VISUAL ENCRYPTION BY PIXEL PROPERTY PERMUTATION

Ayush Surana
Department of CSE
SRM University
Chennai

Dr. Annapurani .K
Department of CSE
SRM University
Chennai

Tazeen Ajmal
Department of CSE
SRM University
Chennai

ABSTRACT

Data Security is a challenging problem in the sphere of Storage and transmission of Data. With the development of heavy computational resources various encryption algorithms are getting more prone to subsequent attacks or threats either via Brute Force, Cipher Attacks or Statistical Attacks and Side Channel Attacks. Thus, for better security there is a need for a more robust Encryption of Data especially unstructured data. This paper has tried to propose A Novel Image encryption technique known as Pixel Property Permutation(PPP) on the basis of Visual Cryptography. In this algorithm any image RGB or Grayscale is first converted to Black and White and compressed to (200*200) pixels. Although this is not necessary and one can use the RGB Image of varying Size. This Guideline is assumed only for simplicity purposes. The pixel values of the image are extracted and stored. These pixel values are then permuted with one another using random permutations. Thus, the encrypted image is obtained. Now, using the encrypted image and the original image a key is generated (superimposing one on the other). The key and the encrypted image are both subjected to the 128-Bit AES Encryption.[1] The decryption process is also very similar. The two AES encrypted images is decrypted using the same key obtained earlier. Now using the cipher key and the AES-decrypted image and overlapping the two the original image is recovered. This process of PPP uses AES encryption algorithm as an added layer of defense which is not essentially necessary but encouraged, this technique is cost effective and can be used to secure highly sensitive data.

1. INTRODUCTION

In this Data Extensive World there are huge amount of data available for various purposes. Accessing these data has become increasingly easy and important. Every corporation wants to secure their data in the most prolific manner possible. In this modern era with the development and the vast influence of the internet makes it highly difficult for enterprises to maintain their information as confidential. The secrecy of the data has to be maintained both at the senders and the receivers end. One such method to secure data is Cryptography. It is the mechanism in which one can study the various mathematical techniques involved in the sphere of securing information and data.Visual Cryptography is the type of cryptography to be applied for Visual Authentication and identification of images both normal and digital. Visual Cryptography is the new technique that uses simple mathematical models for its purposes unlike heavy highly complex models used by the traditional Cryptography. Here, the fundamental principle is to encrypt the data in such a way that the decryption can be performed by the human visual systems.This simple technique of Visual Cryptography Systems(VCS) encrypts an image in different shares and uses these images to give back the original image by superimposing them.

There are many traditional and highly complex encryption algorithms available to us like Blowfish, Twofish, Triple DES, IDEA, AES etc. This paper has tried to use the Advanced Encryption Standard (AES), recognised by the U.S National Institute of Standards and Technology (NIST) in 2001 and is among the most secure algorithm available to us using the Rijndael Cipher to encrypt the different shares (Key and the PPP Image) obtained after performing the PPP Encryption Process. However the AES algorithm is no longer the most secure. There are several methods like Cache timing attacks [2], and Biclique Cryptanalysis [3].

This paper has been dissected as described ahead. Section 2 describes the related works on the topic of Visual Cryptography. Section 3 gives the overview of the proposed algorithm Pixel Property Permutation. Section 4 deals with the in-depth execution of the algorithm and all of its major steps. Section 5 deals with the the various different Cryptanalytic methods available to us to show the robustness of the proposed algorithm. Section 6 gives the streamline of the simulation result when the proposed algorithm was executed. The last section gives a brief summary and future prospects and improvements that can be made on this project which if implemented would immensely be useful for keeping our precious data secure.

2. RELATED WORKS

Here we will discuss some of the works/ideas we studied from the various Research Papers and Illustrations. These Ideas have been classified under suitable sub headings as illustrated below: -

2.1 Chaos based Image EncryptionS.El Assad, M. Farajallah, C. Vladeanu [4], proposed a chaos based block cipher technique for Visual Encryption In their proposed method they have built a chaos based generator in which a Plain Image is sent to a Substitution/ Diffusion chamber and through that chamber it is sent to the 2D Cat Map which is highly efficient in generating a Secret Key from the aforesaid chaotic generator. This in turn gives us the Cipher Image.

