VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belgaum-590018

A Mini Project Report on

"SECURE TRANSMISSION OF MEDICAL IMAGES USING CRYPTOGRAPHY TECHNIQUES"

Submitted in fulfillment of the Requirements for the VI Semester of the Degree of

Bachelor of Engineering In
Electronics and Communication Engineering

By:

MEGHANA SINGH D	1CR20EC087
MADHUSHREE L	1CR20EC078

Under the Guidance of
Dr. Pappa M
Associate Professor,
Dept. of Electronics and Communication Engineering

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI,
BANGALORE-560037

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CELEBRATING 25 YEARS

CMRIT

CMR INSTITUTE OF TECHNOLOGY, BENGALURU.

ACCREDITED WITH A+ GRADE BY NAAC

CERTIFICATE

This is to certify that the Mini Project Report entitled "Secure transmission of medical images using Cryptographic Techniques", prepared by Meghana Singh D, USN ICR20EC087, Madhu Shree L, USN: 1CR20EC078, a bona fide student of CMR Institute of Technology, Bengaluru in partial fulfilment of the requirements for the award of Bachelor of Engineering in Electronics and Communication Engineering of the Visvesvaraya Technological University, Belagavi-590018 during the academic year 2022-23. This is certified that all the corrections and suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The Mini Project has been approved as it satisfies the academic requirements prescribed for the said degree.

----------------------	-----------------------	----------------------
Signature of Guide	**Signature of HOD**	**Signature of Principal**
Dr. Pappa M	**Dr. R Elumalai**	**Dr. Sanjay Jain**
Associate Professor	**Professor & HOD**	**Principal**
Dept. of ECE, CMRIT	**Dept. of ECE, CMRIT**	**CMRIT**

External Viva

Name of the examiners Signature with date

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose consistent guidance and encouragement crowned our efforts with success.

We consider it as our privilege to express the gratitude to all those who guided in the completion of the project. We express my gratitude to Principal, **Dr. Sanjay Jain**, for having provided me the golden opportunity to undertake this project work in their esteemed organization.

We sincerely thank **Dr. R. Elumalai**, HOD, Department of Electronics and Communication Engineering, CMR Institute of Technology for the immense support given to me.

We express our gratitude to our project guide **Dr. Pappa M**, Associate Professor, Department of Electronics and Communication Engineering, CMR Institute of Technology for their support, guidance and suggestions throughout the project work.

Last but not the least, heartful thanks to our parents and friends for their support. Above all, We thank the Lord Almighty for His grace on us to succeed in this endeavor.

MEGHANA SINGH D
(1CR20EC087)
MADHU SHREE L
(1CR20EC078)

ABSTRACT

Image encryption plays a crucial role in ensuring the security and confidentiality of sensitive visual information during transmission and storage. Chaos-based encryption techniques have gained significant attention due to their ability to provide robust and efficient encryption algorithms. This abstract presents an overview of image encryption and decryption methods based on chaos systems.

Chaos systems exhibit highly sensitive dependence on initial conditions, as even small changes in the system's parameters can lead to substantial variations in the output. This property makes chaos systems suitable for developing encryption algorithms that provide a high level of security. In image encryption, chaos systems are utilized to generate pseudorandom sequences that are employed to scramble the pixel values of an image.

The encryption process involves two main stages: key generation and image scrambling. Firstly, a chaotic map generates a sequence of keys based on the initial conditions and control parameters. These keys are used to establish the encryption key for the subsequent encryption and decryption operations. The generated key sequence possesses the properties of unpredictability, nonlinearity, and sensitivity, ensuring a secure encryption scheme. In the image scrambling stage, chaos-based algorithms modify the pixel positions and values of the image using the generated key sequence. Common techniques include permutation and diffusion operations, which shuffle the image pixels and spread the pixel values across the image, respectively.

