

الآية.

(بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ ﴿1﴾ الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ ﴿2﴾ الرَّحْمَنِ الرَّحِيمِ ﴿3﴾ مَا لِكَ يَوْمَ
الَّذِينَ ﴿4﴾ إِيَّاكَ نَعْبُدُ وَإِيَّاكَ نَسْتَعِينُ ﴿5﴾ اهْدِنَا الصِّرَاطَ الْمُسْتَقِيمَ ﴿6﴾ صِرَاطَ الَّذِينَ
أَنْعَمْتَ عَلَيْهِمْ غَيْرِ الْمَغْضُوبِ عَلَيْهِمْ وَلَا الضَّالِّينَ ﴿7﴾)

سورة الفاتحة

Acknowledgements

Best Thanks giving and completed by God Almighty, who created the best everything, which we succeeded in our Almighty what was this research to see the light without the Almighty and his generosity and kindness.

And with all the meanings of thanks and sincere gratitude thanks to Dr. Faisal Mohammed Abdalla, who oversaw the research, which we did not spare days in giving us information and tips and all what can benefit from it in the output of this research.

And I am also pleased to thank all those who contributed to the output of this work, especially beloved girlfriend Fatima Abdalla El-Hag.

We ask Allah Almighty to reward us richly rewarded and keep them stalwarts of science and knowledge.

Abstract

In this research, a new extension of the Playfair cipher algorithm to encrypt images safer manner. The new method is created matrix of 16×16 based on the key being entered by the user to become more secretive, and then is taken image data byte by byte. In addition, the increasing complexity of the algorithm using masking and an XOR. That is, the key is used to generate XORed with the image to encrypt it. The experimental results showed that the use of slightly different secret keys, and the resulting encoded images makes it a completely different picture, and also encrypts the data that contain alphanumeric characters, integers, and most symbols; we know that Playfair technique is best for multiple encryptions.

المستخلص

في هذا البحث، نسخة جديدة من خوارزمية الشفرات (Playfair) لتشفير الصور بطريقة أكثر أماناً . حيث يتم إنشاء مصفوفة 16×16 بناءً على المفتاح الذي يتم إدخاله من قبل المستخدم ليصبح أكثر سرية ،ومن ثم يتم أخذ بيانات الصور بايت تلو الآخر .وبالإضافة إلى ذلك، يتم زيادة تعقيد الخوارزمية باستخدام الاخفاء وإجراء (XOR) ، حيث يتم استخدام المفتاح لتوليد كمية (XORed) مع الصورة لتشفيرها .وأظهرت النتائج التجريبية أن استخدام مفاتيح سرية مختلفين قليلاً مع الصور المشفرة الناتجة يجعل الصورة مختلفة تماماً. وأيضاً تقوم بتشفير البيانات التي تحتوي على الحروف الهجائية، الاعداد الصحيحة وأغلب الرموز ; ونحن نعلم أن تقنية (Playfair) هي الأفضل للتشفير المتعدد.

Table of contents

الأيـة.....	I
Acknowledgements.....	II
Abstract.....	III
المستخلص.....	IV
Table of contents.....	V
Table of Figure.....	VII
List of Tables.....	VIII
Chapter 1.....	1
Introduction.....	1
1.1 Background:	1
1.2 Problem Statement:	1
1.3 Proposed solution:	1
1.4 Research Objectives:.....	1
1.5 Research Importance and Scope:.....	2
1.6 Research Methodology:	2
1.7 Related Work:.....	3
Chapter 2.....	4
Literature Review:.....	4
2.1 Introduction:.....	4
2.2 Two general types of key-based algorithms:.....	4
2.2.1 Symmetric algorithms:	4
2.2.2 Public-Key Algorithms:	5
2.3 Cryptanalysis:.....	6
2.4 Substitution technique:	8
2.4.1 Caesar Cipher	8
2.4.2 Monoalphabetic Ciphers:.....	9
2.4.3 Playfair Cipher:.....	10
2.4.4 Hill Cipher:.....	12
2.4.5 Polyalphabetic Ciphers	12
2.5 Transposition Ciphers:	14
2.6 Related Work:.....	16

2.7 Summary:.....	19
Chapter 3	20
Research Methodology	20
3.1 Introduction :.....	20
3.2 Programinng languages used to implemantion project:	20
3.2.1 Java:	20
3.2.1.1 Java Features:	21
3.2.2 Matlab:	21
3.2.2.1 Features of MATLAB:.....	21
3.2.2.2 Uses of MATLAB:.....	22
3.3 Encryption algorithm 1: The Playfair image encryption:	23
3.4 Decryption algorithm 2: The Playfair image decryption:	24
Chapter 4	27
Implementation	27
4.1 Introduction:.....	27
4.2 Key Space Analysis:.....	27
4.3 Visual Diffusion Test:	27
4.4 Implementation System of modified playfair cipher:	28
4.5 Validation and discution:.....	40
4.6 Summary:.....	41
Chapter 5	42
Conclusion and Recommendations	42
5.1 Conclusion:	42
5.2 Recommendations:	42
References:	43
Appendix	44

Table of Figure

Figure 1.1: Show steps of operations encryption and decryption.....	2
Figure 3.1: Show steps of operations encryption and decryption.....	25
Figure 4.1: Show how login user in the System	25
Figure 4.2: Show how Home page after loin user (khado) in the System.....	29
Figure 4.3:Show how upload image:	30
Figure 4.4 :Show how upload image:	30
Figure 4.5 :Show after upload image:.....	31
Figure 4.6 : Show how delete image:	32
Figure 4.7 :Show page after delete image	32
Figure 4.8 :Show how encrypted and decrypted texts	33
Figure 4.9 : Show After encrypted texts.	33
Figure 4.10 : Show After Decrypted texts.	34
Figure 4.11: Show how encrypted image:	35
Figure 4.12: Show encryption image:.....	36
Figure 4.13: Show how decrypted image.	36
Figure 4.14 : Showing original image after decrypted.	37

List of Tables

Table 1.1: Playfair 6x6 Matrix.....	3
Table 1.2 :Playfair 8x8 Matrix.....	3
Table 2.1: Playfair 5×5 matrix.....	10
Table 2.2: value letter from a to z.....	12
Table 2.3: The Modern Vigenère Tableau.....	13
Table 2.4: Transposition Ciphers.....	14
Table 2.5 : Playfair 7x4 Matrix.....	16
Table 2.6 :Playfair 6x6 Matrix.....	17
Table 2.7: Playfair 8x8 Matrix	19
Table 3.1: Playfair 16x16 Matrix without key	26
Table 3.2: Playfair 16x16 Matrix with the key	26
Table 4.1: the result of Modified Playfair Cipher for Encryption	38
Table 4.2: the result of Modified Playfair Cipher for Encryption	39
Table 4.3: (PSNR & MSE) Values for Standard and Various Cipher Images	40

Chapter 1

Introduction

1.1 Background:

As the general public became more aware of cryptographic uses, the personal and social need for privacy is increased.

The Playfair cipher is one of the ways to protect information is the method of encryption and decryption whereby the sender encrypts the message with a secret key which is known only to the receiver. Once the receiver gets the message the message is decrypted using the same secret key.

The Playfair cipher uses a 5×5 table containing a key word or phrase. Memorization of the keyword and 4 simple rules was all that was required to create the 5×5 table and use the cipher.

1.2 Problem Statement:

The existing playfair technique is based on the use of a 5×5 matrix of letters constructed using a keyword. This algorithm can only allow the text that contains alphabets only. But many algorithms have been proposed that allow text which contains alphabets, integers as well as special symbols using 6×6 matrixes and 10×9 matrixes etc.

All above versions can't encrypt Digital Images, because it does not include all the values of colors components (red , green, blue).

1.3 Proposed solution:

In this research will improve the Playfair cipher in order to apply it on image data.

It is known that a pixel in a true colored image has three color components: RED, GREEN, and BLUE. Their values range is between 0 and 255. So, using a matrix of size 16×16 filled with values between 0 and 255 can be a perfect solution to encrypt color values directly into other intensity levels.

1.4 Research Objectives:

The Playfair encryption process can be applied on pairs of the pixels color components in the original image.

Using modified playfair which makes the algorithm difficult to break.

1.5 Research Importance and Scope:

Images have become a basic element in life and the easiest and simplest way to connected information, and then it became very important to secure these data Images. encryption the Images by modified Playfair cipher 16X16.

1.6 Research Methodology:

We study playfair algorithm and collect image data (three color components: RED, GREEN, and BLUE). Their values range is between 0 and 255. So, using a matrix of size 16X16 filled with values between 0 and 255 And using modified playfair cipher for encrypt the data image to produce cipher image.

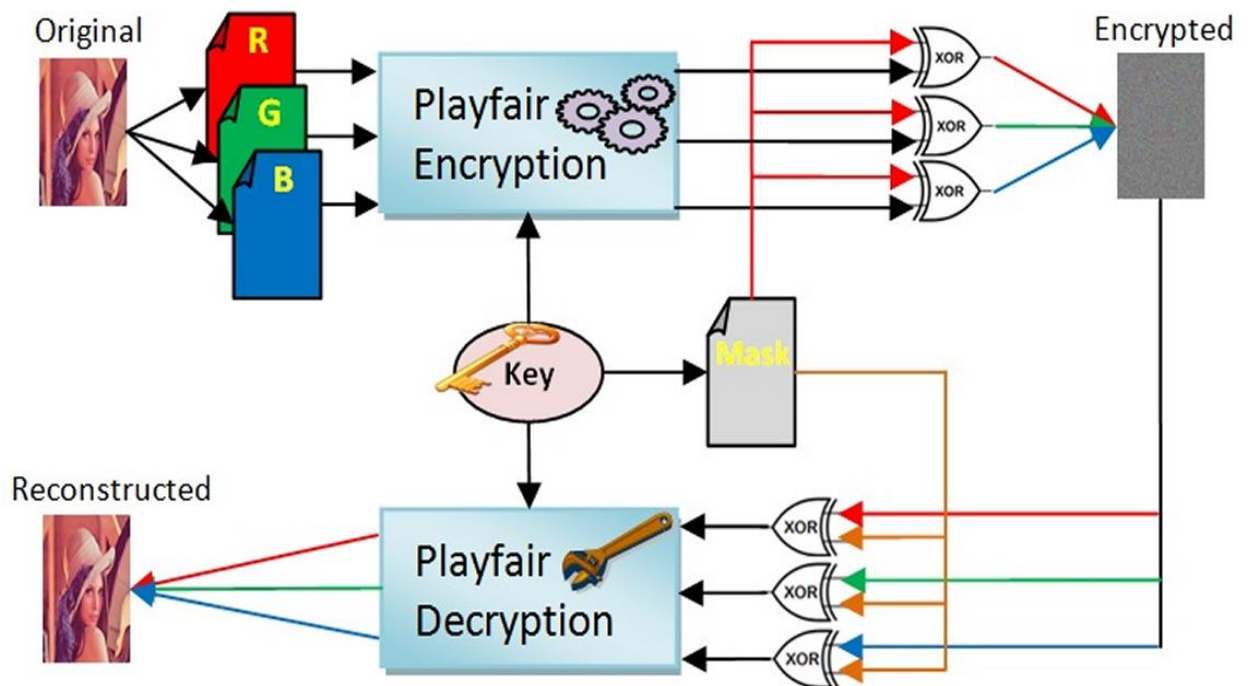


Figure 1.1: Show steps of operations encryption and decryption.