2.2 DES and AES Encryption Algorithm

Zarko S. Stanisavljevic of Bulgaria, Serbia in his paper on Data Encryption Standard [5] Visual Representation has illustrated a method for visual representation using the DES Method. The prescribed method was implemented in the COALA System (CryptOgraphic ALgorithms visual simulAtion) It is designed for the better understanding of the Cryptographic Algorithms. The Specifications about these kinds of systems are thus highly useful in understanding the efforts and effects of the Coala System on the young Minds.

2.3 Visual Transformation based Algorithm

Xiaowei Xu, Scott Dexter and Ahmet M. Eskicioglu [6] present a hybrid image protection scheme to establish a relation between the data encryption key and the watermark. This schemes completes in three steps, 1. Generate shares of the image, and then embade watermark to this image share. Finally this embedded image has to be encrypted with any cipher scheme to get final encrypted image.

2.4 Value Transformation based Permutation

Aloka Sinha and Kehar Singh [7] have proposed a new technique.In this technique the digital signature of the original image is added to the encoded version of the original image. An error code that is best suited is followed to do the encoding of the image, ex: Bose-Chaudhuri Hochquenghem (BCH) code.After decryption of that image at the receiver’s end the digital signature verifies the authenticity of the image.

3. PROPOSED SYSTEM AND ITS MERITS---PIXEL PROPERTY PERMUTATION(PPP)

The System proposed in this Project deals with Visual Cryptography on the basis of Pixel Property Permutation(PPP). This project deals with the idea of extracting the Pixels of an Image as a digital image is read and or recognized by its pixel values alone. This Image is then Converted into Black and White and compressed to 200*200 Pixels. The new Image obtained is stored and its pixel values are randomly permuted into various transparent shares. Each share combined constitutes into an image. This image along with the original image is used to find the Key for Decryption Process. This key along with the encrypted image is processed through the infamous AES Encryption Technique. The Keys of the images encrypted is stored.

These keys are used to decrypt the Cipher key and the encrypted Image. These two images are superimposed to each other to give the resultant image. The resultant image passes through some denoising algorithms to give the original image.

This seemingly complex process has many merits as over the existing systems as has been illustrated below: -

- If trying to brute force this Encryption Algorithm which has two AES Encrypted Images along with a PPP Encrypted Image it will take nearly (3.8*1018+ 3.8*1018 (for two 128-Bit AES) and (200*200!) permutations to counter it. In retrospect it would take nearly 2 Billion Billion Years to Break it by the Fastest Computer. Hence it is extremely secure

- It is very easy to implement. If one uses only PPP for encryption even then it would be highly secure and would consume very little time and space complexity

- It is highly portable and can be implemented on any platform irrespective of the system it is being implemented on

- The space it consumes is also not very large as after compressing the image the size of the image is considerably reduced and this image is used for further illustrations.

- The PPP can be implemented using JAVA / C/ C++/ Python / OpenCV/MATLAB and various other languages thus it gives us the platform independency much essential for any algorithms.

Thus, this algorithm successfully removes most of the demerits of the existing systems and produces a highly cost effective and fast method for Encrypting images for safe and secure transmission and storage

4. ARCHITECTURE OF PIXEL PROPERTY PERMUTATION

The architecture of the proposed Pixel Property Permutation Algorithm(PPP) has been illustrated via the block diagrams Fig 4.1 and Fig. 4.2. Fig 4.1 describes the Encryption Process and the Fig 4.2 describes the decryption process.