This process introduces confusion and diffusion to the image, making it highly resistant to attacks such as statistical analysis, differential analysis, and brute-force attempts. The decryption process utilizes the same chaotic map, initial conditions, and control parameters to regenerate the key sequence. The scrambled image is then subjected to reverse operations, including permutation and diffusion, using the regenerated key sequence. By applying these inverse operations, the original image can be reconstructed accurately, provided the correct encryption key is used. The advantages of chaos-based image encryption include simplicity, high encryption speed, resistance to attacks, and compatibility with various image formats.

Keywords: Image encryption, image decryption, chaos system, chaotic map, key generation, pixel scrambling, permutation, diffusion, security

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

Image encryption and decryption based on chaos systems is a fascinating field that combines the principles of chaos theory with the security of image data. Chaos theory deals with the study of complex, unpredictable systems that exhibit sensitive dependence on initial conditions. By harnessing the properties of chaotic systems, image encryption algorithms can create a high level of randomness and nonlinearity, which are essential for ensuring secure image communication. The encryption process in chaos-based image encryption involves converting the plain image into a cipher image using chaotic maps or systems. Decryption, on the other hand, involves reversing the encryption process to recover the original image from the cipher image. The decryption algorithm utilizes the same chaotic system and associated parameters used in encryption, along with the correct decryption key, to reconstruct the original image accurately. By applying the inverse chaotic transformations, the cipher image is transformed back into its original form. Chaos-based image encryption offers several advantages over traditional encryption techniques. Firstly, chaotic systems provide a high level of randomness, making it difficult for attackers to predict the encryption process or identify patterns in the encrypted image. Secondly, chaos-based encryption algorithms can achieve a large key space, ensuring strong encryption. Additionally, these algorithms are computationally efficient, allowing for real-time encryption and decryption of images. However, it's worth noting that chaos-based image encryption also faces certain challenges. The sensitivity to initial conditions and parameters in chaotic systems can lead to security vulnerabilities if an attacker manages to determine the precise system parameters. Moreover, the robustness of chaosbased encryption against various attacks, such as chosen-plaintext attacks or statistical attacks, needs to be thoroughly analyzed to ensure its effectiveness in practical scenarios. In conclusion, image encryption and decryption based on chaos systems leverages the principles of chaos theory to provide secure and efficient methods for protecting image data. This field continues to evolve as researchers explore new techniques and algorithms to enhance the security and robustness of chaos-based image encryption systems.

In the modern era of medicine, technological advancements have revolutionized the way healthcare professionals diagnose and treat various conditions. Among these advancements, medical imaging has emerged as a vital tool in the diagnostic and treatment process. With the increasing reliance on medical images such as X-rays, CT scans, MRIs, and ultrasounds, the importance of storing and preserving these images has become paramount. In this introduction, we will explore the compelling reasons why medical images should be saved, emphasizing their role in advancing healthcare and improving patient outcomes.

1. Continuity of Care: Medical images serve as crucial references for physicians and specialists, ensuring continuity of care for patients. By preserving and saving medical images, healthcare providers can access previous records and compare them with new scans, enabling accurate diagnosis, effective treatment planning, and monitoring disease progression. This continuity allows medical professionals to make informed decisions, reducing the risk of misdiagnosis or unnecessary interventions.

2. Enhanced Collaboration: Saving medical images facilitates seamless collaboration and knowledge sharing among healthcare professionals. By storing and archiving images in secure and accessible databases, multiple specialists can review and analyze the same images, leading to a collective and comprehensive assessment of a patient's condition. This collaborative approach can foster interdisciplinary discussions, expedite diagnosis, and result in more effective treatment strategies.

3. Longitudinal Analysis: Medical images stored over time create a valuable resource for conducting longitudinal studies and research. By analyzing images from various time points, researchers can gain insights into disease progression, treatment efficacy, and long-term patient outcomes. These longitudinal studies contribute to the development of evidence-based medicine, supporting the refinement of existing treatment protocols and the discovery of novel therapeutic approaches.