1.7 Related Work:

- Ravindra et al in [2].proposed an extension to the traditional Playfair algorithm. Their approach suggested using a 6 x 6 matrix instead of 5 x 5. The matrix is constructed in a similar way to the classic technique except that beside the set of alphabets this matrix is large enough to accommodate numerical digits (0 to 9) as well. Furthermore, the I/J was not counted as one l letter. Instead, Kamal et al.in [3].placed I and J in two separate cells in order to avoid ambiguity at decryption time.

Table 1.1: Playfair 6x6 Matrix

C	R	Y	P	T	O
A	B	D	E	F	G
H	I	J	K	L	M
N	Q	S	U	V	W
X	Z	0	1	2	3
4	5	6	7	8	9

- Currently, a new extension of classical Playfair cipher was presented by Hamad et al. in [4]. The proposed ciphering technique provides 8×8 amino acid codons substitution matrix. Furthermore, an interweaving step was added for more secured results.

Table 1.2 :Playfair 8x8 Matrix

G	M	A	I	L	.	C	B
D	E	F	H	K	P	Q	R
S	T	U	V	W	X	Y	Z
0	1	2	3	7	8	9	!
J	O	N	_	4	6	5	@
*	-	\$	#	,	/	+	?
;	%	=	&	'	\)	[
]	:	<	(>	“	{	}

Chapter 2

Literature Review:

2.1 Introduction:

Cryptography is where security engineering meets mathematics; it provides us with the tools that underlie most modern security protocols.

Computer security people often ask for non-mathematical definitions of cryptographic terms. The basic terminology is that cryptography refers to the science and art of designing ciphers; cryptanalysis to the science and art of breaking them; while cryptology, often shortened to just crypto, is the study of both. The input to an encryption process is commonly called the plaintext, and the output the ciphertext. Thereafter, things get somewhat more complicated. There are a number of cryptographic primitives—basic building blocks, such as block ciphers, stream ciphers, and hash functions.

Block ciphers may either have one key for both encryption and decryption, in which case they're called shared key (also secret key or symmetric), or have separate keys for encryption and decryption, in which case they're called public key or asymmetric. A digital signature scheme is a special type of asymmetric crypto primitive.

2.2 Two general types of key-based algorithms:

2.2.1 Symmetric algorithms:

Sometimes called conventional algorithms, are algorithms where the encryption key can be calculated from the decryption key and vice versa. In most symmetric algorithms, the encryption key and the decryption key are the same. These algorithms, also called secret-key algorithms, single-key algorithms, or one-key algorithms, require that the sender and receiver agree on a key before they can communicate securely. The security of a symmetric algorithm rests in the key; divulging the key means that anyone could encrypt and decrypt messages. As long as the communication needs to remain secret, the key must remain secret. [7]

Encryption and decryption with a symmetric algorithm are denoted by:

$$EK(M) = C$$

$$DK(C) = M$$

Symmetric algorithms can be divided into two categories. Some operate on the plaintext a single bit (or sometimes byte) at a time; these are called **stream algorithms** or **stream ciphers**. Others operate on the plaintext in groups of bits. The groups of bits are called **blocks**, and the algorithms are called **block algorithms** or **block ciphers**. For modern computer algorithms, a typical block size is 64 bits—large enough to preclude analysis and small enough to be workable. (Before computers, algorithms generally operated on plaintext one character at a time. You can think of this as a stream algorithm operating on a stream of characters.) [7].

2.2.2 Public-Key Algorithms:

called (asymmetric algorithms) are designed so that the key used for encryption is different from the key used for decryption. Furthermore, the decryption key cannot (at least in any reasonable amount of time) be calculated from the encryption key. The algorithms are called “public-key” because the encryption key can be made public: A complete stranger can use the encryption key to encrypt a message, but only a specific person with the corresponding decryption key can decrypt the message. In these systems, the encryption key is often called the **public key**, and the decryption key is often called the **private key**. The private key is sometimes also called the secret key, but to avoid confusion with symmetric algorithms, that tag won’t be used here.

Encryption using public key K is denoted by:

$$EK(M) = C$$

Even though the public key and private key are different, **decryption with the corresponding private key is denoted by:**

$$DK(C) = M$$

Sometimes, messages will be encrypted with the private key and decrypted with the public key; this is used in digital signatures. Despite the possible confusion,

these operations are denoted by, respectively:

$$EK(M) = C$$

$$DK(C) = M$$

2.3 Cryptanalysis:

The whole point of cryptography is to keep the plaintext (or the key, or both) secret from eavesdroppers (also called adversaries, attackers, interceptors, interlopers, intruders, opponents, or simply the enemy). Eavesdroppers are assumed to have complete access to the communications between the sender and receiver.

Cryptanalysis is the science of recovering the plaintext of a message without access to the key. Successful cryptanalysis may recover the plaintext or the key. It also may find weaknesses in a cryptosystem that eventually lead to the previous results. (The loss of a key through noncryptanalytic means is called a **compromise**.)

An attempted cryptanalysis is called an **attack**. A fundamental assumption in cryptanalysis, first enunciated by the Dutchman A. Kerckhoffs in the nineteenth century, is that the secrecy must reside entirely in the key.

Kerckhoffs assumes that the cryptanalyst has complete details of the cryptographic algorithm and implementation while real-world cryptanalysts don't always have such detailed information; it's a good assumption to make. If others can't break an algorithm, even with knowledge of how it works, then they certainly won't be able to break it without that knowledge.

There are four general types of cryptanalytic attacks. Each of them assumes that the cryptanalyst has complete knowledge of the encryption algorithm used:

1. Ciphertext-only attack. The cryptanalyst has the ciphertext of several messages, all of which have been encrypted using the same encryption algorithm. The cryptanalyst's job is to recover the plaintext of as many messages as possible, or better yet to deduce the key (or keys) used to encrypt the messages, in order to decrypt other messages encrypted with the same keys.

Given: $C_1 = E_k(P_1), C_2 = E_k(P_2), \dots, C_i = E_k(P_i)$

Deduce: Either $P_1, P_2, \dots, P_i; k$; or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

2. Known-plaintext attack. The cryptanalyst has access not only to the ciphertext of several messages, but also to the plaintext of those messages. His job is to deduce the key (or keys) used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same key (or keys).

Given: $P_1, C_1 = E_k(P_1), P_2, C_2 = E_k(P_2), \dots, P_i, C_i = E_k(P_i)$

Deduce: Either k , or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

3. Chosen-plaintext attack. The cryptanalyst not only has access to the ciphertext and associated plaintext for several messages, but he also chooses the plaintext that gets encrypted. This is more powerful than a known-plaintext attack, because the cryptanalyst can choose specific plaintext blocks to encrypt, ones that might yield more

information about the key. His job is to deduce the key (or keys) used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same key (or keys).

Given: $P_1, C_1 = E_k(P_1), P_2, C_2 = E_k(P_2), \dots, P_i, C_i = E_k(P_i)$, where the cryptanalyst gets to choose P_1, P_2, \dots, P_i

Deduce: Either k , or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

4. Chosen-ciphertext attack. The cryptanalyst can choose different ciphertexts to be decrypted and has access to the decrypted plaintext. For example, the cryptanalyst has access to a tamperproof box that does automatic decryption. His job is to deduce the key.

Given: $C_1, P_1 = D_k(C_1), C_2, P_2 = D_k(C_2), \dots, C_i, P_i = D_k(C_i)$

Deduce: k

This attack is primarily applicable to public-key algorithms and will be discussed in Section 19.3. A chosen-ciphertext attack is sometimes effective against a symmetric algorithm as well. (Sometimes a chosen-plaintext attack and a chosen-ciphertext attack are together known as a **chosen-text attack**.)

There are at least three other types of cryptanalytic attack:

5. Adaptive-chosen-plaintext attack. This is a special case of a chosen-plaintext attack. Not only can the cryptanalyst choose the plaintext that is encrypted, but he can also modify his choice based on the results of previous encryption. In a chosen-plaintext attack, a cryptanalyst might just be able to choose one large block of plaintext to be encrypted; in an adaptive-chosen-plaintext attack he can choose a smaller block of plaintext and then choose another based on the results of the first, and so forth.

6. Chosen-key attack. This attack doesn't mean that the cryptanalyst can choose the key; it means that he has some knowledge about the relationship between different keys. It's strange and obscure, not very practical.

7. Rubber-hose cryptanalysis. The cryptanalyst threatens, blackmails, or tortures someone until they give him the key. Bribery is sometimes referred to as a **purchase-key attack**. These are all very powerful attacks and often the best way to break an algorithm.

Known-plaintext attacks and chosen-plaintext attacks are more common than you might think. It is not unheard-of for a cryptanalyst to get a plaintext message that has been encrypted or to bribe someone to encrypt a chosen message.

2.4 Substitution technique:

A substitution technique is one in which the letters of plaintext are replaced by other letters or by numbers. If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns

In classical cryptography, there are four types of substitution ciphers:

— **A simple substitution cipher, or monoalphabetic cipher,** is one in which each character of the plaintext is replaced with a corresponding character of ciphertext. The cryptograms in newspapers are simple substitution ciphers.

— **A homophonic substitution cipher** is like a simple substitution cryptosystem, except a single character of plaintext can map to one of several characters of ciphertext. For example, “A” could correspond to either 5, 13, 25, or 56, “B” could correspond to either 7, 19, 31, or 42, and so on.

— **A polygram substitution cipher** is one in which blocks of characters are encrypted in groups. For example, “ABA” could correspond to “RTQ,” “ABB” could correspond to “SLL,” and so on.

— **A polyalphabetic substitution cipher** is made up of multiple simple substitution ciphers. For example, there might be five different simple substitution ciphers used; the particular one used changes with the position of each character of the plaintext.

2.4.1 Caesar Cipher [6]:

It was used by Julius Caesar. The Caesar cipher involves replacing each letter of the alphabet with the letter standing three places further down the alphabet

For example

Plain: meet me after the toga party

Cipher: PHHW PH DIWHU WKH WRJD SDUWB

Transformation is made using the following mapping:

Plain: a b c d e f g h i j k l m n o p q r s t u v w x y z

Cipher: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Let us assign a numerical equivalent to each letter from 0 to 25.

Then the algorithm may be expressed as follows. For each plaintext letter p , substitute the ciphertext letter C :

$$C = E(p) = (p + 3) \bmod 26$$

A shift may be of any amount, so that general Caesar algorithm is

$$C = E(p) = (p + k) \bmod 26,$$

where k takes on a value in the range 1 to 25. The decryption algorithm is simply

$$p=D(C)=(C-k) \bmod 26$$

If it is known that a given ciphertext is a Caesar cipher, then a brute-force cryptanalysis is easily performed: simply try all possible 25 keys.

Three important characteristics of this problem enable us to use brute-force cryptanalysis:

1. The encryption and decryption algorithms are known
2. There are only 25 keys to try
3. The language of the plaintext is known and easily recognizable

In most networking situations algorithms are assumed to be known. Brute-force analysis is impractical when algorithm employs large of keys. The 3rd characteristic is also significant. If the language of the plaintext is not known, then the plaintext output may not be recognizable.