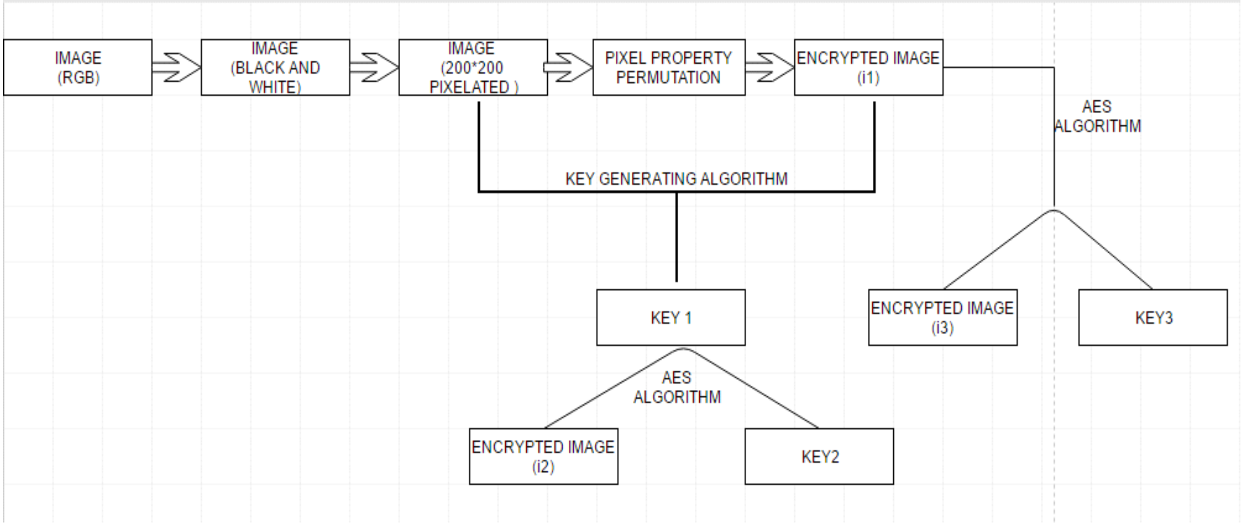


Fig 4.1 Block Diagram of the Encryption Process by converting an RGB Image to Black and White

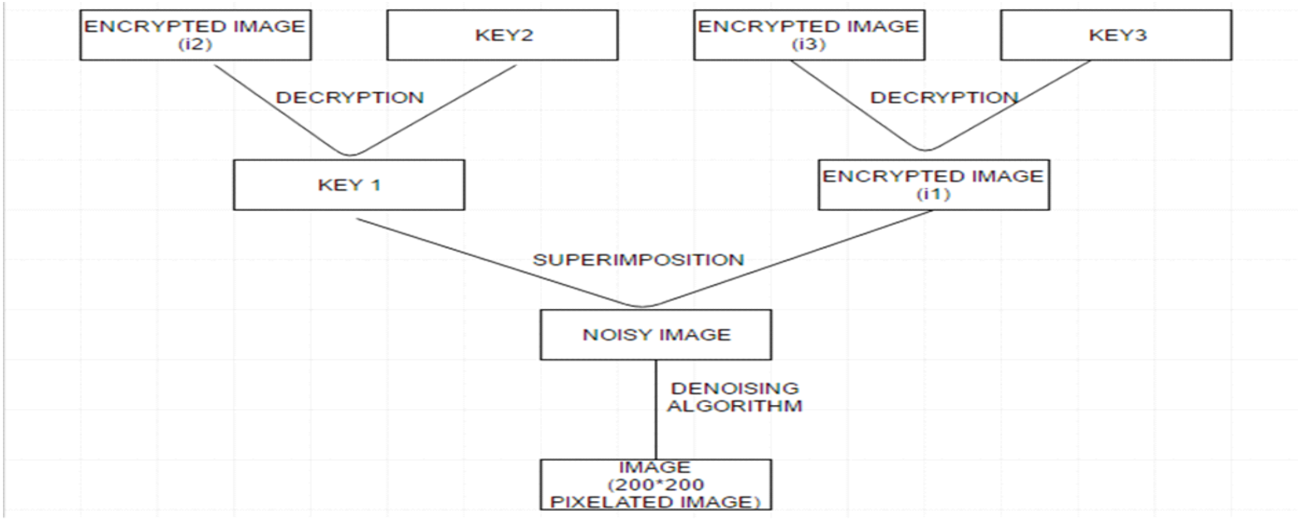


Fig 4.2 Block Diagram for the Decryption Process

The step by step process for Securing the visual data through PPP has been illustrated in the modules given below which describes the detailed analysis of the proposed algorithm.

