International Journal of Engineering & Technology, 7 (4.10) (2018) 59-63

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Image Encryption Using Orthogonal Hill Cipher Algorithm

K. Madhusudhan Reddy[1*]
Anirudh Itagi[2], Saransh Dabas[3] and Bonam Kamala Prakash[4]

[1]Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology, Vellore, 632014, India
[2,3,4]Vellore Institute of Technology, Vellore, 632014, India
** Corresponding author E-mail: drkmsreddy@yahoo.in*

Abstract

The Hill Cipher algorithm is a practical algorithm which requires only preliminary knowledge of matrices which makes it quite useful. But, for decrypting the encoded information the key matrix needs to have an inverse (The key matrix needs to be invertible) or else it is impossible to decrypt the given information. The main objective of our paper is to introduce an unorthodox and improved version of Hill Cipher which will make encryption and decryption of an image easier. This paper has diverged from the conventional method by using an orthogonal matrix (where the transpose of a matrix is equal to its inverse) as our key matrix. This method is faster and easier as finding the transpose of a matrix is much simpler than finding the inverse of a given matrix and hence, facilitating the process of faster execution. To further break down the process into an even simpler one, we have added a few more constraints. To show its application, we have used this method to encrypt images using MatLab.

Keywords: *Orthogonal Matrix, Hill Cipher, Invertibility, Encryption.*

1. Introduction

In this era, there have been many breakthroughs in digital communication and modern means of digital transfer of information is constantly replacing the conventional means thus increasing the emphasis on security and safe transfer of information, thus making cryptography crucial. Conventional Encryption can also be called as symmetric encryption or single key encryption. In fact, it was a type of encryption that was used before the development of public-key encryption. Conventional encryption can, in turn, be classified into the categories of classical and modern techniques. The security of the information has always been an important problem but its importance has been increasing with each passing day. Cryptographic algorithms can be divided into Symmetric and Asymmetric key algorithms. In symmetric algorithms, the receiver and the sender share a common key.

Asymmetric algorithms deal with a pair of keys (both public, private) which happen to be related mathematically. The primary polygraph cipher having specific advantages in data encryption is the Hill Cipher. These days, information security is beginning to play a major role in transmission of data as well as in its storage. Images are used in a wide range of processes which increases the importance of protection of image data from unauthorised access. The field of information concealment, image hiding or encrypting methods and algorithms employs image encryption to fulfil an important role in their processes.

Cryptography uses intricate mathematical calculations and logic for its process of encryption. Cryptography can also be thought of as an art, it restores people's faith in the electronic world by helping them to remain safe. People can carry on their online processes without the fear of their data being stolen or getting hacked and with the knowledge that their privacy is secure. The recent times have seen an upsurge of online transactions by people which calls for more encrypted and safer means and cryptography serves that purpose. The world today thrives on electronic transactions online by using online banking, payment sources like Paytm, Paypal, etc., This increase in using the online resources has resulted in increased dependence on cryptography and authentication. The transfer of information, while making sure that nobody understands it, is the purpose of cryptography. Here, the attempt is to make sure that when the enemy receives the message, it is in encrypted format so that he doesn't understand it. Confidentiality is the often considered as the main premise in the field of information shielding. Secure communication is one of the direct applications of cryptography. [1]

The central management problem keeps crippling the secure communication from becoming common and thus reducing their effectiveness. The production of public-key cryptography creates a large and an effective group of people who can communication securely with one another even if they had never communicated before.

In visual cryptography, an attacker cannot judge or find relevant information but can alter it without knowing the repositories of data which can create problems in decryption. There may be some bit flipping or bit alterations during communication also. To overcome this problem people have proposed many ways to detect the tampering [2]. The hill cipher is a classical symmetric cipher based on matrix manipulation. It has several advantages including its resistance to frequency analysis and its simplicity due to the fact that it uses matrix multiplication and inversion for encryption and decryption. However, it succumbs to the branded plaintext attack [3-4]. Substitution Cipher is a technique that transforms a given text into random or stochastic data bits. There are many cryptographic substitution algorithms like Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Polyalphabetic Ciphers, One Time Pad, etc. [5-6]

In this paper, we have used an improved version Hill Cipher method which uses an orthogonal matrix as its key matrix. The

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

primary aim of this paper has overcome the problems that arise when using a random matrix as a key matrix because if the key matrix is not invertible, then the message cannot be decrypted. To further simplify the calculations that are involved in calculating the inverse of a key matrix, we have used an orthogonal matrix since finding a transpose is much simpler than finding the inverse. We have also added a few more constraints to make the computation simpler.

