authentication of image .

Fig. 12. Decryption of the Image.



Click on the KeyGen button to generate the key

Fig. 13. Generation of Key for Decrypting the Image.

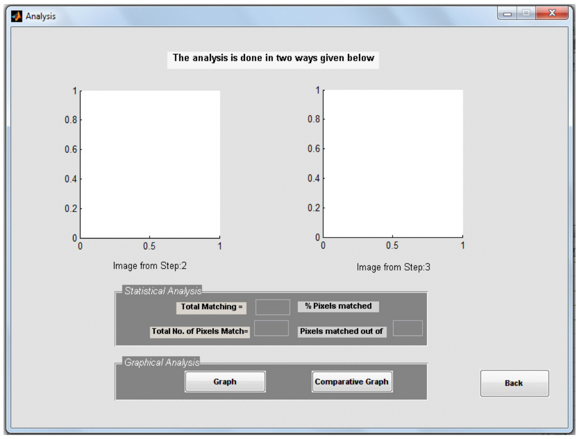


To Decrypt the encrypted image click on Decrypt button .It use private key to decrypt the image .Then decrypted image is displayed on the screen.At last click on Check button to check the Authentication of image .

Fig. 14. Decrypted Image.

Now the user has the option to check the authenticity of the image for this he has to choose the option of check than a window will pop up shown in the figure 15. The authenticity of the image is check on the basis of following two ways:
-Statistical Analysis
-Graphical Analysis

In Statistical analysis we will check the numbers of pixel match of the image outcome after the decoding of the image and after the decryption of the image. In graphical analysis we will check the authenticity of the image on the basis of gray level or intensity value and the size of the image.

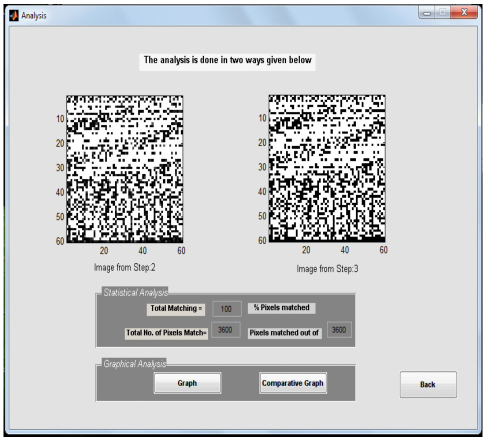


The analysis is done in two ways given below

Total Matching = % Pixels matched

Total No. of Pixels Match= Pixels matched out of

Fig. 15. Statistical analysis of the image.



The analysis is done in two ways given below

Total Matching = 100 % Pixels matched

Total No. of Pixels Match= 3600 Pixels matched out of 3600

Fig. 16. Statistical analysis output of the image.

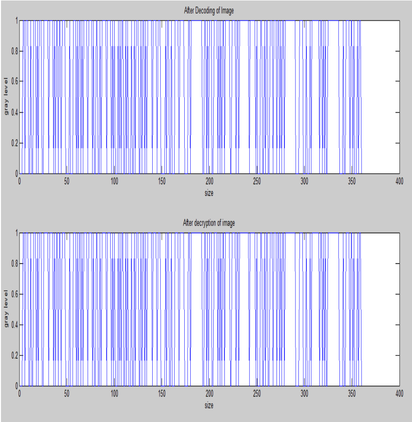


Fig. 17. For the unaffected image.

B. Graphical Analysis

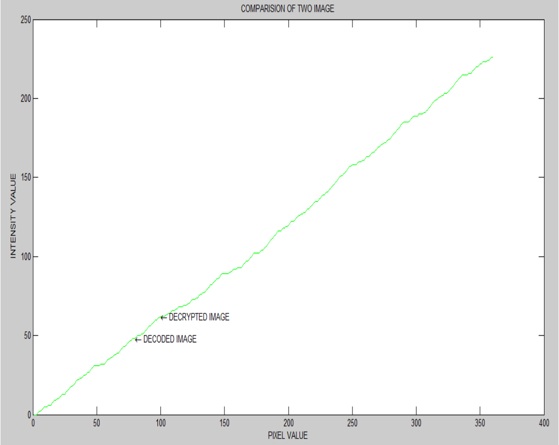
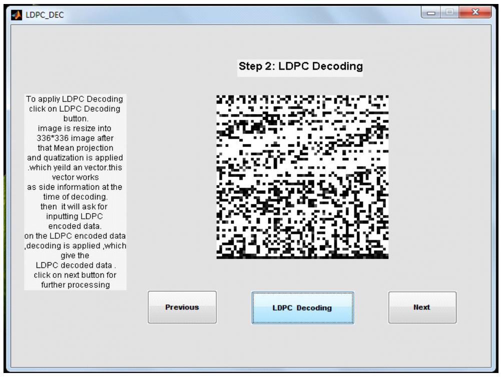


Fig. 18. Comparative graph for unaffected image.