4.1 Compression and Conversion Of Image

Unstructered Data comprises of various different typpes of images like RGB Colour image which stores three different set of Pixel values Red, Green and Blue and Grey Scale reflects the amount of light for each pixel, little light generates dark pixels and the large light generates the Bright pixels. The first step of this Encryption technique is that the Image either in RGB or in GreyScale needs to be converted to Black and White Binary Image. One approach towards conversion of an RGB color image into the Black and White image is to take each of the RGB values and convert it into one value that reflects the brightness of that pixel. This can be accomplished by taking the average of the three values (R + G +B)/3. But this will be highly erroneous. So, what we can do is that we can allow the brightness to be perceived majorly by the green color and thereby take a weighted average of it for e.g.: 0.6G +0.2B + 0.2R.

After converting the image, we have to compress the image without any loss to most of its features. Such a compression technology is termed as Lossless Compression. There are various algorithms to accomplish lossless compression differing on the quality of the compression to its varying complexity. There are various image formats that perform the lossless compression like PNG, GIF whereas there are some that uses some lossy methods like TIFF and MNG.

In our procedure the Image is converted and compressed because it would save a lot of complexity while performing random Permutations. However, if necessary then this algorithm could also work without converting and compressing the image thereby rendering this step null and void.

4.2 Digital Image Pixel Extraction

Digital images are a binary representation of a two-dimensional image that can be stored depending upon the whether the image resolution is fixed or not, in a vector or else a raster form. Raster images are the type of images that comprises of a set of digital values called pixels. They are the smallest element in an image containing values representing the brightness of a given colour at a specified time. After we have successfully converted an RGB Image to its corresponding Black and White or Grey Scale Image, we now proceed to extract all those values

Pixels in an image consists of two major properties position and value. The pixel position is defined by the *(x, y)* co-ordinates and the pixel values indicate the RGB or in our case the greyscale value. A digital image essentially comprises of m rows and n columns of pixels. Thus, these values can easily be extracted by converting the image into a textual image. If the image is an RGB image then each pixel value would have to store three different values red green and blue. All these values are stored in an n*m matrix which will then be used to perform permutation of over these values.

4.3 Random Permutation of the Pixel Property

This is one of the major step proposed in this project. Here we will try to permutate the value of the Pixels of an Image. This pixel matrix will be exposed to various permutating algorithms which can later be reversed to give us the decrypted image.This encryption scheme uses a Secret Key Sharing Algorithm which require 2 Keys. This Encryption process is completely based on changing the Pixel Position and its Values. To increase the efficiency of the Encryption Process the values are permuted using a special Random Permutations. It is these Random Permutations that can be useful to derive at the desired results. For easier Understanding we are going to assume that the image is in Black and White. The same can be accomplished in an RGB Image also but with higher Complexity. In the previous step of this process we had extracted the pixel values in an m*n Matrix. Now all these Pixel Values need to be encoded. For each cell we can have multiple values. If an Image is strictly Black and White then we can simply encode the values nearing Black to 0 and the values nearing White to 1. If not then we can keep the values in their 8-bit Binary Form. Once we have accomplished encoding the matrix we will now perform Permutations. The following enlists the proper Process to accomplish it:

- The Random Permutations used here is accomplished by using the Fisher-Yates Shuffle Algorithm also known as the Knuth Shuffle algorithm to shuffle the finite sequence of Numbers.
- Shuffle the Elements present in the i^{th} row with each other by selecting randomly a number and swapping it with the first index.
- The number that was selected is thus stroked out and is not taken into consideration.
- The same process is repeated for all the elements in a single row and in all the rows in the matrix.
- The above can also be accomplished using Columns instead of rows, but whichever convention is chosen needs to be followed.