2.4.2 Monoalphabetic Ciphers:

With only 25 keys Caesar cipher is far from secure. A dramatic increase in the key space may be achieved by allowing an arbitrary substitution. If instead of

Plain: a b c d e f g h i j k l m n o p q r s t u v w x y z

Cipher: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

The cipher line can be any permutation of the 26 alphabetic symbols, then there are $26!$ Or greater than $4 \cdot 10^{26}$ possible keys. There is however another line of attack. If the cryptanalyst knows the nature of the plaintext (e.g., noncompressed English text), then the analyst can exploit the regularities of the language.

2.4.3 Playfair Cipher:

Playfair cipher algorithm is based on with use of 5 X 5 matrix of letters constructed using a keyword. The 5 X 5 matrix can only allow 25 characters, hence the letters I/J count as one. If we encrypt the plaintext which is having the letter I/J and when we decrypt the ciphertext at the receive end, the receiver will be under ambiguity whether to consider I or J in his text, because the meaning can be changed with the change of the letters. This algorithm can only useful for the plain text containing of alphabets but it is failed for the plain text containing of alphanumeric values. **For example:**

**Table 2.1: Playfair 5×5 matrix
Key = simple**

s	i/j	m	p	l
e	a	b	q	d
f	g	h	k	n
o	q	r	t	u
v	w	x	y	z

The traditional Playfair cipher [3] uses 25 uppercase alphabets with I=J or Q omitted. A secret keyword is chosen and the 5 x 5 matrix is built up by placing the keyword without any duplication of letters from left to right and from top to bottom. The other letters of the alphabet are then placed in the matrix. The message is then broken a groups of 2 letters. In case of duplication of letters in a group one of the letters is used as padding and is placed between the letters. In case of odd number of characters the same padding is applied at the end. The substitution happens depending on the following three rules.

□ In case of letters of in the same row the letters to the right of each letter are taken. Wrapping happens in case one of the letters is at the last column.

□ In case of letters in the same column the letters to the bottom of each letter are taken. Again wrapping happens in case any letter is in the last row.

□ In case the letters are neither in the same row or column a rectangle is made with the letters and the letters at the opposite corners are taken.

In case of decryption the reverse process is done with the cipher text and we get back the plain text [6].

- **Limitations of 5x5 Matrix:**

The existing Playfair technique is based on the use of a 5 X 5 matrix of letters constructed using a keyword. This algorithm can only allow the text that contains alphabets only.

- **Attack of playfair :**

Playfair ciphers represent an improvement in security over substitution ciphers, but it is still relatively easy to attack them using a slightly more sophisticated form of frequency analysis. (The frequencies in English of the various digrams are well-known — can you guess what is the most common digram?) An entertaining, accurate, and surprisingly detailed discussion of a playfair attack appears in Dorothy L. Sayers' mystery novel *Have His Carcase*, (Victor Gollancz Ltd, 1932). Cryptanalysis of playfair ciphers is also explained in detail in Chapter 7, section II of the US Army Field Manual.

2.4.4 Hill Cipher:

This cipher is somewhat more difficult to understand than the others, but it is an important point about cryptanalysis that will be useful later on.

Another interesting multi-letter cipher is the Hill cipher, developed by the mathematician Lester Hill in 1929. The encryption algorithm takes m successive plaintext letters and substitutes for them m ciphertext letters.

The substitution is determined by m linear equations in which each character is assigned a numerical value ($a = 0, b = 1 \dots z = 25$).

Table 2.2: value letter from a to z.

a	B	c	D	e	F	g	h	I	j	k	L	M	n	o	P	q	R	s	t	U	v	w	x	y	z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

For $m = 3$, the system can be described as follows:

$$c_1 = (k_{11}P_1 + k_{12}P_2 + k_{13}P_3) \text{ mod } 26$$

$$c_2 = (k_{21}P_1 + k_{22}P_2 + k_{23}P_3) \text{ mod } 26$$

$$c_3 = (k_{31}P_1 + k_{32}P_2 + k_{33}P_3) \text{ mod } 26$$

This can be expressed in term of column vectors and matrices:

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} \text{ mod } 26$$

Or

$$C = KP \text{ mod } 26$$

where C and P are column vectors of length 3, representing the plaintext and ciphertext, and K is a 3 x 3 matrix, representing the encryption key. Operations are performed mod 26.

2.4.5 Polyalphabetic Ciphers [6] :

Another way to improve on the simple monoalphabetic technique is to use different monoalphabetic substitutions as one proceeds through the plaintext message. The general name for this approach is **polyalphabetic substitution cipher**. All these techniques have the following features in common:

1. A set of related monoalphabetic substitution rules is used.
2. A key determines which particular rule is chosen for a given transformation.

The best known, and one of the simplest, such algorithm is referred to as the Vigenère cipher. In this scheme, the set of related monoalphabetic substitution rules consists of the 26 Caesar ciphers,

with shifts of 0 through 25. Each cipher is denoted by a key letter, which is the ciphertext letter that substitutes for the plaintext letter a. Thus, a Caesar cipher with a shift of 3 is denoted by the key value *d*. To aid in understanding the scheme and to aid in its use, a matrix known as the Vigenère tableau is constructed (Table 2.1). Each of the 26 ciphers is laid out horizontally, with the key letter for each cipher to its left. A normal alphabet for the plaintext runs across the top. The process of encryption is simple: Given a key letter *x* and a plaintext letter *y*, the ciphertext letter is at the intersection of the row labeled *x* and the column labeled *y*; in this case the ciphertext is V.

Table 2.3: The Modern Vigenère Tableau

		Plaintext																									
		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Key	a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
	c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
	d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
	e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
	f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
	g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
	h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
	i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
	j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
	k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
	l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
	m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
	n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
	o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
	u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
	v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
	x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
	y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
	z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

To encrypt a message, a key is needed that is as long as the message. Usually, the key is a repeating keyword. For example, if the keyword is *deceptive*, the message "we are discovered save yourself" is encrypted as follows:

key: *deceptivedeceptivedeceptive*
plaintext: wearediscoveredsaveyourself
ciphertext: ZICVTWQNGRZGVTWAVZHCQYGLMGJ

Decryption is equally simple. The key letter again identifies the row. The position of the ciphertext letter in that row determines the column, and the plaintext letter is at the top of that column.

2.5 Transposition Ciphers:

All the techniques examined so far involve the substitution of a ciphertext symbol for a plaintext symbol. A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters. This technique is referred to as a transposition cipher. The simplest such cipher is the rail fence technique, in which the plaintext is written down as a sequence of diagonals and then read off as a sequence of rows. For example, to encipher the message "meet me after the toga party" with a rail fence of depth 2, we write the following:

m e m a t r h t g p r y e t e f e t e o a a t
 The encrypted message is
 MEMATRHTGPRYETEFETEOAAT

This sort of thing would be trivial to cryptanalyze. A more complex scheme is to write the message in a rectangle, row by row, and read the message off, column by column, but permute the order of the columns. The order of the columns then becomes the key to the algorithm. For example[6]:

Key: 3 4 2 1 5 6 7

Plaintext: attack postponed until two am xyz

Table 2.4: Transposition Ciphers

	1	2	3	4	5	6	7
A	T	T	A	c	K	P	
O	S	T	P	o	N	e	
D	U	N	T	i	L	t	
W	O	A	M	x	Y	z	

Ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ

A pure transposition cipher is easily recognized because it has the same letter frequencies as the original plaintext. For the type of columnar transposition just shown, cryptanalysis is fairly straightforward and involves laying out the ciphertext in a matrix and playing around with column positions. Digram and trigram frequency tables can be useful.

The transposition cipher can be made significantly more secure by performing more than one stage of transposition. The result is a more complex permutation that is not easily reconstructed. Thus, if the foregoing message is re-encrypted using the same algorithm,

Key: 3 4 2 1 5 6 7

Input: t t n a a p t

m t s u o a o

d w c o i x k

n l y p e t z

Output: NSCYAUOPTTWLTMDNAOIEPAXTTOKZ

To visualize the result of this double transposition, designate the letters in the original plaintext message by the numbers designating their position. Thus, with 28 letters in the message,

the original sequence of letters is

01 02 03 04 05 06 07 08 09 10 11 12 13 14

15 16 17 18 19 20 21 22 23 24 25 26 27 28

After the first transposition we have

03 10 17 24 04 11 18 25 02 09 16 23 01 08

15 22 05 12 19 26 06 13 20 27 07 14 21 28

which has a somewhat regular structure. But after the second transposition, we have

17 09 05 27 24 16 12 07 10 02 22 20 03 25

15 13 04 23 19 14 11 01 26 21 18 08 06 28

This is a much less structured permutation and is much more difficult to cryptanalyze.

2.6 Related Work:

- (Aftab Alam et al., 2011 [2]) A keyword is used to construct 7x4 matrix using letters and symbols „*“ and „#“ which is the base for this Playfair Algorithm. The 7x4 matrix is constructed by filling keyword with no repeating letters. Here the keyword “CRYPTO” is used. The remaining spaces are filled with the rest of alphabets. As shown in the table 2, the last cell is filled by the symbol “#” and the remaining cell that is before the last cell is filled by the symbol “*”.

Table 2.5 : Playfair 7x4 matrix

C	R	Y	P
T	O	A	B
D	E	F	G
H	I	J	K
L	M	N	Q
S	U	V	W
X	Z	*	#

The same rules of playfair 5x5 matrix are used here to encrypt the plaintext with the following modification.

When same letters fall in a pair it adds “*” so that the message BALLS become BAL*LS.

If a word consists of odd number of letters, it will add symbol “#” to complete the pair. So BIT becomes BI T#. The symbol # is simply ignored when the ciphertext is decrypted.

- Limitations of 7x4 Matrix
 - o 26 characters only can take as a keyword without any repetition.
 - o The space between two words in the plaintext is not considered as one character.

- It cannot use numbers and special characters except „*“ and „#“.
 - It is not case sensitive.
 - It ignores the symbols „*“ and „#“ at the time of decipherment.
- (Ravindra Babu K et al., 2014 [8]) This playfair algorithm is based on the use of a 6x6 matrix using letters and numbers. Here also the keyword “CRYPTO” is used. The matrix is constructed by filling the letters of the keyword from left to right and from top to bottom, remaining cells of the matrix are filled by uppercase alphabets and numbers ignoring the letters of the keyword as in Table 3 [4]. This algorithm cannot consider the letters I and J as one character. Place I and J in two different cells in order to avoid the ambiguity at the time of decipherment. The rules of playfair 5x5 matrix are used to encrypt the plaintext.

Table 2.6 :Playfair 6x6 Matrix

C	R	Y	P	T	O
A	B	D	E	F	G
H	I	J	K	L	M
N	Q	S	U	V	W
X	Z	0	1	2	3
4	5	6	7	8	9

The Existing Playfair Algorithm using 6x6 Matrix overcome the problem of 5x5 Matrix:

Letters I and J are counted as two letters.

The alphabets and numbers are used in the plaintext and the keyword.