Chapter 2

METHODOLOGY

Encrypting and decrypting images based on chaos systems can provide a level of security and protection for sensitive visual data. MATLAB provides a convenient platform for implementing such encryption and decryption algorithms. Here's a general methodology for image encryption and decryption using a chaos system in MATLAB:

1. Chaos System Selection: Choose a suitable chaos system that exhibits chaotic behavior, such as the Logistic map, Lorenz system, or Henon map. You can define the equations of the chosen chaos system within MATLAB.

2. Key Generation: Generate a secret key using the chaos system. The key should be a set of initial conditions or parameters for the chaos system. The key generation process should ensure that the key is unpredictable and has sufficient entropy.

3. Image Preprocessing: Load the image you want to encrypt and preprocess it as necessary. Convert the image to grayscale or RGB, depending on your encryption requirements. Resize the image if needed to match the chaos system's dimensionality.

4. Image Encryption: Perform the encryption process using the chaos system and the generated secret key. The general steps involved in the encryption process are as follows: a. Initialize the chaos system with the secret key. b. Iterate the chaos system for a certain number of iterations to achieve the chaotic state. c. Perform a pixel-wise or block-wise transformation on the image using the chaotic values generated by the chaos system. d. Apply additional confusion and diffusion operations to enhance the security of the encrypted image. e. Repeat the above steps for a sufficient number of rounds to strengthen the encryption.

5. Image Decryption: To decrypt the encrypted image, you need to perform the reverse operations of the encryption process. The decryption process involves the following steps: a. Initialize the chaos system with the same secret key used during encryption. b. Iterate the chaos system to generate the chaotic values. c. Reverse the confusion and diffusion operations applied during encryption. d. Restore the original image by applying the inverse of the pixel-wise or block-wise transformation.

6. Visualization and Analysis: Display the encrypted and decrypted images using MATLAB's image processing functions. Compare the original image with the decrypted image to evaluate the quality of the decryption process. Calculate relevant metrics, such as peak signal-to-noise ratio (PSNR) or structural similarity index (SSIM), to assess the quality and fidelity of the decrypted image.