When the plain text image is a greyscale image, the pixel values in the image is constructed depending upon the amount or intensity of light. It carries only intensity information. The pixel values in this matrix comprises of an 8-bit binary value which indicates different shades of grey. These values are stored in the W*H matrix, where W=width and H=height. All the bits of the 8-bit binary value of each and every pixel in the image are permuted with each other using the method described above.

The Process explained above takes about O(n^2) complexity which will exponentially increase if more Colors are to be Encoded and is taken into consideration.

4.4 Key-Generating Algorithm

The Image obtained above is used to generate the Key required to be used for the purpose of Encryption. The image obtained above is Superimposed with the original Image to give us another Image. This image can be then be superimposed with the PPP image to give us the Original Image.

Superimposing can be described as the process of combining multiple images and overlaying them on top of each other. Superimposition of two-dimensional Images can cause for the production of moiré patterns. Superposition of parallel and correlated layers gives rise to line moiré patterns. When Superimposition takes place over two identical Patterns there may arise randomly selected patterns at a small Scaling difference or available at a steep angle. All the above factors may cause for a distortion in the image and hence needs to be dealt with. The Glass moiré and shape moiré patterns may also arise due to the complex shapes that are sometimes created. Once we get this Image then we can use this image as our Key for the decryption of the Encrypted Image obtained after performing the PPP Process.

The Encrypted PPP image obtained from the above step and the Plaintext Image obtained is used to generate the Key used for the decryption process. The key is generated using each of the 8- bit binary value of the image complementing them with each of the 8-bit binary pixel value of the PPP image. This key obtained can thus be used for the purpose of decrypting the GreyScale image.

4.5 AES Encryption of the Key and The PPP Image

Another step of security is added by encrypting the key and the PPP image with AES Encryption technique, the most widely known encryption technique which has been found to be much faster than the earlier known standard DES. The AES encryption algorithm is used here as an added layer of defense which is not essentially necessary but encouraged for much greater security essential for highly sensitive data. The AES Encryption Algorithm is best defined as a symmetric block cipher encryption developed by using a Rijndael Cipher comprising of 128, 192 and 256-bit Keys. It is among the most secure software available in the industry today. We use this algorithm to encrypt the key and the Image obtained after performing encryption via the PPP method. This would enable us to secure the data already secured in a more prolific manner.

4.6 Decryption Process

The Keys and the image encrypted through the PPP technique needs to be decrypted back to obtain the original image. The decryption process is very similar to the encryption process.

The encryption of the visually encrypted image through AES Algorithm gives rise to a key. This unique key varies from time to time and this key is useful to decrypt the image back to the visually encrypted format. The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms need to be separately implemented, although they are very closely related. The key is also simultaneously decrypted back to give us the Key image. This key along with the visually encrypted image via PPP is superimposed on each other to give us the original image. However, while superimposing these images we tend to get some noise In our image in the form of moiré patterns. Thus, we use a sophisticated denoising algorithm to remove that unwanted noise and give us the resultant image.

If our plaintext image is greyscale then superimposition will not work. The image can be converted to a Black&White Image which can later be converted Greyscale but there is a better approach. During the random permutations instead of permuting the 8-bit Binary Value with each other we can permute the 8-bit Binary Values of an index with each other bit value. This random permutation can be recovered by superimposing the values of the Key image obtained in the step number 4.4.

5. CRYPTANALYSIS

Cryptanalysis is the process of analyzing the information systems to study the hidden aspects of the systems. They are used to breach the cryptographic security systems and breach the hidden encrypted material even without knowing the Key. [10] Mathematical analysis on the other hand analyses the cryptographic algorithm for the study of side-channel attacks that take into consideration not the algorithm as its own but also its implementation. Some of these techniques are discussed below: -

5.1 Brute Force Attack

In a Brute Force Attack, the algorithm is implemented as a black box and we try all the possible combinations of the key until the correct key is developed. The complexity of such an attack depends on the key-generating Algorithm and the size of the key.