2. Generation of Orthogonal Key Matrix

For the encryption technique, the Orthogonal Hill (OTH) Cipher algorithm that has proposed uses orthogonal matrix as the key matrix. The procedure that has adopted to generate the orthogonal matrix is elaborated below

The matrix A is said to be orthogonal matrix if $A A^T = I$ or transpose of A is equal to its inverse. The structure that has been followed for the generation of orthogonal key matrix is valid for matrix whose elements are real number. The algorithm that we have proposed can be used for square matrices with order 'n' provided that 'n' is even. We have illustrated it by taking the example of a 4x4 matrix.

$$\text{Let A} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} \quad (1)$$

$$a_{11} = \begin{bmatrix} a & b \\ e & f \end{bmatrix}, \; a_{12} = \begin{bmatrix} c & d \\ g & h \end{bmatrix}, \quad (2)$$

$$a_{21} = \begin{bmatrix} i & j \\ m & n \end{bmatrix}, \; a_{22} = \begin{bmatrix} k & l \\ o & p \end{bmatrix} \quad (3)$$

$$\text{Therefore, A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad (4)$$

$$\text{Which implies } A^T = \begin{bmatrix} a_{11}^T & a_{21}^T \\ a_{12}^T & a_{22}^T \end{bmatrix} \quad (5)$$

A is an orthogonal matrix which implies $A A^T = I$
Or $A^T = A^{-1}$

Consequently,

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} * \begin{bmatrix} a_{11}^T & a_{21}^T \\ a_{12}^T & a_{22}^T \end{bmatrix} = I_{4x4} \quad (6)$$

Therefore,

$$a_{11} a_{11}^T + a_{12} a_{12}^T = I \quad (7)$$

$$a_{11} a_{21}^T + a_{12} a_{22}^T = 0 \quad (8)$$

$$a_{21} a_{11}^T + a_{22} a_{12}^T = 0 \quad (9)$$

$$a_{21} a_{21}^T + a_{21} a_{22}^T = I \quad (10)$$

These are the conditions which are arrived at using the property of the orthogonal matrices but this isn't simple enough to carry out analysis. That is why some more constraints have been added to simplify the process.

Constraint: $a_{11} = a_{12}$
Which results in $a_{22} = -a_{21}$

$$\text{Now A} = \begin{bmatrix} a_{11} & a_{11} \\ -a_{22} & a_{22} \end{bmatrix} \quad (11)$$

$$\text{And, A } A^T = I \quad (12)$$

$$\begin{bmatrix} 2*a_{11}*a_{11}^T & 0 \\ 0 & 2*a_{22}*a_{22}^T \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (13)$$

Equating it,
$2\, a_{11}\, a_{11}^T = I = 2\, a_{22}\, a_{22}^T$

$$a_{11} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \text{(say)}$$

Simplifying it, we get the expressions for a, b, c and d

$$a = \frac{2*b*c*d}{2*d^2 - 1} \quad (14)$$

$$b = \frac{2*a*d*c}{2*c^2 - 1} \quad (15)$$

$$c = \frac{2*a*d*b}{2*b^2 - 1} \quad (16)$$

$$d = \frac{2*a*b*c}{2*a^2 - 1} \quad (17)$$

By substituting values for a, b, c, d which satisfy the above equation we are able to generate a_{11} and similarly a_{22}.
Hence by following this procedure, the key matrix can be generated, thus eliminating the complexity involved in the regular Hill Cipher method.