- Limitations of 6x6 Matrix:
 - This 6x6 matrix can only take 36 characters as a keyword without duplicates.
 - Space between two words in plaintext is not considered as one character.
 - The matrix cannot accept special character.
 - It is not case sensitive.
 - When plaintext word consists of odd number of characters, a spare letter X is added with the word to complete the pair. In the decryption process this X is simply ignored. This creates confusion because X is a valid character and it can be a part of plaintext, so we cannot simply remove it in decryption process.
 - When repeating plaintext letters that fall in the same pair are separated by a filler letter, such as X. This letter X affects the plaintext at the time of decipherment [11].
- (Shiv Shakti Srivastava et al., 2011 [5])Playfair cipher using 8×8 matrix and hence it would be using 64 grids. The proposed system not only encrypts the alphabets but also the numerals and special characters. It also shows space between words where required. The system uses different blocks for different alphabet, numerals and symbols. In Proposed System, it is used at the time of encryption to provide space between two words, ^ is used for stuffing between two alphabets if they are repeated in a pair and ^ will also be used to put at the end to get the last alphabet in pair if the total length comes out to be odd. At the time of decryption it will be replaced by blank space of one

alphabet and the symbol ^ will be discarded. Rules for encoding and decoding will be same as that for existing playfair cipher.

Table 2.7: Playfair 8x8 Matrix [10]

G	M	A	I	L	.	C	B
D	E	F	H	K	P	Q	R
S	T	U	V	W	X	Y	Z
0	1	2	3	7	8	9	!
J	O	N	_	4	6	5	@
*	-	\$	#	,	/	+	?
;	%	=	&	'	\)	[
]	:	<	(>	“	{	}

- **Advantage of 8X8 playfair cipher :**

- We can Encrypt & Decrypt any type of plain text (Alphabetical, Numerical & Special Symbols)
- Identification of individual diagrams is difficult.
- Frequency analysis difficult.

2.7 Summary:

All above versions from playfair can't try to encrypt Digital Images; then we want to improve or modify traditional playfair cipher to encrypt Images.

Chapter 3

Research Methodology

3.1 Introduction :

In this Research, we introduce an improvement over the Playfair cipher in order to apply it on image data. It is known that a pixel in a true colored image has three color components: RED, GREEN, and BLUE. Their values range is between 0 and 255. So, using a matrix of size 16×16 filled with values between 0 and 255 can be a perfect solution to encrypt color values directly into other intensity levels. Here, the key is expected to be an integer number that is supplied as the seed value in a random permutation module to randomly construct the substitution matrix. Then, the Playfair encryption process can be applied on pairs of the pixels color components in the plain image. The resultant scrambled image is not the final output yet. However, the proposed system adopts an XOR operation as an additional step to improve security. Here, the secret key is used once more to generate a random mask that has the same dimensions as any of the image's color component matrices. This random mask is XORed with the scrambled image in order to produce the cipher-image. This additional step process guarantees that the resultant cipher image is completely different from the plain image even if two similar keys were used. Figure 2 gives an overview of the proposed ciphering system or a more detailed look, Algorithms 1 and 2 list the steps of the encryption and the decryption processes respectively.

3.2 Programing languages used to implemantion project:

3.2.1 Java:

Every time you use a computer, you execute various applications that perform tasks for you. For example, your e-mail application helps you send and receive e-mail, and your Web browser lets you view Web pages from Web sites around the world. Computer programmers create such applications by writing computer programs.

A Java **application** is a computer program that executes when you use the **java command** to launch the Java Virtual Machine (JVM). Let us consider a simple application that displays a line of text. The program illustrates several important Java language features. Java uses notations that may look strange to nonprogrammers. In addition, for your convenience, each program we present in this book includes line numbers, which are not part of actual Java programs.

3.2.1.1 Java Features:

- Simple.
- Object oriented
- Interpreted
- Portable
- Reliable
- Secure
- Multithreaded
- Dynamic

3.2.2 Matlab:

MATLAB is a environment for scientific computing that is ideal for computations that require extensive use of arrays and graphical analysis of data.

- It is a interpreted language (no compiler); scripts can be saved as .m files
- Array indices begin with 1 (compare to 0 in C or Java)
- Arrays are passed by value to functions (no pointers)
- Array elements are accessed with the format A(1,2) (compare to the format A[0][1] in C or Java)
- Powerful matrix mathematical functions are built-in (e.g., \ for Gaussian elimination or least-squares solution methods for linear systems).

3.2.2.1 Features of MATLAB:

Following are the basic features of MATLAB:

- It is a high-level language for numerical computation, visualization and application development.

- It also provides an interactive environment for iterative exploration, design and problem solving.
- It provides vast library of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and solving ordinary differential equations.
- It provides built-in graphics for visualizing data and tools for creating custom plots.
- MATLAB's programming interface gives development tools for improving code quality maintainability and maximizing performance.
- It provides tools for building applications with custom graphical interfaces.
- It provides functions for integrating MATLAB based algorithms with external applications and languages such as C, Java, .NET and Microsoft Excel.

3.2.2.2 Uses of MATLAB:

MATLAB is widely used as a computational tool in science and engineering encompassing the fields of physics, chemistry, math and all engineering streams. It is used in a range of applications including:

- Signal Processing and Communications.
- Image and Video Processing.
- Control Systems.
- Test and Measurement.
- Computational Finance.
- Computational Biology.

3.3 Encryption algorithm 1: The Playfair image encryption:

input : Plain image and Secret key

output: Cipher image

1. Read the plain image as RED, GREEN and BLUE matrices.
2. If the plain image has an odd-number dimension append a row or column of zeros to the end to make it even.
3. Construct a Key Square: 16 x16 matrix of random integer numbers between 0 and 255 using the secret key.
4. For each pair of color components in the RED plane of the plain-image do the following:
 - (a) If the values are in different rows and columns, replace the pair with the values at the opposite corners of the rectangle defined by the original pair and maintain their order.
 - (b) If the values appear on the same row of the matrix, replace them with the values to their immediate right respectively (wrapping around to the left side).
 - (c) If the values appear on the same column of the matrix, replace them with the values immediately below respectively (wrapping around to the top side of the column).
5. Use the secret Key to generate a mask made up with a random permutation of the numbers between 0 and 255.
6. XOR the resultant scrambled image with the generated random mask.
7. Repeat step 4 to 6 for GREEN and BLUE color planes of the plain image.
8. Return the resultant image as the cipher-image.

3.4 Decryption algorithm 2: The Playfair image decryption:

input : Cipher image and Secret key

output: Plain image

1. Read the Cipher image as RED, GREEN and BLUE matrices.
2. Use the secret Key to generate a mask made up with a random permutation of the numbers between 0 and 255.
3. XOR the RED color plane of the Cipher image with the generated random mask.
4. Construct a Key Square: 16×16 matrix of random integer numbers between 0 and 255 using the secret key.
5. For each pair of the resultant XORed RED plane of the Cipher image do the following:
 - (a) If the values are in different rows and columns, replace the pair with the values at the opposite corners of the rectangle defined by the original pair and maintain their order.
 - (b) If the values appear on the same row of the matrix, replace them with the values to their immediate right respectively (wrapping around to the left side).
 - (c) If the values appear on the same column of the matrix, replace them with the values immediately below respectively (wrapping around to the top side of the column).
6. Repeat step 3 to 5 for GREEN and BLUE color planes of the cipher image.
7. Return the resultant image as the Plain-image.

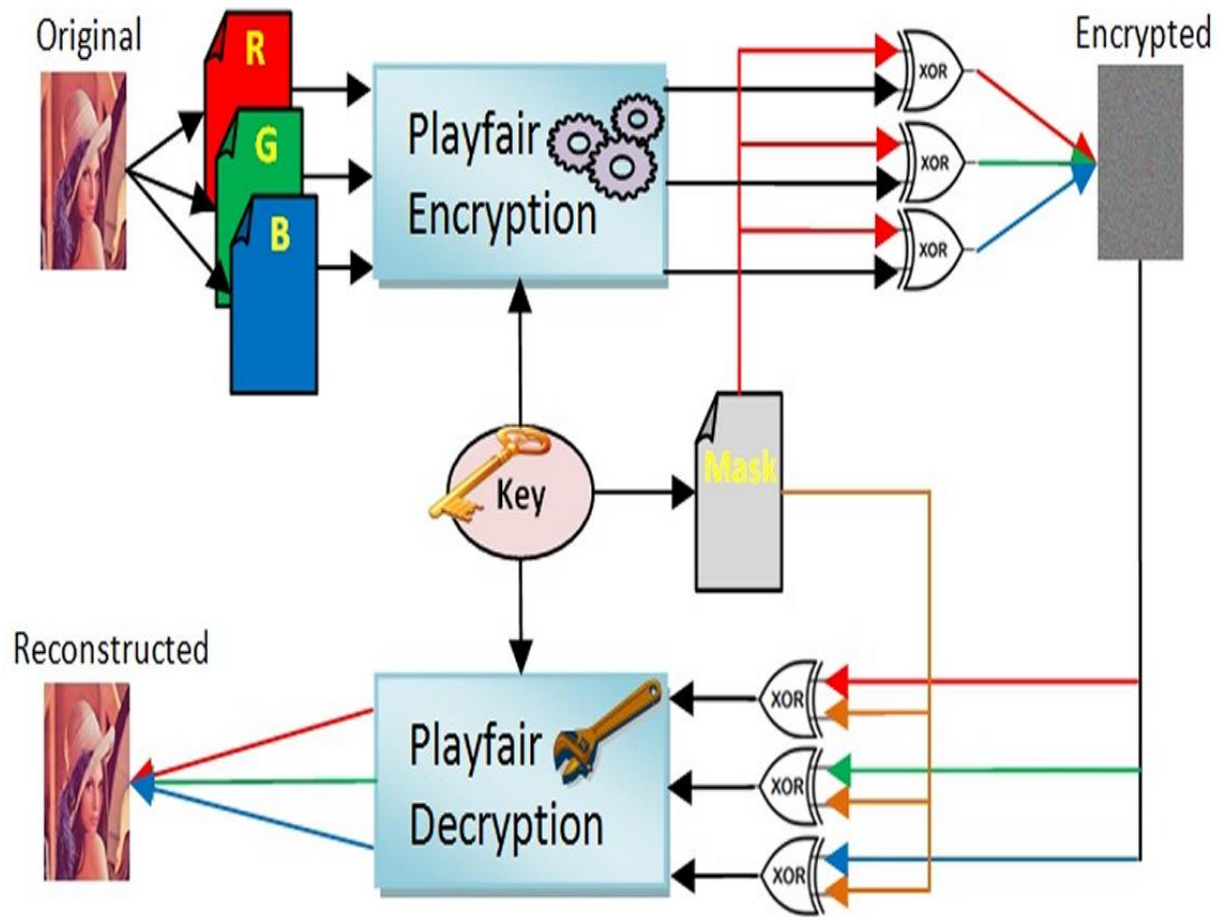


Figure 3.1: Show steps of operations encryption and decryption.

Table 3.1: Playfair 16x16 Matrix without key, fill all cells matrix by numbers from (0 to 255) Ascending

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	110	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	145	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

- If Key = Playfair587
- Translate to integer: without repeating
- Secret key = 80 108 97 121 102 105 114 5 8 7. [9]

Table 3.2: Playfair 16x16 Matrix with the key.

80	108	97	121	102	105	114	5	8	7	0	1	2	3	4	6
9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	81	82	83	84	85	86	87	88	89
90	91	92	93	94	95	96	98	99	110	101	103	104	106	107	109
110	111	112	113	115	116	117	118	119	120	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	145	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Chapter 4

Implementation

4.1 Introduction:

The new method was implemented using java and MATLAB for analysis. Three standards RGB color images were used for benchmark comparisons in different sizes.