Key Size: - The maximum size of the key obtained depends on the encoding performed as described in the module 4.2. Let us take the best-case scenario in which the image is converted into Black and White. This encoding is performed on a 200*200 pixelated image. Now that means there are 40,000 pixels in our image. Depending on the Random Permutation of these pixels we get that each pixel could have either a value of 0 or a value of 1 and this thing would continue for 40,000 indices. Such a huge complexity would make it impossible to brute force with the current computing ability. Now, during decryption of an RGB image the encoding of the colored values gives rise to vectors in place of indices. Now trying to identify all these values in each of the 40000 vectors would cause the encryption process to be increased to a much higher complexity.

5.2 Ciphertext Only Attack

This is the method which uses various different stages for the decryption process and determines the candidates for each stage. A perfect set of candidates can be used to successfully obtain the decrypted process. There can be various methods using which one can establish a ciphertext attack. Reverse Transposition [8] is the process of analyzing the transposed vector of the given matrix and using this transformation one can calculate the number of iterations required to calculate the value of the different possible candidates. The iterations will be the factorial of its indices. As Reverse Engineering of the Random Permutation cannot be accomplished. Thus, the average number of decryptions needed to get the correct solution would still be half of the total number of combinations.

5.3 Chosen Plaintext Attack

This is the type of attack in which the Cryptanalyst has the idea and the knowledge of the encryption algorithm which uses the Random Permutations(RP). Even after the knowledge of these random Permutations is not useful for the process of analysis as the analyst cannot significantly determine the correct value of a particular index value by reverse engineering the algorithm. Thereby resulting in a not so efficient way of decrypting the encrypted Image.

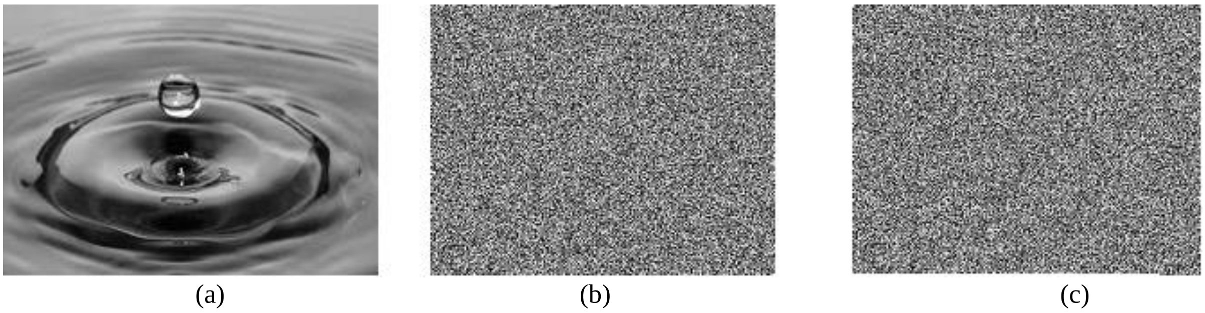
The basic method at disposal of the cryptanalyst is the encryption algorithm. Thus, he takes various plaintexts and subjugates them to the encryption algorithm to try and understand the algorithm in an effort to find and establish the key. He takes into account various different scenarios and different plaintexts until he is in a position to make a calculated guess on the Key obtained. However, as the Encryption process is established using the method of Random Permutation using the Fisher-Yates Shuffle Algorithm the process is much more time dependent and using the same plaintext different Keys are obtained thereby restricting the analyst to analyze and reverse engineer the algorithm. Thus, even after knowing the Encryption Algorithm it becomes effectively difficult for the person to establish a common link between the number plaintext and the key obtained and how to use the two to get the decrypted image.

5.4 Known Plain Text Analysis

In this analysis we assume that the Cryptanalyst has the knowledge of the plaintext and the Ciphertext. He knows the Width W and the Height H of the Image matrix. He even knows the Encoding criteria i.e. whether Black and White or RGB and also knows the way the image was encoded. Thus because of knowing these things performing Reverse Transposition took about half the factorial of the total number of iterations. However here the number of candidate keys obtained is significantly decreased. However due to the large values of the Width W and the Height H the value of the iterations is still quite large for easy extraction and decryption.