Algorithm:

1. Select any arbitrary 2x2 matrix a_{11}(It will be one of the blocks of the main matrix)
2. Generate the second block which is the same as that of the first block($a_{11} = a_{12}$)
3. Select another arbitrary matrix (this will be the third block), a_{21}
4. Find the final block by using $a_{22} = -a_{21}$
5. By combining all the blocks, the required key matrix is obtained.

Fig. 1 illustrates the Flow diagram for the Orthogonal Matrix generation,

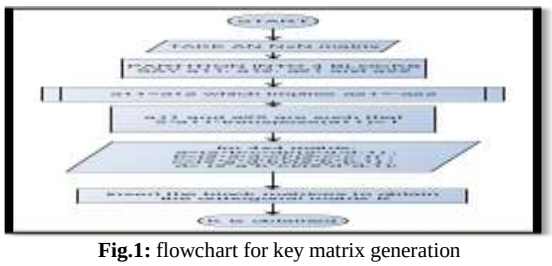


Fig.1: flowchart for key matrix generation

3. Image Encryption Using OTH Cipher

We know that the original Hill Cipher method can be used for encrypting both grey scale and color images, the procedure that we have proposed can also be adopted for grey scale and color images and we have also verified the results using MatLab. For the color images, we can convert the whole image into a three dimensional matrix by splitting the image into three different colors namely red, blue and green. We use the key matrix that we have generated earlier and we encrypt the given matrix.

The Encryption Algorithm:

1. Import the image onto MatLab
2. Assign a variable to it (say b)
3. b is a Three Dimensional Array, whose constituents hold the RGB (Red Green Blue) value of the Pixels of the image.
4. Split b into its 3 different matrices say b1, b2 and b3,using:
   b1=b(:,:,1);
   b2=b(:,:,2);
   b3=b(:,:,3);
5. It is noteworthy to remember that the values inside these matrices are in "uint8" format. That is, they range from 0-255 only and are not subject to any matrix operation.
6. Convert them to "double" format matrices: bb1=double(b1); bb2=double(b2); bb3=double(b3);
7. Now apply the OTH cipher key to these matrices. The K obtained here is orthogonal and obtained from the above mentioned algorithmbbe1=K*bb1; bbe2=K*bb2; bbe3=K*bb3;
8. The new matrices obtained are encrypted and can now be inserted back into the Pixel format, after converting them back to uint8 format.
   To convert them back to uint8:
   bbeu1= uint8 (bbe1);
   bbeu2= uint8 (bbe2);
   bbeu3= uint8 (bbe3);
9. These are the new RGB values of the encrypted image.
10. To view the image, insert them back into the Array form:
    A=bbeu1 (say)
    A(:,:,2)=bbeu2;
    A(:,:,3)=bbeu3;
11. To view the encrypted image:
    Image(A)
12. Thus the encrypted image is obtained.
    In order to get back the original image, the same procedure is to be followed, only change being that K is replaced by KI = inv(K) = K^{-1}

Flowchart:

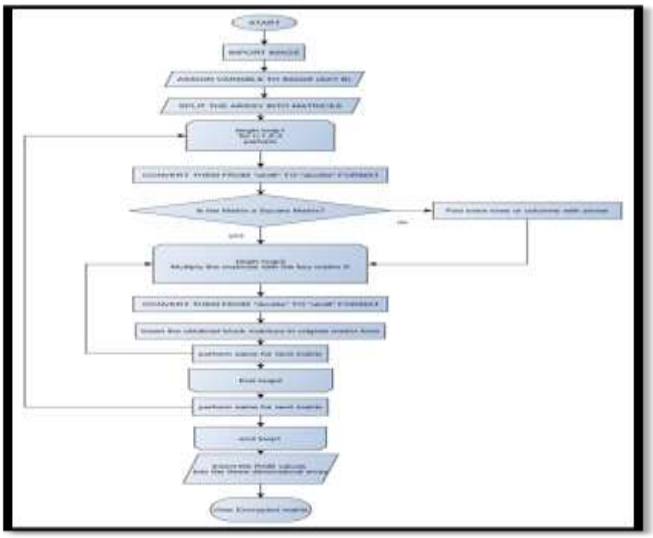


Fig.2: the Encryption Algorithm flowchart

4. Simulation

Generation of the key matrix using Gram-Schmidt orthogonalisation:

We use the projection formula as:

$$proj_{\vec{u}}(\vec{v}) = \frac{\langle \vec{v}, \vec{u} \rangle}{\langle \vec{u}, \vec{u} \rangle} \vec{u} \tag{18}$$

Where $\langle \vec{v}, \vec{u} \rangle$ is the inner product of vectors $\vec{v}$ and $\vec{u}$. Now, we take set of basis elements n, n being the dimension of pixels we want to encrypt as nxn matrix. Let this set of basis elements be $\{\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, ..., \vec{v}_n\}$. Then the set of orthonormal basis is calculated as $\{\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4, ..., \vec{e}_n\}$ as:

$$\vec{u}_1 = \vec{v}_1;\ \vec{e}_1 = \frac{\vec{u}_1}{\|\vec{u}_1\|} \tag{19}$$

$$\vec{u}_2 = \vec{v}_2 - proj_{\vec{u}_1}(\vec{v}_2);\ \vec{e}_2 = \frac{\vec{u}_2}{\|\vec{u}_2\|} \tag{20}$$

$$\vec{u}_3 = \vec{v}_3 - proj_{\vec{u}_1}(\vec{v}_3) - proj_{\vec{u}_2}(\vec{v}_3)\ \vec{e}_3 = \frac{\vec{u}_3}{\|\vec{u}_3\|} \tag{21}$$

So on till,

$$\vec{u}_n = \vec{v} - \sum_{j=1}^{n-1} proj_{\vec{u}_j}(\vec{v}_j);\ \vec{e}_n = \frac{\vec{u}_n}{\|\vec{u}_n\|} \tag{22}$$

Now these orthonormal basis vectors $\{\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4, ..., \vec{e}_n\}$ are then placed in an n × n matrix and this matrix is then used as encryption matrix for encrypting the image information by distorting its pixels by multiplying the pixel matrix with this matrix. While coding in C programming language we are working with two matrices, "matA" and "matQ1". First the code asks for dimension of the matrix and then matA matrix takes the input values from the user as basis elements $\{\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, ..., \vec{v}_n\}$ using 'inputMatrix' function and then proj and normalize functions calculate the projection and normalized values respectively and finally these normalized values are put in n × n matrix to get the orthogonal matrix "matQ1".

```
void inputMatrix(double* matrix,int m,int n){
  int i,c=m*n;
  for(i=0;i<c;i++)
    scanf("%lf",matrix+i);
}
```

(a)

```
double dot(double a[],double b[],int dimen){
  int i;
  double ans=0;
  for(i=0;i<dimen;i++)
    ans+=a[i]*b[i];
  return ans;
}
```

(b)

```
void proj(double a[],double e[],int dimen,double r[]){
  double q=dot(e,a,dimen)/dot(e,e,dimen);
  multiply(e,q,dimen,r);
}
```

(c)

```
void proj(double a[],double e[],int dimen,double r[]){
  double q=dot(e,a,dimen)/dot(e,e,dimen);
  multiply(e,q,dimen,r);
}
```

(d)

```
void multiply(double x[],double a,int dimen,double r[]){
  int i;
  for(i=0;i<dimen;i++)
    r[i]=x[i]*a;
}
```

(e)

```
void clone(double a[],int dimen,double r[]){
  int i;
  for(i=0;i<dimen;i++)
    r[i]=a[i];
}
```

(f)

```
void add(double a[],double b[],int dimen,double r[]){
  int i;
  for(i=0;i<dimen;i++)
    r[i]=a[i]+b[i];
}
```