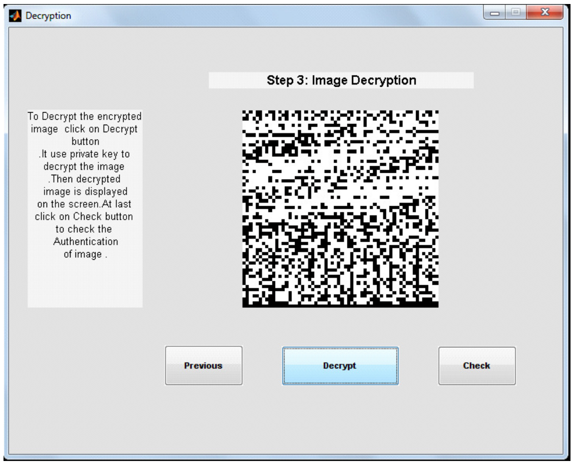
C. Graphical User Interface for affected or tempered Image

Consider the same digital image which was previously authenticated and its results shows that the image received is 100% authenticated one now some changes has been done with the same image with its properties. The server side will perform the same procedure for encoding and encryption for the affected digital image. The client side actions are shown here in this section because this part checks the authenticity of the image. As shown the image here is taken is the same image but it is being tempered now using this image the same process of the client side is repeated to check the authenticity of the image.



To appliy LDPC Decoding
click on LDPC Decoding
button.
image is resize into
336*336 image after
that Mean projection
and quatization is applied
.which yeild an vector.this
vector works
as side information at the
time of decoding.
then it will ask for
inputting LDPC
encoded data.
on the LDPC encoded data
,decoding is applied ,which
give the
LDPC decoded data .
click on next button for
further processing

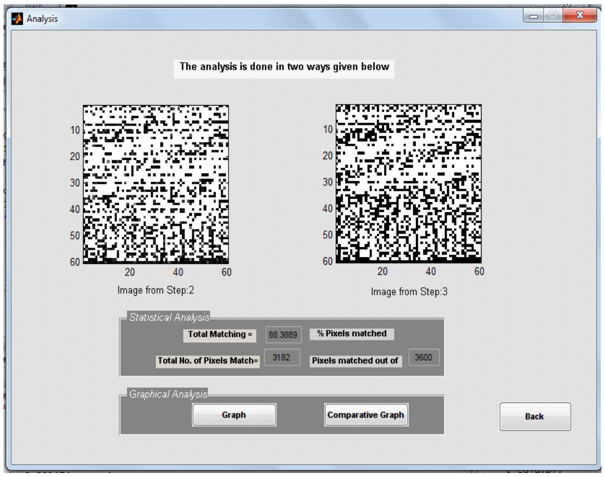
Fig. 19. The decoded image.



To Decrypt the encrypted
image click on Decrypt
button
.It use private key to
decrypt the image
.Then decrypted
image is displayed
on the screen.At last
click on Check button
to check the
Authentication
of image .

Fig. 20. Decrypted image.

D.. Statistical Analysis



The analysis is done in two ways given below

Image from Step:2

Image from Step:3

Statistical Analysis

Total Matching = 88.3889 % Pixels matched

Total No. of Pixels Match= 3182 Pixels matched out of 3600

Graphical Analysis

Fig. 21. Statistical Analysis for the tempered image.

Table 1: Comparative Result.

Comparative result		
Pixels match out of 3600	**For Original image**	**For Tempered image**
No. of pixel matched	3600	3182
No. of pixel unmatched	0	418
% of pixel matched	100%	88.38%

E. Graphical Analysis

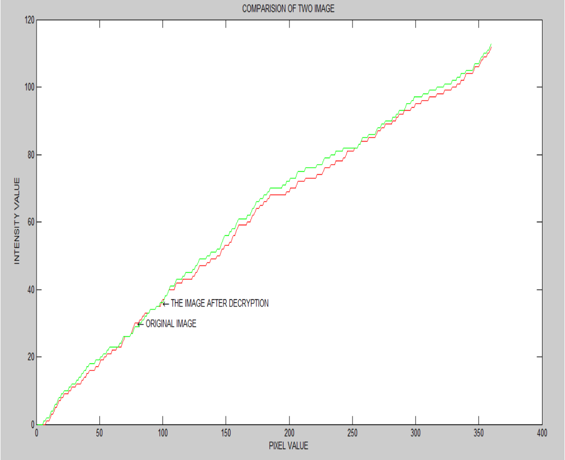


Fig. 22. Comparative graphs for the tempered image.