6. SIMULATION RESULTS

This section provides the basic simulation results obtained in trying to accomplish the task of Encryption. A basic image was taken in our simulation and converted to a Black and White image. This black and white 'water droplet' image is compressed to a size of 200*200 pixels. This image was now used to perform PPP on it. The PPP Encrypted image along with the key is shown below. This experiment was constructed on a 2.25GHz Intel Core i5 2nd Generation with 5GB Memory. Fig 5 depicts the entire simulation that existed on it. Fig 5(a) depicts the image and the Fig 5(b) depicts the PPP image obtained. Fig 5(c) depicts the Key obtained by following the Key-Generating Algorithm. The algorithm was carried out on a Java Platform with NetBeans as the IDE.



(a) (b) (c)

Fig 6.1'Water Droplet' Plain Image (a), the PPP Encrypted image(b), the Key Generated(c) during a simulation

The simulation time for all these experiments with varying size of the Image has been described in the table below, which has the dimensions of the plaintext image along with the encryption time(ET), decryption time(DT) taken for performing each of those random Permutations.

Table 6.1 Decryption and Encryption time as the size increases

Image	Compressed (*Black and White)* Image Size	PPP Image Size	Encryption Time (in sec)	Decryption Time(sec)
1	200 * 200	200 * 200	5.1	2.4
2	300 * 300	300 * 300	6.5	3.7
3	400 * 400	400 * 400	42.3	16

7. CONCLUSION AND FUTURE WORK

This Project streamlines on the basis of Privacy and data Security. It embodies itself as privacy and data security being a fundamental right of the citizens. Thus, in this project a model Image Encryption Technique has been illustrated that has been at the foundation of this project all along. The Image Encryption practiced under this Project has been accomplished using Pixel Property Permutation(PPP).

The basic Idea behind the PPP is that any digital image is made up of pixels small atomic units having a value and showcasing a colour. These pixel values are then permuted with each other so as to divide the entire report into different shares which in turn would give us nothing but a hazy image impossible to decipher. Thus, this is a major advantage and boost in the era of Visual Encryption. These techniques when used for the proper Data Encryption procedure enabled us to save the data in a more secure form than many of the technologies available to man. Given the high complexity, the proposed algorithm is secure against brute force, ciphertext only attacks, plaintext attacks and chosen plaintext attacks as has been proven above.

This technique has used AES Encryption Technique as an outer layer of defense and it uses this defensive measure to make it impenetrable via brute force or Cipher Attacks. Thus, these methods made the encryption more secure

This algorithm works perfectly well when the image is converted to a Black and White Image. However, once the image is taken and is to be stored as an RGB Image it assumes it to be a 24-bit image. This results in loosing some of the features of the image and might not give us the lossless image decryption. However in future once the Implementation becomes more robust theoretically we should be able to perform easily taking much less time the algorithm of Pixel Property Permutation.

REFERENCES

1. Roeder, Tom.:Symmetric-Key Cryptography.

2. Daniel J. Bernstein :Cache-timing attacks on AES in 2005

3. Andrey Bogdanoy, Dimitry Khovratovich and Christian Rechberger, :Biclique Cryptanalysis of the Full AES

4. S.El Assad, M. Farajallah, C. Vladeanu, :Chaos-Based Block Ciphers: An Overview, 2011

5. Zarko S. Stanisavljevic, :Data Encryption Standard Visual representation,1997

6. Sang-Su Lee, :Phase Masking Visual Cryptography using Interferometer. 2008

7. Schmeh, Klaus, :Cryptography and public key infrastructure on the Internet, p. 45 2001

8. Karthik Chandrashekar Iyer and Aravinda Subramanya,:Image Encryption by Pixel Property Separation

9. Min-Sung Koh, Esteban Rodriguez-Marek, Claudio Talarico,:A Novel Data Dependent Mulltimedia Encryption Algorithm secure against chosen Plaintext Attacks, ICME 2007