(g)

```
void QRComputeQ(double* x,int m,int n,double* Q){
  int i,j;
  double *ai,*ui,*ej;
  double u[m][n],tmp[n];
  for(i=0;i<m;i++){
    ai=x+i*n;
    ui=u[i];
    clone(ai,n,ui);
    for(j=0;j<i;j++){
      ej=Q+j*n;
      proj     (ai,ej,n,tmp);
      multiply(tmp,-1,n,tmp);
      add      (ui,tmp,n,ui);
    }
    normalize(ui,n,Q+i*n);
  }
}
```

(h)

Fig.3: Software simulation: (a) Taking input values, (b) Calculating the dot product, (c) Calculating the projection, (d) Normalization, (e) Multiplication needed by 'proj' function, (f) Temporary value storage 'r' matrix, (g) Addition of two matrices and (h) Computation of Q matrix (orthogonal matrix)

The steps used to generate the matrix:

1. Taking input values: Figure.3 (a)
2. Calculating the dot product: Figure.3 (b)
3. Calculating the projection: Figure.3 (c)
4. Normalization: Figure.3 (d)
5. Multiplication needed by 'proj' function: Figure.3 (e)
6. Temporary value storage 'r' matrix: Figure.3 (f)
7. Addition of two matrices: Figure.3 (g)
8. Computation of Q matrix (orthogonal matrix): Figure.3 (h)

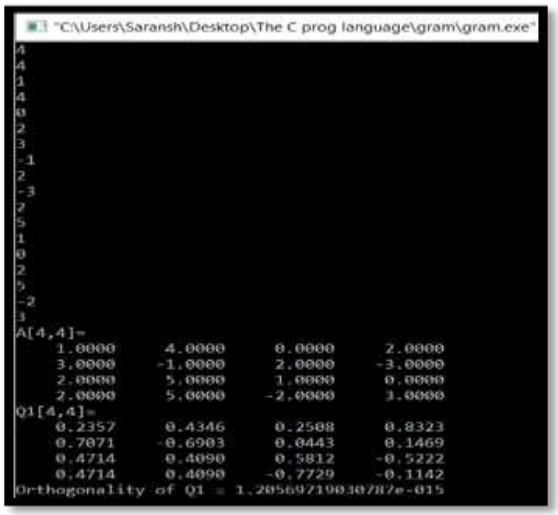
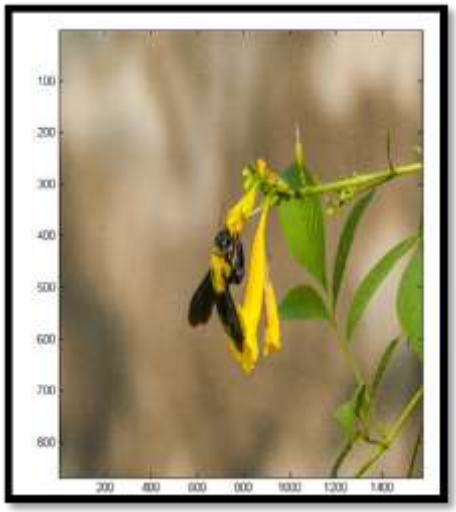


Fig.4: Sample example for 4×4 matrix

5. Results

Before encryption, the original images: Figure. 5(a) and 5(b):



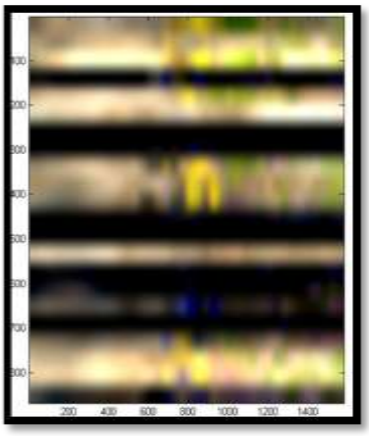
(a)



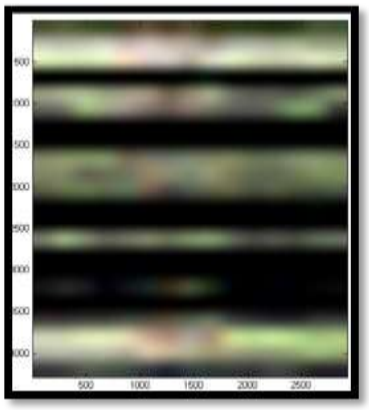
(b)