Table [4:1] shows the result of new encryption method using one secret key on the images. Obviously the results showed the Decipher Image and original image is same.

4.2 Key Space Analysis:

Cryptanalysis is a field that attempt to find techniques to decrypt a message without prior knowledge on it ciphering method. Cryptanalysis is what the layperson calls “breaking the code”. Together with cryptography they are called *cryptology*.

In traditional Playfair cipher, obtaining the key is relatively straightforward if both plain-text and cipher-text are known. However, usually only the cipher text will be available. Thus, guessing some of the words based on knowledge about the origin of the message can be of a great help in reconstructing the substitution matrix. It should be recognized that guessing some of the plain-text and using that to reconstruct the key square is by far the easiest way to crack this cipher.

Cryptanalysis of the Playfair cipher for image is much more difficult than normal simple Playfair substitution cipher, because in this case digraphs represent pairs of pixels instead of pairs of letters. Applying the same analogy of frequency analysis requires inspecting 65536 pixel digraphs compared with only 676 in case of letter digraphs.

4.3 Visual Diffusion Test:

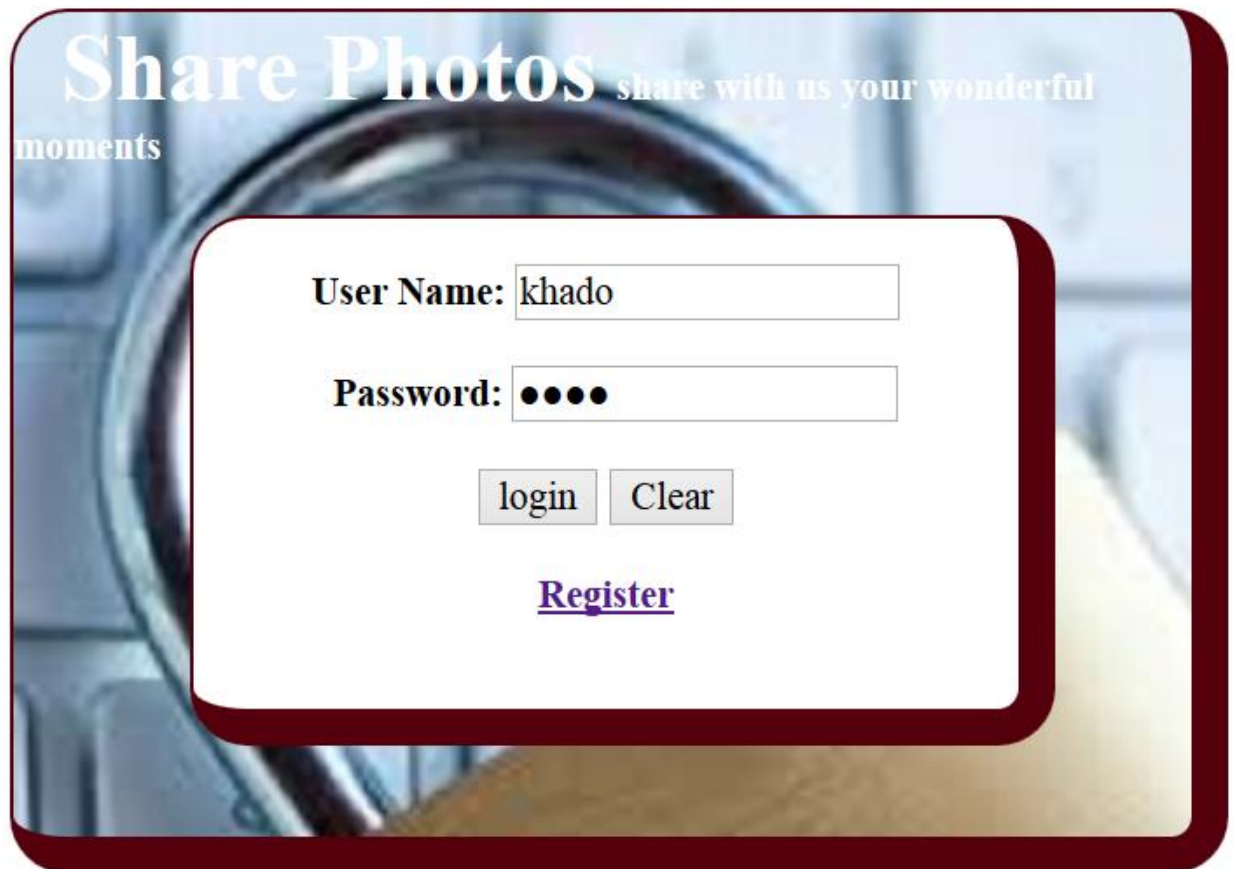
More experimentation has been conducted to visually judge the diffusion in the resulted images using similar key values. The popular PSNR metric was employed as a similarity measure. **PSNR can be computed using the following formula:**

$$PSNR = 10 \times \log\left(\frac{(\max f(x, y))^2}{MSE}\right) \quad (1)$$

$$MSE = \frac{1}{X \times Y} \sum_{x,y} (f(x, y) - p(x, y))^2 \quad (2)$$

Where $f(x, y)$ and $p(x, y)$ are the compared images of size $X \times Y$ and MSE denotes the Mean Square Error. PSNR values are often expressed in decibels (dB) where the values will run to infinity if the two examined images are identical.

4.4 Implementation System of modified playfair cipher:



The image shows a web application interface for a photo-sharing system. At the top, the text "Share Photos" is displayed in a large, white, serif font, followed by the tagline "share with us your wonderful moments" in a smaller, white, sans-serif font. Below this, there is a white login form with a dark red border. The form contains two input fields: "User Name:" with the text "khado" entered, and "Password:" with four black dots representing a masked password. Below the input fields are two buttons: "login" and "Clear". At the bottom of the form, there is a link labeled "Register" in a purple, underlined font. The background of the page is a blurred image of a camera lens.

Figure 4.1: Show how login user in the System.

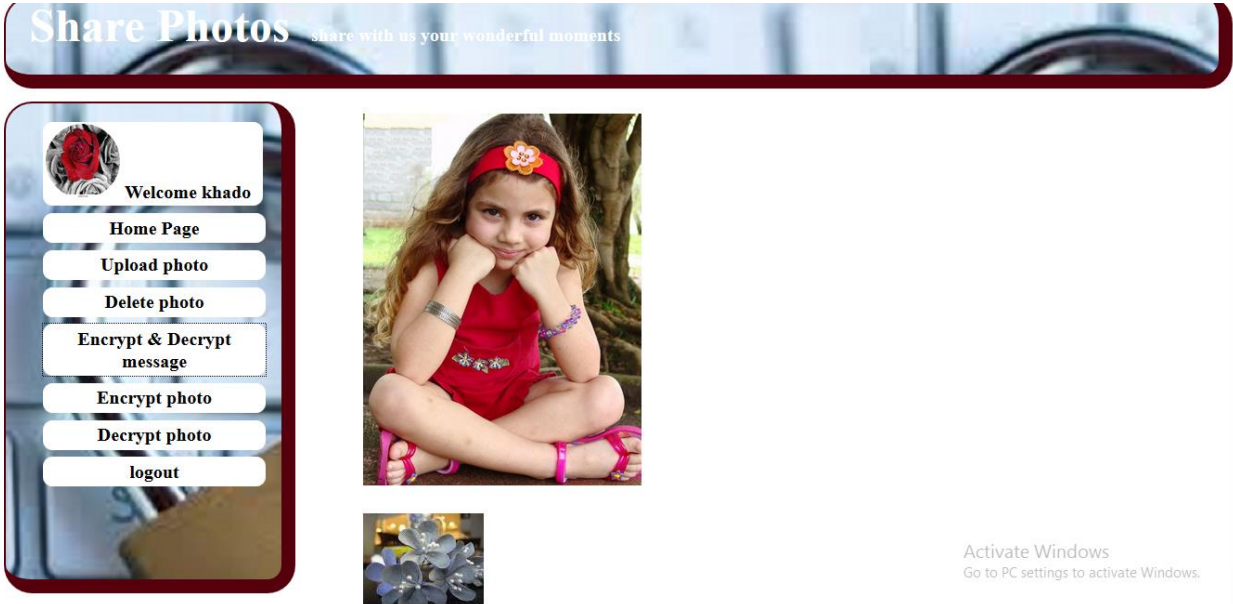


Figure 4.2: Show how Home page after login user (khado) in the System.

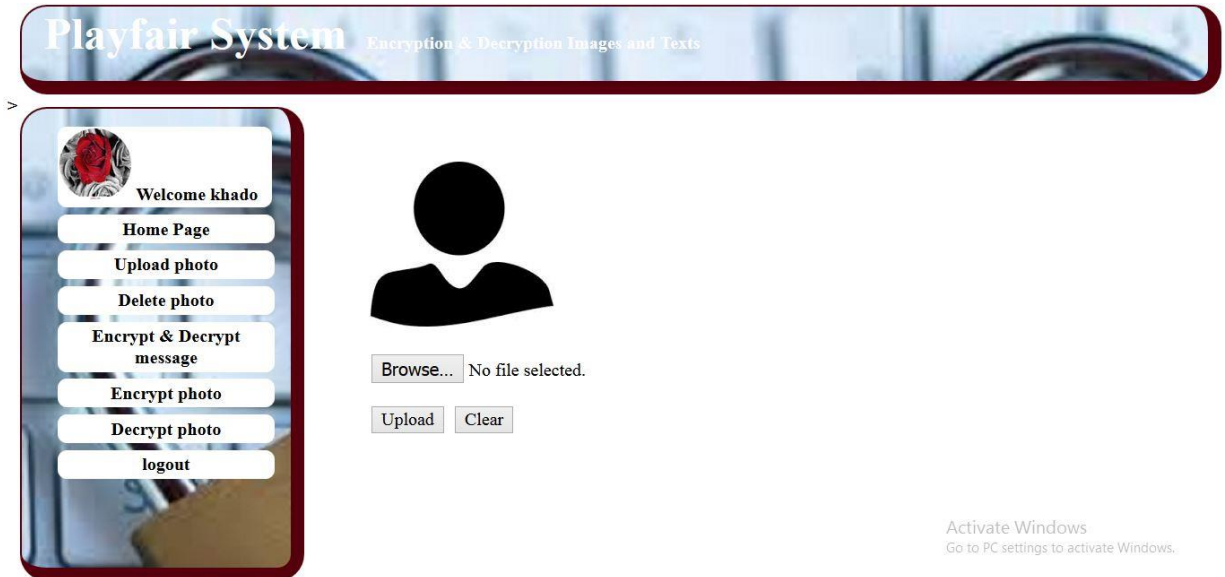


Figure 4.3: Show how upload image:

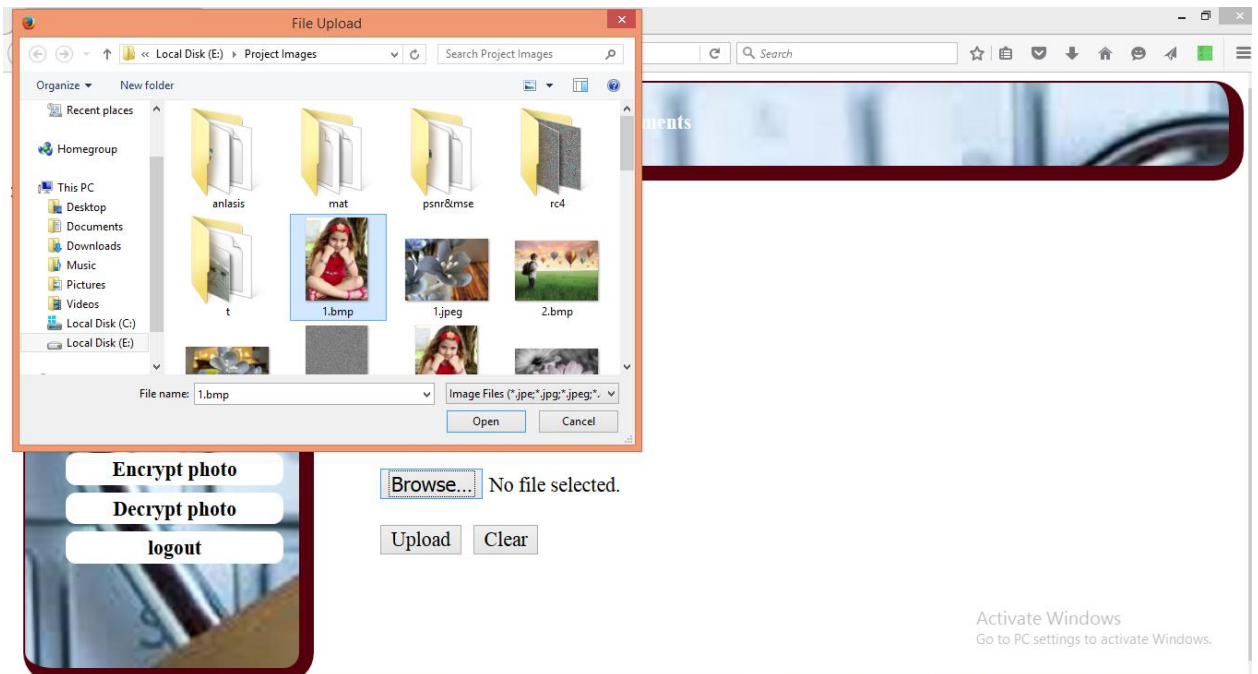


Figure 4.4 : Show how upload image:

 Welcome khado

- Home Page
- Upload photo
- Delete photo
- Encrypt & Decrypt message
- Encrypt photo
- Decrypt photo
- logout



Browse... 1.bmp

Upload Clear

Activate Windows
Go to PC settings to activate Windows.

Figure 4.5 :Show after upload image:

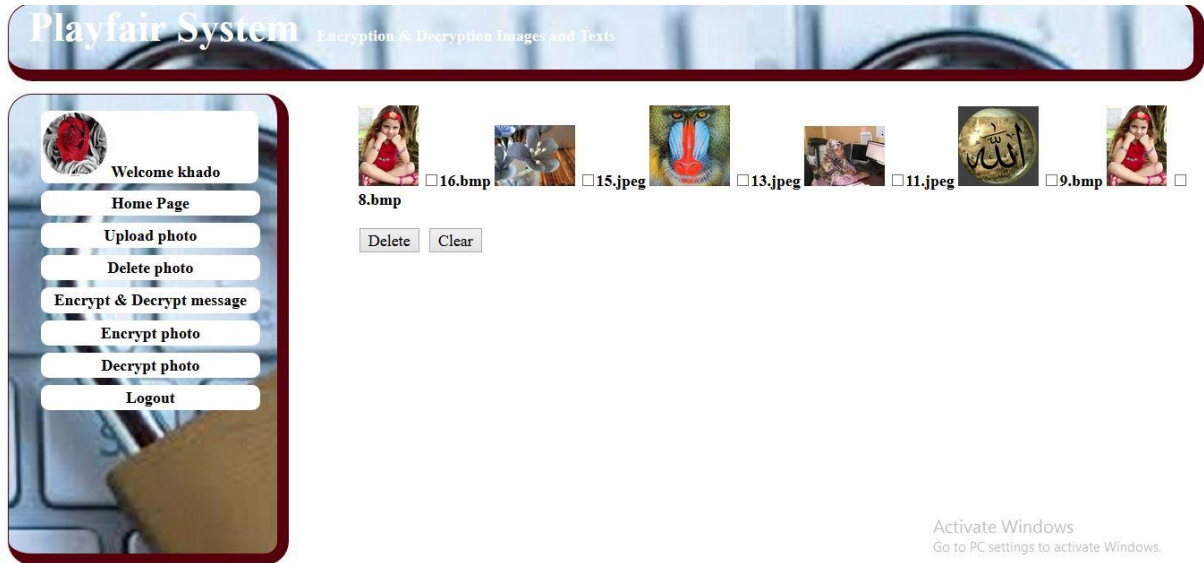


Figure 4.6 : Show how delete image:

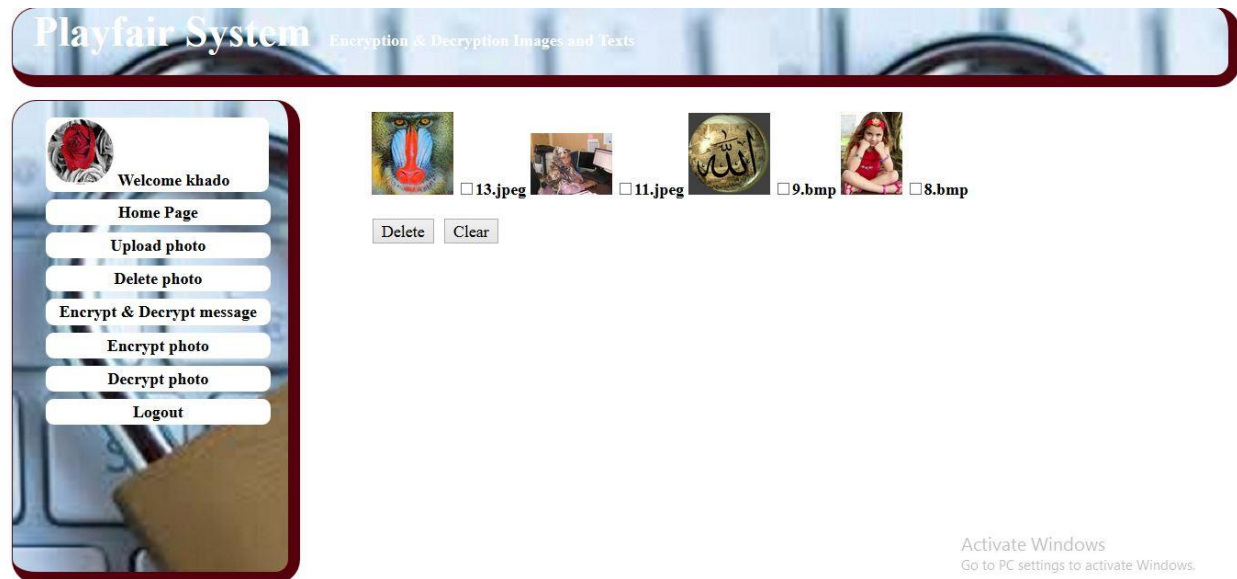


Figure 4.7 :Show page after delete image

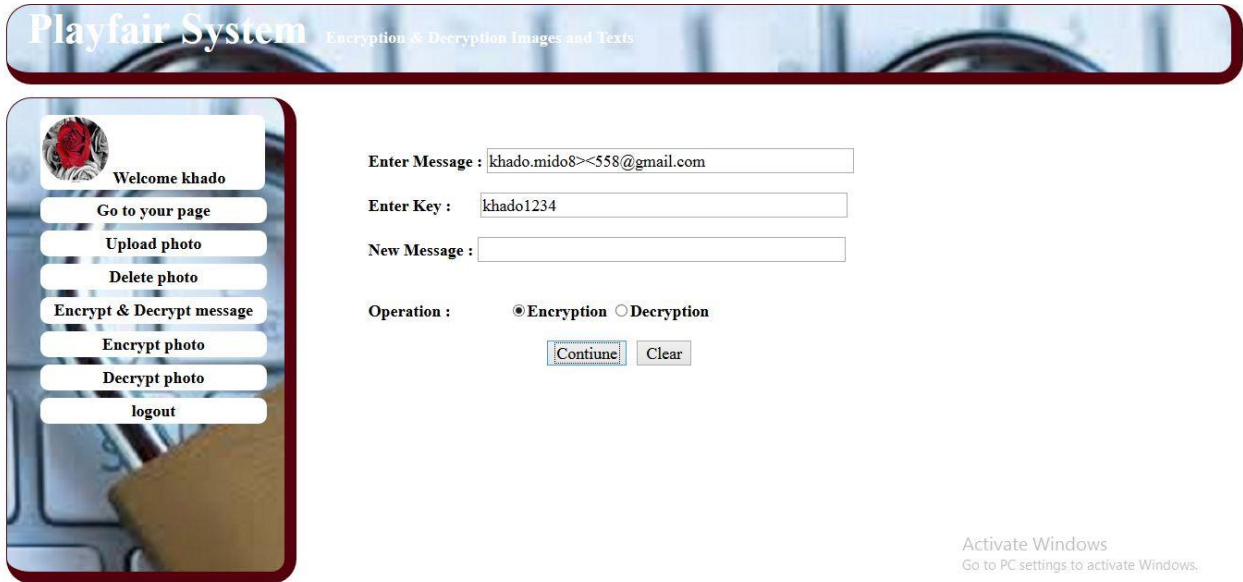


Figure 4.8 :Show how encrypted and decrypted texts

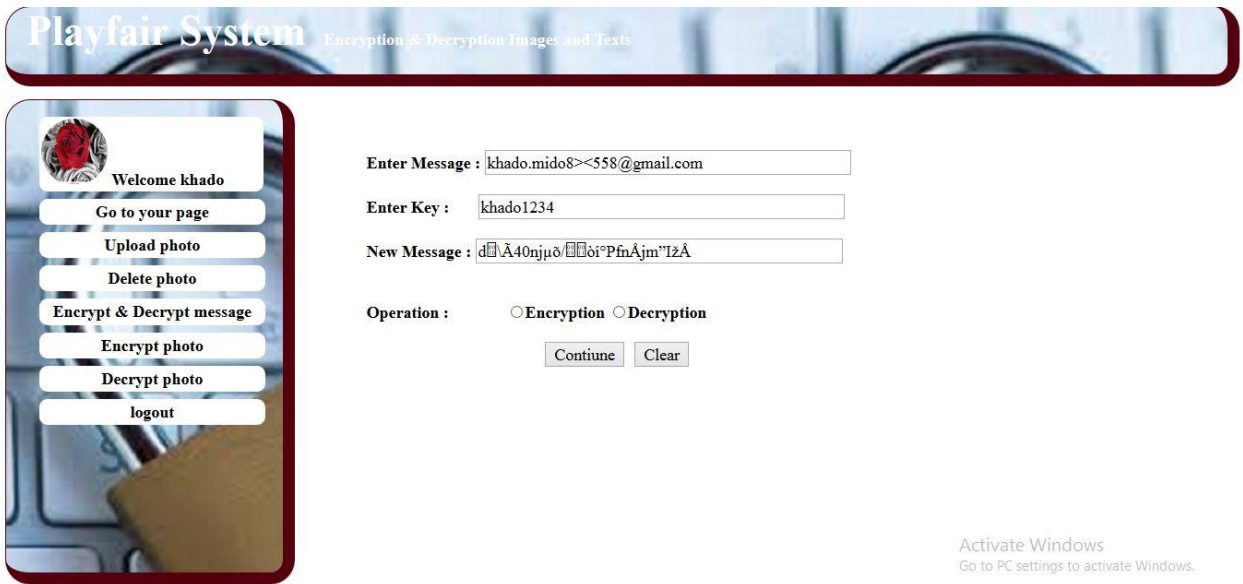



Figure 4.9 : Show After encrypted texts.