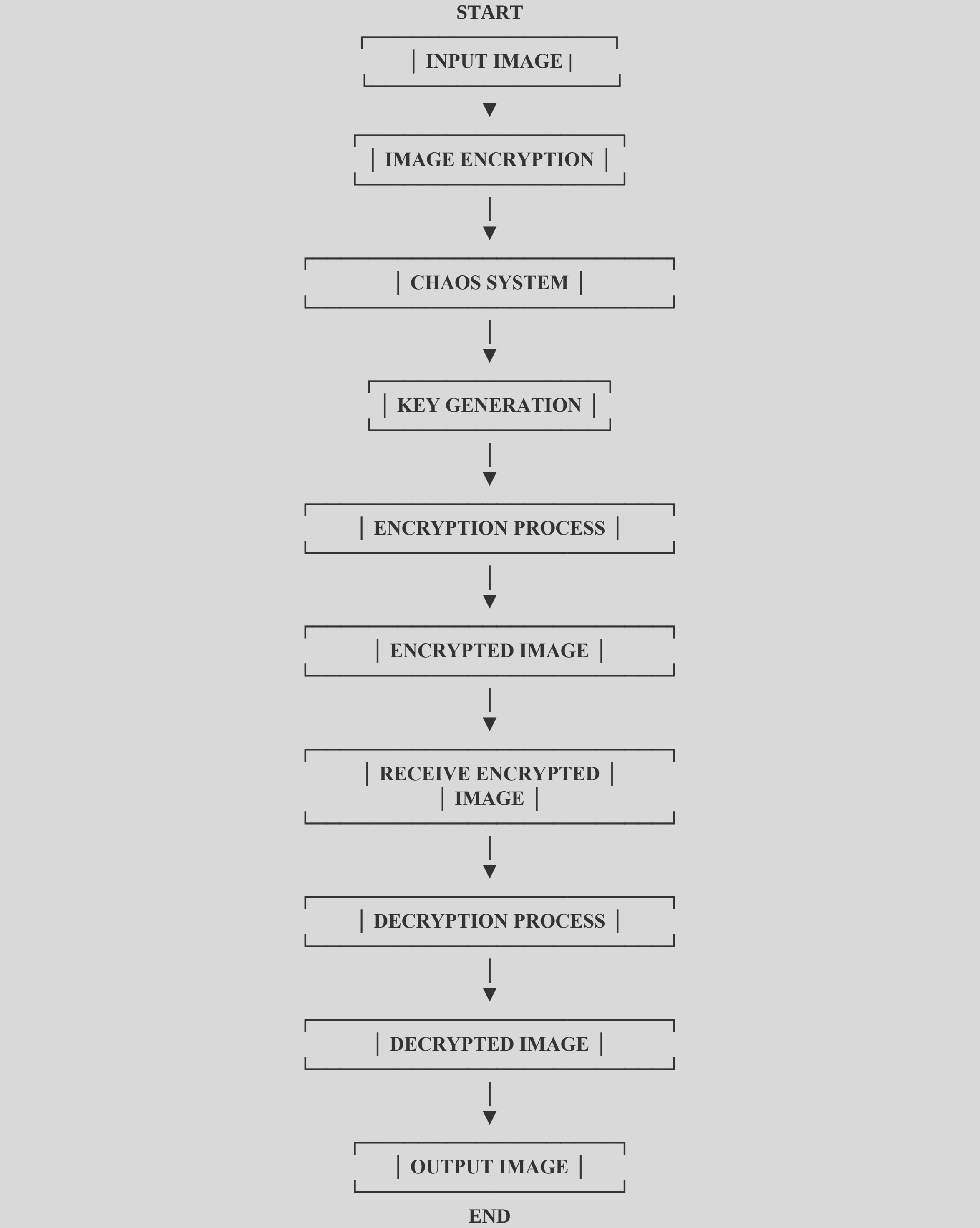
Selection of Cryptographic Algorithms: The first step in implementing secure transmission is selecting appropriate cryptographic algorithms. This involves choosing algorithms for encryption and decryption, as well as for key exchange. Commonly used encryption algorithms include Advanced Encryption Standard (AES), Triple Data Encryption Standard (3DES), or Rivest-Shamir-Adleman (RSA) encryption. These algorithms ensure the confidentiality of the medical images by converting them into unreadable ciphertext.

Key Management: Effective key management is crucial for secure transmission. Symmetric encryption algorithms such as AES and 3DES require a shared secret key for both encryption and decryption. Asymmetric encryption algorithms like RSA use public-private key pairs. In the case of medical image transmission, a secure key exchange mechanism is necessary to ensure that only authorized parties have access to the decryption keys. Key management systems, such as the use of secure key servers or the implementation of public key infrastructure (PKI), can help establish and distribute encryption keys securely.

Encryption of Medical Images: Once the cryptographic algorithms and keys are established, the medical images should be encrypted before transmission. The selected encryption algorithm, along with the encryption key, is applied to convert the images into ciphertext. This process ensures that even if the transmitted data is intercepted, it remains unreadable to unauthorized individuals. It is important to ensure that the encryption process is efficient and does not compromise the integrity or quality of the medical images.

Decryption at the Receiving End: Upon receiving the encrypted medical images, decryption is performed at the authorized destination. The decryption process involves using the appropriate decryption algorithm and the corresponding decryption key. This step restores the medical images to their original format, making them accessible for viewing and analysis by authorized healthcare professionals.

[2.1] FLOWCHART



This flowchart illustrates the general steps involved in the process of image encryption and decryption based on a chaos system.

The key steps include:

1. Input Image: Start with the original image you want to encrypt.

2. Image Encryption: Perform image encryption using a chosen chaos system.

3. Chaos System: Implement the selected chaos system to generate pseudo-random sequences or parameters required for encryption.

4. Key Generation: Generate the encryption key based on the chaotic sequences or parameters obtained from the chaos system.