V. CONCLUSION

This paper investigated the robustness of the scheme for image authentication described in [4] and proposed several improvements. We obtained a robust algorithm for image authentication and recovery using distributed source coding. It is clear from the results in Figure 8 that the proposed scheme gives almost the same performance as an oracle decoder or a decoder using the EM algorithm for parameter estimation [7], using methods that have much less computational complexity.

This thesis presents and investigates a novel image authentication scheme that distinguishes legitimate encoding variations of an image from tampered versions based on distributed source coding and statistical methods. A two-state lossy channel model represents the statistical dependency between the original and the target images. Tampering degradations are captured by using a statistical image model, and legitimate compression noise is assumed to be additive white Gaussian noise.

The rates of falsely deemed tampered blocks can reach zero, while keeping the undetected tampered pixel rates at about 2%, since most of the blocks falsely deemed untampered have only a few pixels tampered. In most cases, 1D and 2D spatial models achieve a lower undetected tampered pixel rate at a given falsely deemed tampered block rate.

Slepian-Wolf coding that exploits the correlation between the original and the target image projections achieves significant rate savings. The Slepian-Wolf decoder is extended using expectation maximization algorithms to address target images that have undergone contrast, brightness, and affine warping adjustment. The localization decoder infers the tampered locations and decodes the Slepian-Wolf bit stream by applying the sum product algorithm over a factor graph which represents the relationship among the Slepian-Wolf bit stream, projections of the original image and the target image, and the block states. Spatial models are applied to exploit the spatial correlation of the tampering. Distributed source coding is an ideal tool for the image authentication problem in which the data sent for authentication are highly correlated to the information available at the receiver.

REFERENCES

[1]. R. Wolfgang and E. Delp, "A watermark for digital images," *International Conference on Image Processing, Proceedings., ,* vol. **3**, pp. 219–222 vol.3, Sep 1996.

[2]. R. W. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video,"*Proceedings of the IEEE*, vol. **87**, no. 7, July 1999, pp. 1108–1126.

[3]. C.S. Lu, C.-Y. Hsu, S.-W. Sun, and P.-C. Chang, "Robust mesh-based hashing for copy detection and tracing of images," *IEEE International Conference* on *Multimedia and Expo, 2004. ICME '04. 2004*, vol. **1**, pp. 731–734 Vol.1, June 2004.

[4]. Y.C. Lin, D. Varodayan, and B. Girod, "Image authentication based on distributed source coding," *IEEE International Conference on Image Processing,* 2007. ICIP 2007., vol. **3**, pp. III –5–III –8, 16 2007-Oct. 19 2007.

[5]. D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," Information Theory, *IEEE Transactions on*, vol. **19**, no. 4, pp. 471–480, Jul 1973.

[6]. Y.C. Lin, D. Varodayan, and B. Girod, "Image authentication and tampering localization using distributed source coding," *IEEE 9th Workshop on Multimedia Signal Processing*, 2007. MMSP 2007., pp. 393–396,Oct. 2007.

[7]. Y.C. Lin, D. Varodayan, T. Fink, E. Bellers, and B. Girod, "Authenticating contrast and brightness adjusted images using distributed source coding and expectation maximization," *IEEE International Conference on Multimedia and Expo*, ICME 2008, Hannover, Germany, June 2008.

[8]. "Localization of tampering in contrast and brightness adjusted images using distributed source coding and expectation maximization," *IEEE International Conference on Image Processing, 2008. ICIP 2008.* 15th, pp. 2204–2207, Oct. 2008.

[9]. W. E. Ryan. (1997) A turbo code tutorial. [Online]. http://www.ece.arizona.edu/ryan/publications/turbo2c.pdf

[10]. J. Garcia-Frias, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters,* vol. **5**, no. 10, pp. 417–419, Oct 2001.

[11]. P. Dong, J. Brankov, N. Galatsanos, Y. Yang, and F. Davoine, "Digital watermarking robust to geometric distortions," *IEEE Transactions on Image Processing*, , vol. **14**, no. 12, pp. 2140–2150, Dec. 2005.