A vertical sidebar menu with a profile picture of a red rose. Below the profile picture is the text "Welcome khado". The menu items are: "Go to your page", "Upload photo", "Delete photo", "Encrypt & Decrypt message", "Encrypt photo", "Decrypt photo", and "logout".

Enter Message :

Enter Key :

New Message :

Operation : Encryption Decryption

Activate Windows
Go to PC settings to activate Windows.

Figure 4.10 : Show After Decrypted texts.

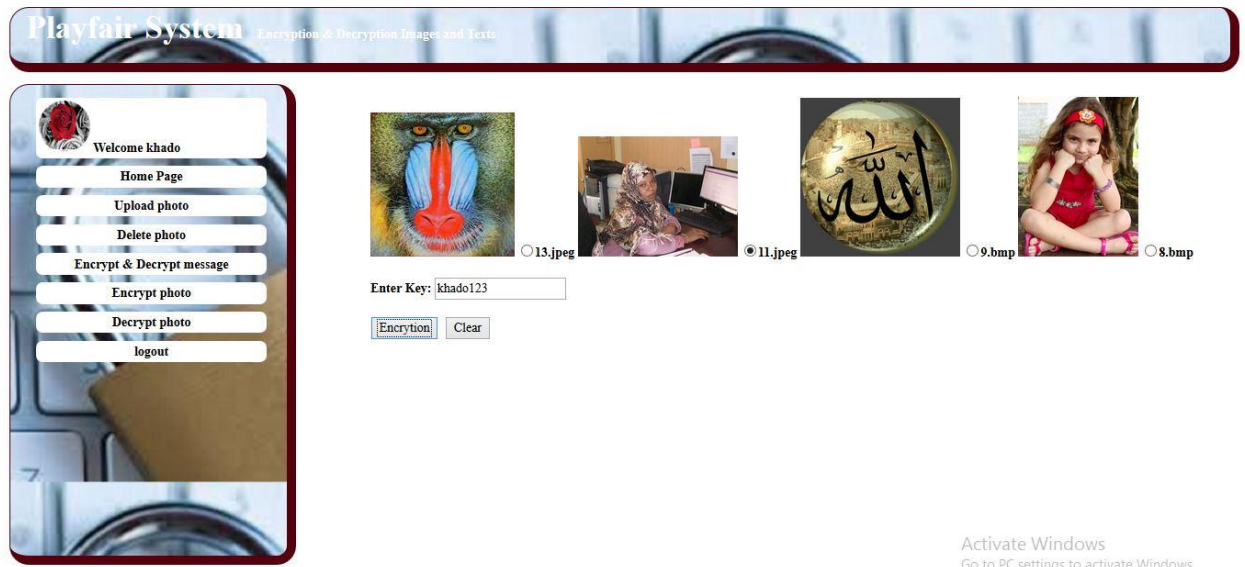


Figure 4.11: Show how encrypted image:

Steps of encryption:

- Choose (Encrypt photo) from left menu.
- Chose the image you want to encrypt.
- Enter your secret key.
- Press (Encryption) choice.

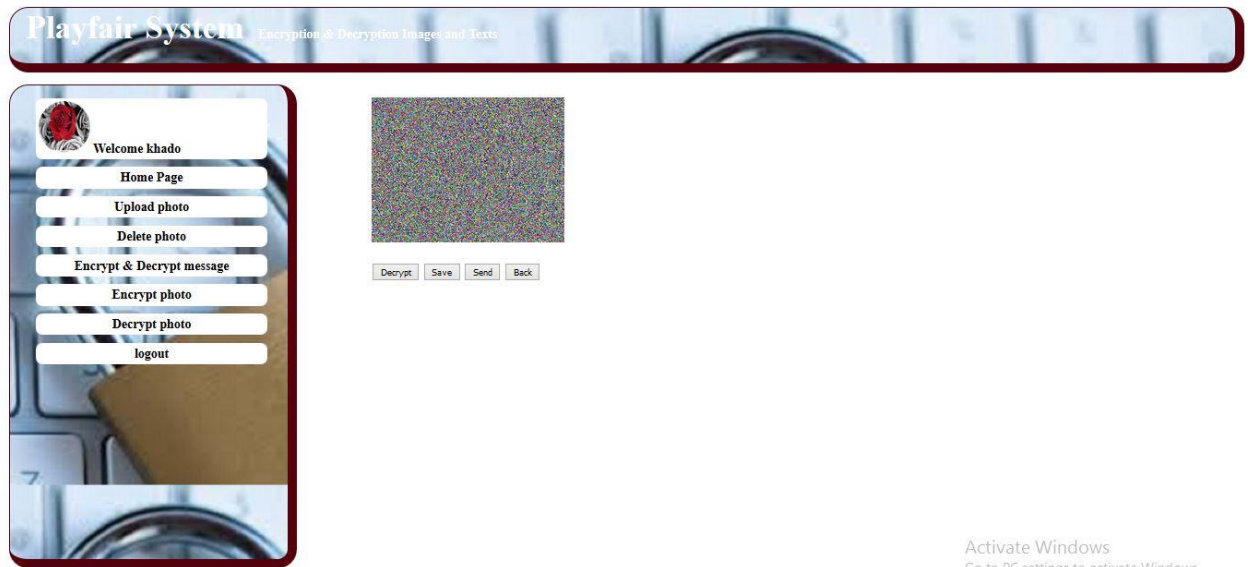


Figure 4.12: Show encryption image:

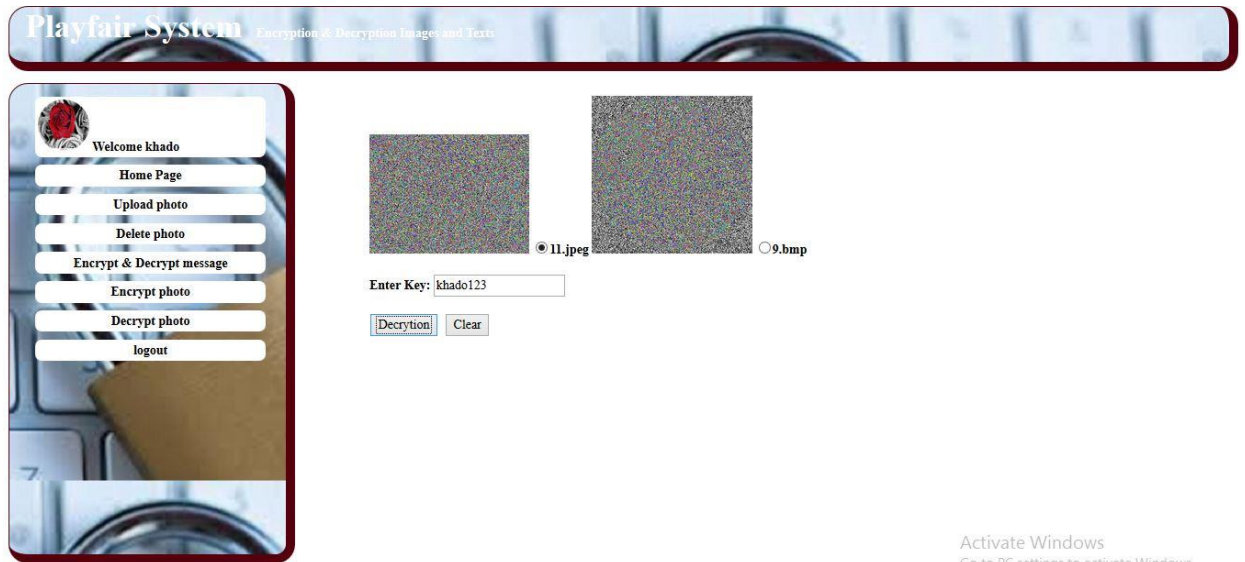


Figure 4.13: Show how decrypted image.



A vertical sidebar menu with a profile picture of a woman and the text "Welcome khado". Below the profile are several white buttons with black text: "Home Page", "Upload photo", "Delete photo", "Encrypt & Decrypt message", "Encrypt photo", "Decrypt photo", and "logout".



Activate Windows
Go to PC settings to activate Windows.

Figure 4.14 : Showing original image after decrypted.

Table 4.1: the result of Modified Playfair Cipher for Encryption, Decryption images and (PSNR& MSE) Values for Standard Images and Cipher Images using same key.




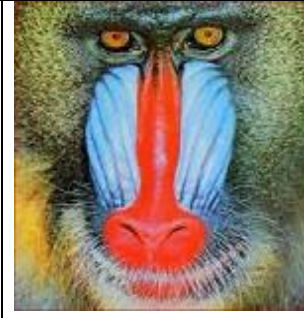
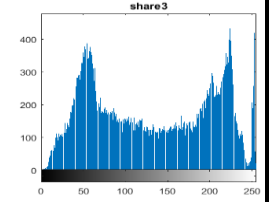
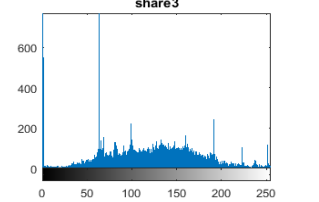
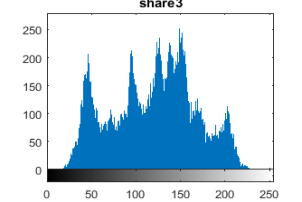
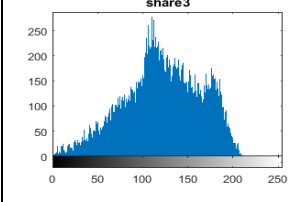
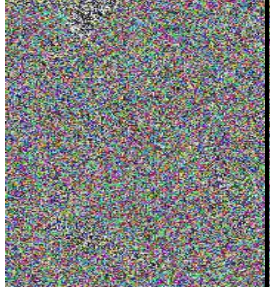
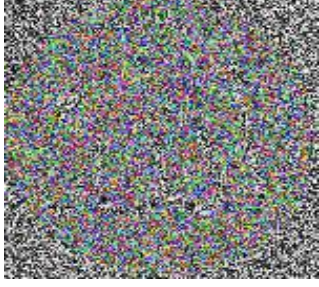


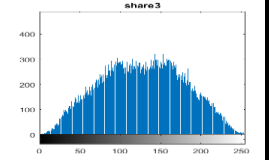
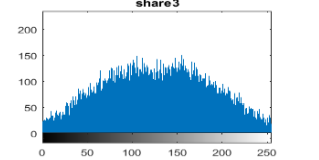
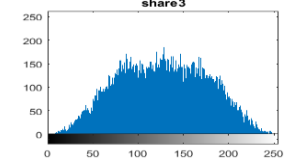
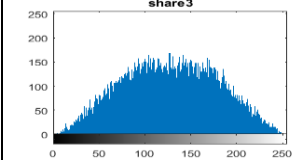



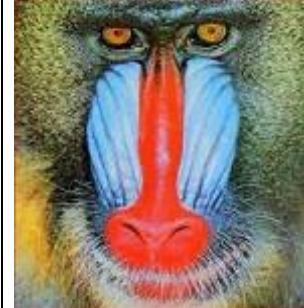
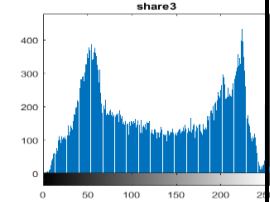
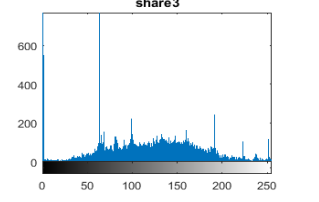
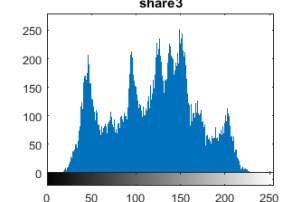
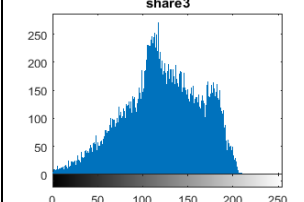
Original Image				
Histogram of original image				
Encrypted Image				
Histogram of Cipher image				
Decrypted Image				
Histogram after decrypted image				
Analysis	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB

Table 4.2: the result of Modified Playfair Cipher for Encryption, Decryption different size images and (PSNR& MSE) Values for Standard Images and Cipher Images using same key.