5. Encryption Process: Apply the encryption algorithm using the generated key to encrypt the input image.

6. Encrypted Image: Obtain the encrypted image as the output of the encryption process.

7. Image Transmission: Transmit the encrypted image through the desired communication channel or medium.

8. Decryption Process: Apply the decryption algorithm using the same chaos system and key to decrypt the received encrypted image.

9. Decrypted Image: Obtain the decrypted image as the output of the decryption process.

10. Output Image: Display or save the decrypted image as the final result.

Chapter 3

IMPLEMENTATION

Encryption Algorithm:

1. Input: Image I (M x N x nc), Number of Rounds rounds
2. Initialize an empty encrypted image I_enc with the same dimensions as I.
3. For each color channel i from 1 to nc: a. Perform the encryption process for the i-th color channel of I using the specified encryption algorithm: [i]. Divide the image into blocks (e.g., 2x2 or 4x4) for diffusion. [ii]. Apply a series of encryption operations (e.g., permutation, substitution, diffusion) on each block for rounds iterations. [iii]. Store the encryption key or any necessary parameters (e.g., chaos system states) for decryption. [iv]. Assign the encrypted blocks to the corresponding locations in I_enc for the i-th color channel.
4. Output: Encrypted image I_enc.

Decryption Algorithm:

1. Input: Encrypted Image I_enc (M x N x nc), Encryption Parameters (e.g., SX containing chaos system states)
2. Initialize an empty decrypted image I_dec with the same dimensions as I_enc.
3. For each color channel i from 1 to nc: a. Retrieve the encryption parameters (e.g., chaos system states) for decryption. b. Perform the decryption process for the i-th color channel of I_enc using the specified decryption algorithm: [i]. Divide the encrypted image into blocks (matching the encryption process). [ii]. Apply a series of decryption operations (inverse of encryption operations) on each block for rounds iterations, using the stored parameters. [iii]. Assign the decrypted blocks to the corresponding locations in I_dec for the i-th color channel.
4. Output: Decrypted image I_dec.

[3.1] PROGRAM

```
clc
clear all
close all
I=imread('medical_image.jpg');
[M,N,nc]=size(I);
if mod(M,2)==1
 M=M+1;
end
if mod(N,2)==1
 N=N+1;
end
I=imresize(I,[M N]);
subplot(131)
imshow(I)
title('Original Image')
rounds=2;
for i=1:nc

[I_enc(:,:,i),SX{i}]=Encrypt(I(:,:,i)
,rounds);
end
subplot(132)
imshow(I_enc)
title('Encrypted Image')
for i=1:nc
I_dec(:,:,i)=Decrypt(I_enc(:,:,i),SX{
i});
end
subplot(133)
imshow(I_dec)
title('Decrypted Image')
y1=I(:);
y2=I_dec(:);
MSE=sum((y1-y2).^2)/length(y1)
impsnr=psnr(I_dec,I)
function [I_enc,SS]=Encrypt(I,rounds)
if nargin<1
 error('You must enter at least
the image')
elseif nargin<2
 rounds=2;
 warning('Rounds not found, we
will use 2 rounds')
end
rounds=ceil(rounds);
if rounds<1
error('Rounds is less than 1')
end
[M,N]=size(I);
% To Make Sure That image size is
even
if mod(M,2)==1
 I(M+1,:)=uint8(0);
 M=M+1;
end
if mod(N,2)==1
 I(:,N+1)=uint8(0);
 N=N+1;
end
MN=M*N;
for round_iter=1:rounds
% #2
P=double(I(:));
% #3
x=(sum(P)+MN)/(MN+(2^23));
for i=2:6
 x(i)=mod(x(i-1)*1e6,1);
```

```
end
% #sys 1
a=10;b=8/3;c=28;d=-1;e=8;r=3;
L=@(t,x)[a*(x(2)-x(1))+x(4)-x(5)-x(6)
 c*x(1)-x(2)-x(1)*x(3)
 -b*x(3)+x(1)*x(2)
 d*x(4)-x(2)*x(3)
 e*x(6)+x(3)*x(2)
 r*x(1)];
N0=.9865*MN/3;
MN3=ceil(MN/3);
[T,Y]=ode45(L,[N0 MN3],x);
% #4
L=Y(1:MN3,1:2:5);
L=L(:);L=L(1:MN);
% #5
[L2,S]=sort(L);
SS{round_iter}=S;
% #6
R=P(S);
% #7
R_=reshape(R,[M N]);
A=[89 55;55 34];
for i=1:2:M
 for j=1:2:N Cx=[R_(i,j) R_(i,j+1)
 R_(i+1,j) R_(i+1,j+1)];
 fz=Cx*A;
 C(i,j)=fz(1,1);
 C(i,j+1)=fz(1,2);
```

```
 C(i+1,j)=fz(2,1 C(i+1,j+1)=fz(2,2);
 end
end
I=mod(C,256);
P=I(:);
end
I_enc=uint8(I);
function I_dec=Decrypt(I_enc,SS)
if nargin<2
 error('You must enter all the data
: Encrypted Image and the Key')
end
if isa(SS,'cell')~=1
 error('The Key must be Cell')
end
[M,N,nc]=size(I_enc);
C=double(I_enc);
A_=[34 -55;-55 89];
rounds=length(SS);
for round_iter=rounds:-1:1
 for i=1:2:M
 for j=1:2:N
 Cx=[C(i,j) C(i,j+1)
 C(i+1,j) C(i+1,j+1)];
 fz=Cx*A_;
 D(i,j)=fz(1,1);
 D(i,j+1)=fz(1,2);
 D(i+1,j)=fz(2,1);
 D(i+1,j+1)=fz(2,2);
```