Fig.5: Images before encryption (a) original image1 and (b) original image2

After Encryption by application of OTH Cipher:



(a)



(b)

Fig.6: Images after encryption (a) encrypted image of image1 and (b) encrypted image of image2

Clearly, the encryption algorithm has encrypted the images beyond recognition or beyond any resemblance to the original image. Concretely, the matrix obtained by the above mentioned method of generation orthogonal matrix has been used in the OTH cipher to encrypt the images.
Fig.6(a) and Fig.6(b) illustrates how the algorithm has changed the pixel values of the images resulting in them appearing altered and making them unable to represent any visual meaning.
An important observation here is that the encryption algorithm is only as effective as more abstract or more random the orthogonal matrix may appear.

6. Conclusion

This paper has delineated on how to properly encrypt images by using OTH Cipher technique. The proposed encryption mechanism considerably reduces the problems faced while encrypting images using original Hill Cipher Technique. In this paper, a unique algorithm has been developed for generation of key matrix and it also elaborates on generation of key matrix using Gram-Schmidt orthogonalisation by means of the software code that has been written. Further, the images that have been encrypted substantiate the algorithms that have been mentioned earlier.

References

[1] Stallings, William. Network security essentials: applications and standards. Pearson Education India, 2007.
[2] Stallings, William, and Mohit P. Tahiliani. Cryptography and network security: principles and practice. Vol. 6. London: Pearson, 2014.
[3] Feistel, Horst. "Cryptography and computer privacy." Scientific american 228.5 (1973): 15-23.
[4] Acharya, Bibhudendra, et al. "Image encryption using advanced hill cipher algorithm." International Journal of Recent Trends in Engineering 1.1 (2009).
[5] Sastry, V. U. K., and N. Ravi Shankar. "Modified Hill Cipher with Interlacing and Iteration 1." (2007).
[6] Acharya, B., Jena, D., Patra, S. K., & Panda, G. (2009, January). Invertible, Involutory and Permutation Matrix Generation Methods for Hill Cipher System. In Advanced Computer Control, 2009. ICACC'09. International Conference on (pp. 410-414). IEEE.
[7] Hill, Lester S. "Concerning certain linear transformation apparatus of cryptography." The American Mathematical Monthly 38.3 (1931): 135-154.
[8] Hill, Lester S. "Cryptography in an algebraic alphabet." The American Mathematical Monthly 36.6 (1929): 306-312.
[9] Ganesan, K., and R. Anandan. "Version Control using Cryptographic Access Control." In Information Technology, 2006. ICIT'06. 9th International Conference on, pp. 192-196. IEEE, 2006.
[10] Sastry, V. Umakanta, N. Ravi Shankar, and S. Durga Bhavani. "A Modified Hill Cipher Involving Interweaving and Iteration." IJ Network Security 10, no. 3 (2010): 210-215.
[11] Schneier, Bruce. Applied cryptography: protocols, algorithms, and source code in C. john wiley & sons, 2007.
[12] Hori, G. (2011, May). Natural gradient approach in orthogonal matrix optimization using cayley transform. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on (pp. 2116-2119). IEEE.
[13] Sastry, V. U. K., Murthy, D. S. R., & Bhavani, S. D. (2010). A Block Cipher Having a Key on One Side of the Plain Text Matrix and its Inverse on the Other Side. International Journal of Computer Theory and Engineering, 2(5), 805.
[14] Ismail, I. A., Mohammed Amin, and Hossam Diab. "How to repair the Hill cipher." Journal of Zhejiang University-Science A 7.12 (2006): 2022-2030.
[15] Khalaf, A. A., El-karim, M. S. A., & Hamed, H. F. (2015, July). Proposed triple hill cipher algorithm for increasing the security level of encrypted binary data and its implementation using FPGA. In Advanced Communication Technology (ICACT), 2015 17th International Conference on (pp. 454-459). IEEE.