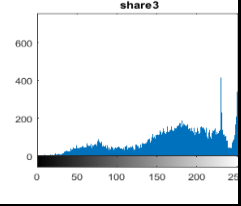
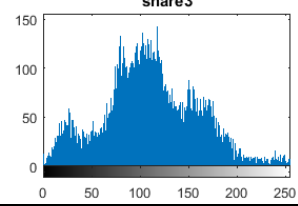
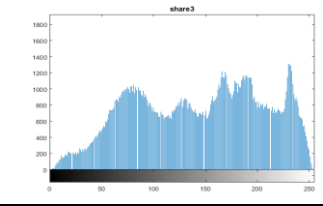
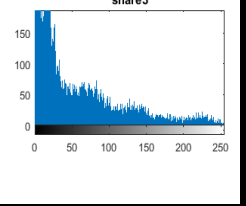
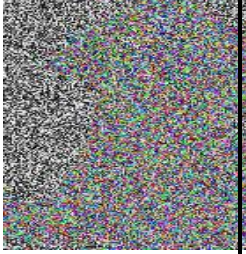



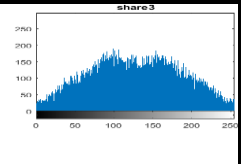
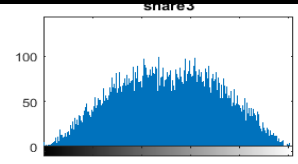
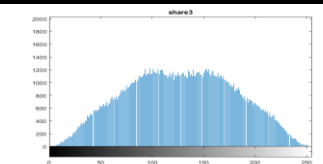
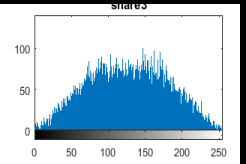




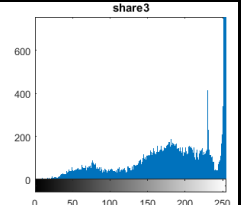
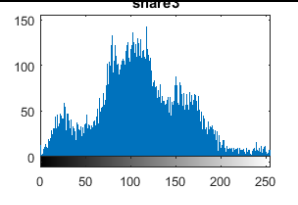
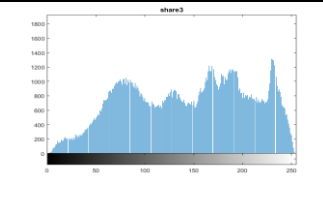
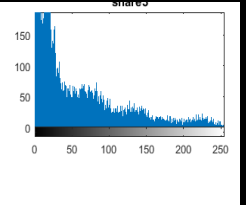



Original Image				
Histogram of original image				
Encrypt- ed Image				
Histogram of Cipher image				
Decrypt- ed Image				
Histogram of Decipher image				
Analysis	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB	MSE = 0.0 PSNR =Inf dB

Table 4.3: (PSNR & MSE) Values for Standard Images and Various Cipher Images Using Different Secret Keys.

Images	Key	Khadiga	Cipher456	Khadiga1
	Khadiga	MSE = 0.0 PSNR=Inf dB	MSE = 5256 PSNR =10.8 dB	MSE = 5338.9 PSNR =10.82 dB
	Cipher456	MSE = 5256 PSNR =10.8 dB	MSE = 0.0 PSNR=Inf dB	MSE = 5285.9 PSNR=10.86
	Khadiga1	MSE = 5338.9 PSNR =10.82 dB	MSE = 5285.9 PSNR=10.86	MSE = 0.0 PSNR=Inf dB
	Khadiga	MSE = 0.0 PSNR=Inf dB	MSE = 3515.89 PSNR =12.63 dB	MSE = 3546.9 PSNR =12.59 dB
	Cipher456	MSE = 3515.89 PSNR =12.63 dB	MSE = 0.0 PSNR=Inf dB	MSE = 3557.66 PSNR =12.58 dB
	Khadiga1	MSE = 3546.9 PSNR =12.59 dB	MSE = 3557.66 PSNR =12.58 dB	MSE = 0.0 PSNR=Inf dB
	Khadiga	MSE = 0.0 PSNR=Inf dB	MSE = 3185.43 PSNR =13.0 dB	MSE = 3146.99 PSNR =13.11 dB
	Cipher456	MSE = 3185.43 PSNR =13.0 dB	MSE = 0.0 PSNR=Inf dB	MSE = 3211.99 PSNR =13.02 dB
	Khadiga1	MSE = 3146.99 PSNR =13.11 dB	MSE = 3211.99 PSNR =13.02 dB	MSE = 0.0 PSNR=Inf dB

4.5 Validation and discution:

Showed the results in Tables [4.3] the PSNR values showing further information on the diffusion aspect using different keys on various standard images.in addition, a comparison of plain image and cipher image histograms is shown in Tables[4.1, 4.2].

4.6 Summary:

Showned the results in Table [4.1] the change in images to using Different Secret Keys, analyzes demonstrated by (PSNR) that little change in the secret key leads to a very big change in the image.

Chapter 5

Conclusion and Recommendations

5.1 Conclusion:

The classic Playfair Cipher can only be useful for a plain-text consisting of alphabets. However, a number of recently proposed extensions succeeded to encrypt alphanumeric data using different approaches.

In this research a new method for encrypting images using playfair algorithm is introduced. So, instead of the classical 5×5 matrix, the new method constructs 16×16 key matrix for a better alignment with image pixel data. In addition, an XOR procedure has been adopted for a more secure and yet scrambled results.

The experimental results showed that the key space of the new technique makes it hard for the attacker to perform a frequency analysis in tables [4.1,4.3] based on the used pixel digraphs.

Analyzes demonstrated that little change in the secret key leads to a very big change in the image. Obviously the results the randomness of the resultant ciphered images, PSNR values and histogram comparisons were also deployed to show the robustness of the new cipher.

5.2 Recommendations:

This thesis suggested in the near future, implement Modified playfair for encryption and decryption audio, video and other media, also using any other techniques for implemented.

References:

- [1] Enhancing the Security of Playfair Square Cipher by Double Substitution and Transposition techniques Jawad Ahmad Dar1 , Amit Verma
- [2] A Modified Version of Playfair Cipher Using 7x4 Matrix. August 2013. Alam, A. Aftab; Khalid, B. Shah; Salam, C. Muhammad
- [3] Enhanced the Security of Playfair Technique using Excess 3 Code (XS3) and Caesar Cipher. October 2014. Zubair Iqbal Bhumika Gupta, Kamal Kr. Gola Prachi Gupta
- [4] S. Hamad, A Novel Implementation of an Extended 8x8 Playfair Cipher Using Interweaving on DNA-encoded Data, (IJECE) 4(1) (2014).
- [5] A Novel Approach to Security using Extended Playfair Cipher, Shiv Shakti Srivastava, Nitin Gupta, International Journal of Computer Applications (0975 – 8887) Volume 20– No.6, April 2011.
- [6] William Stallings, “Cryptography and Network Security: Principles and Practice”, 4th Edition, Prentice Hall, 2006.
- [7] Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C (cloth) (Publisher: John Wiley & Sons, Inc.) Author(s): Bruce Schneier
- [8] Enhanced the Security of Playfair Technique using Excess 3 Code (XS3) and Caesar Cipher .Ravindra Babu K, S.Uday Kumar, A. Vinay Babu, I.V.N.S. Aditya, P.Komuraiah, “An Extension to Traditional Playfair Cryptographic Method”. International Journal of Computer Applications (0975 – 8887), Volume 17- No.5, March 2011.
- [9] Ramaraju PV, Nagaraju G, Chaitanya RK. Image Encryption and Decryption using Advanced Encryption Algorithm. Discovery, 2015, 29(107), 22-28
- [10] A Modified Version Of Extended Playfair Cipher (8x8). ¹Gaurav Shrivastava*, ²Manoj Chouhan, ³Manoj Dhawan ^{1,2,3} Assistant Professor, Department of Information Technology , Shri Vaishnav Institute of Technology & Science, Indore, Madhya Pradesh, INDIA.
- [11] Extension of Playfair Cipher using 16X16 Matrix. S.S.Dhenakaran, PhD. M. Ilayaraja Assistant Professor , Computer Science and Engineering & Research Scholar Alagappa University, Karaikudi, India

Appendix

PSNR and MSE:

```
L=(0:255);%
pixel_max = (L-1); % setting the maximum value that a pixel can
assume
% comment the following two lines if the frames are already in
YCbCr
img1= imread('1.bmp');
img2=imread('111.bmp');
img1 = rgb2ycbcr(img1); % converting from RGB to YCbCr
img2 = rgb2ycbcr(img2); % converting from RGB to YCbCr
img1 = img1(:,:,1); % extracting the luminance component (y)
img2 = img2(:,:,1); % extracting the luminance component (Y)
img1 = img1(:); % converts a matrix into a monodimensional array
img2 = img2(:); % converts a matrix into a monodimensional array
x=0;
img1=double(img1);
img2=double(img2);
x=(img1-img2).^2;
mse=mean(x); % here is the MSE
psnr=10*log10(((pixel_max).^2)/(mse)); % and here is the PSNR
fprintf('\nMSE: %f ', mse);
fprintf('\nPSNR: %f dB', psnr);
```

Histogram:

```
x=imread('202020.bmp');
imshow(x);
a_gray = rgb2gray(x);
imhist(a_gray);title('share3');
%subplot(1,2,1), imshow(m);
%figure;
%subplot(1,2,1), imshow(x);
%subplot(1,2,2), imshow(x);
%imh=imadjust(x,[0.3,0.6],[0.1,1.0]);
%imh1=histeq(x);
%figure;
%subplot(2,2,1), imshow(imh);title('stret');
%subplot(2,2,2), imshow(imh);
```

Table : other encrypted images

Images	Histogram	Encrypted