CHAPTER 4

RESULT

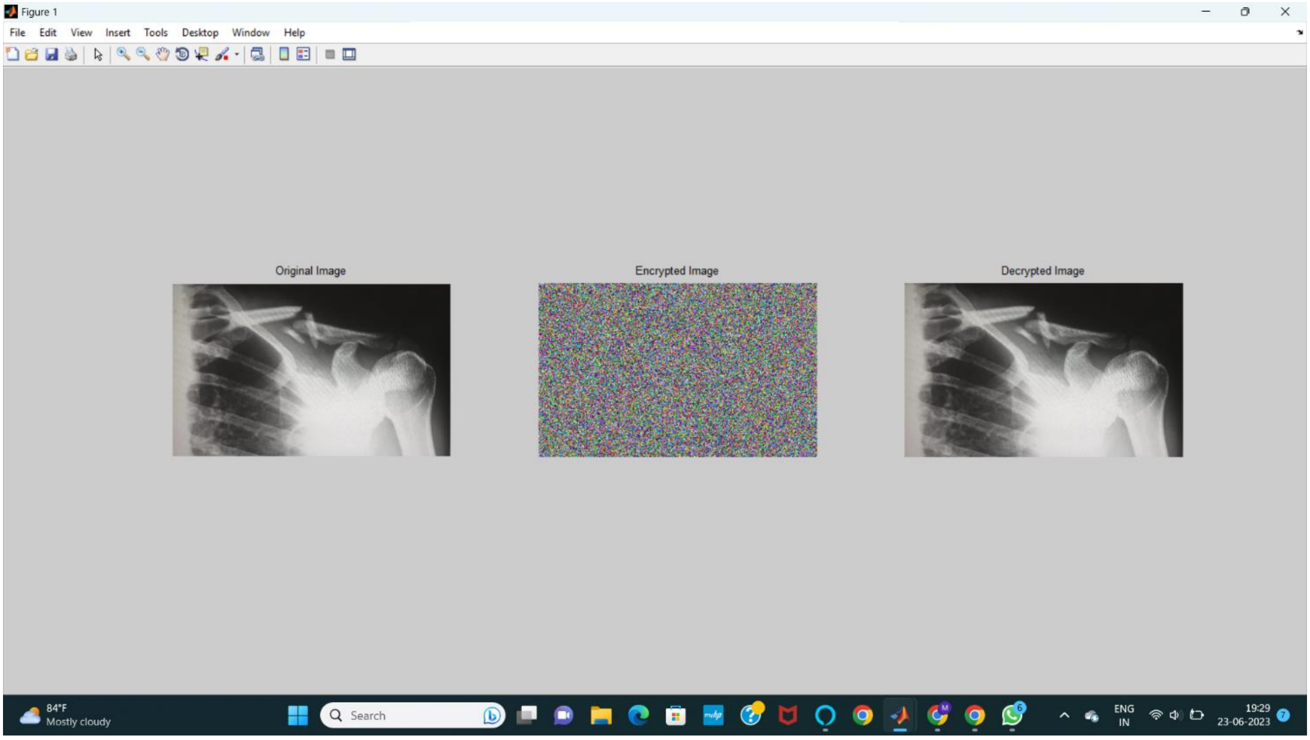


Fig. 5.1 Encrypted and Decrypted form of Original image

Software used: Matlab R2014a

Encryption is the process of converting the original image into an encrypted form to protect its contents from unauthorized access. It involves applying a cryptographic algorithm that transforms the image data into a scrambled or unreadable format. The encryption algorithm typically uses a secret key, known only to the authorized recipient, to perform the encryption.

Decryption is the reverse process of encryption. It involves applying the inverse of the encryption algorithm, along with the correct decryption key, to the ciphertext. This process retrieves the original image from its encrypted form, restoring it to its original content and format.

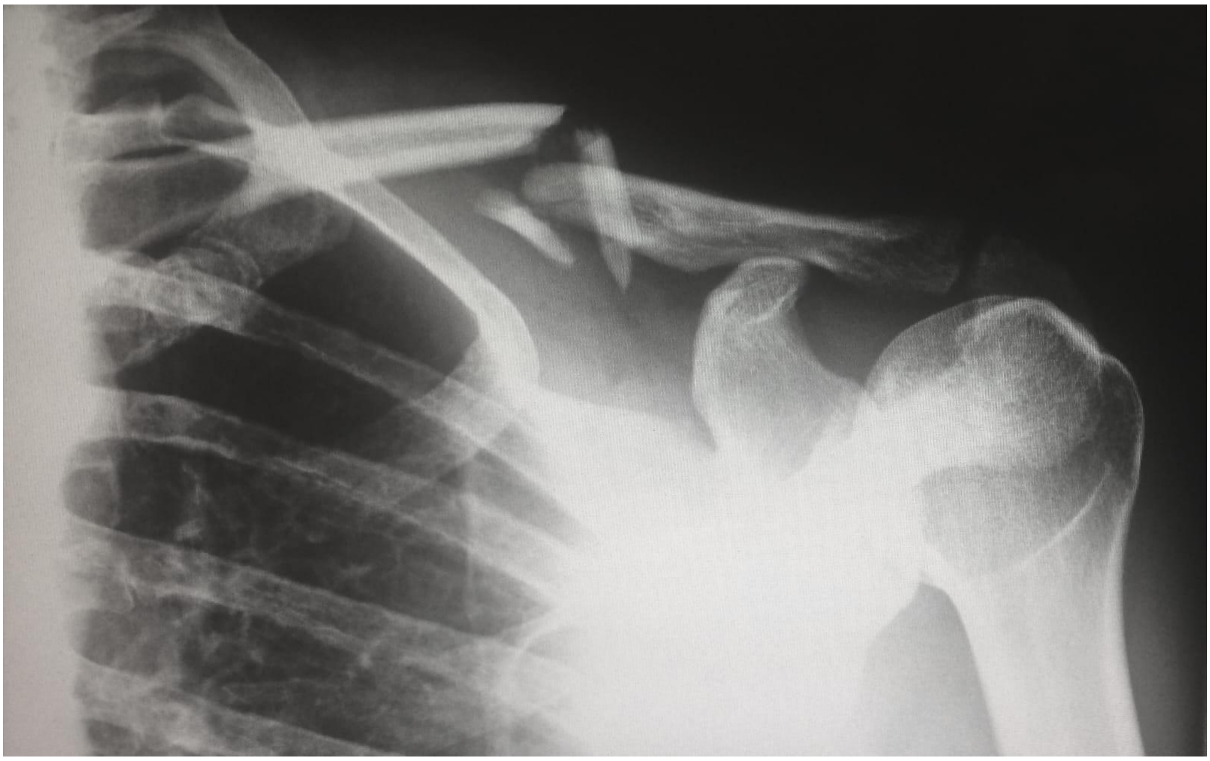


Fig 5.2 Medical Image used in Encryption and Decryption process

Type of file : JPG
File Size : 98.4 Kb (1,00,779 Bytes)
Size on disk : 100 Kb (1,02,700 Bytes)

Medical images often contain sensitive and personal information about patients, including their medical conditions, diagnostic tests, and anatomical details. Protecting the privacy of patients is a fundamental ethical and legal requirement.

Unauthorized access to medical images can lead to breaches of patient confidentiality and potential harm to individuals, including identity theft and discrimination. Medical images are often used for research, collaboration, and education purposes. Securing these images ensures that they are only accessible to authorized individuals or organizations.

CHAPTER 5

FUTURE SCOPE

The secure transmission of medical images by encryption and decryption using chaos systems in MATLAB has promising future applications. Here are some potential areas of scope: Privacy and Confidentiality: Medical images contain sensitive patient information, and ensuring their privacy and confidentiality during transmission is critical. Encryption techniques based on chaos systems can provide an additional layer of security, making it difficult for unauthorized individuals to access or interpret the transmitted images. Telemedicine and Remote Diagnostics: With the increasing adoption of telemedicine and remote diagnostics, the secure transmission of medical images becomes crucial. Encryption techniques can help protect the integrity and confidentiality of the images, enabling healthcare professionals to diagnose and treat patients remotely while maintaining privacy.

CONCLUSION

The secure transmission of medical images is crucial to protect patient privacy and ensure the integrity of sensitive healthcare data. Encryption and decryption techniques play a vital role in achieving this security. One approach is to utilize chaos systems for encryption and decryption processes, which can provide robust and efficient encryption algorithms. MATLAB, a widely used programming language, provides a suitable environment for implementing such algorithms. In conclusion, the secure transmission of medical images can be achieved by employing encryption and decryption techniques based on chaos systems, implemented using MATLAB. The decryption process, using the same chaos system algorithm, allows authorized parties to retrieve and view the original medical images. This methodology helps maintain the confidentiality, integrity, and privacy of sensitive medical data.

CHAPTER 6

REFERENCES

[1] Abraham Sinkov, Elementary Cryptanalysis: A Mathematical Approach, Mathematical Association of America, 1966. ISBN 0-88385-622-0

[2] Kester, Quist-Aphetsi. "A cryptosystem based on Vigenère cipher with varying key." International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 1, no. 10 (2012): pp-108.

[3] Kester, Q. A., & Koumadi, K. M. (2012, October). Cryptographic technique for image encryption based on the RGB pixel displacement. In Adaptive Science & Technology (ICAST), 2012IEEE 4th International Conference on (pp. 74-77). IEEE.

[4] Kester, Q. A., & Danquah, P. (2012, October). A novel cryptographic key technique. In Adaptive Science & Technology (ICAST), 2012 IEEE 4th International Conference on (pp. 7073). IEEE.

[5] Moni Naor and Adi Shamir, ”Visual cryptography”, in Proceedings of Advances in Cryptology EUROCRYPT 94, LNCS Vol. 950, pages 1-12. Springer-Verlag, 1994.

[6] Zhi Zhou; Arce, G.R.; Di Crescenzo, G., "Halftone visual cryptography," Image Processing, IEEE Transactions on , vol.15,no.8, pp.2441,2453, Aug. 2006.

[7] Chandramathi, S., Ramesh Kumar, R., Suresh, R., & Harish, S.(2010). An overview of visual cryptography. International Journal of Computational Intelligence Techniques, ISSN, 09760466.