2022 44th Annual International Conference of
the IEEE Engineering in Medicine & Biology Society (EMBC)
Scottish Event Campus, Glasgow, UK, July 11-15, 2022

2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) | 978-1-7281-2782-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/EMBC48229.2022.9871499

"From Kilobytes to Kilodaltons": A Novel Algorithm for Medical Image Encryption based on the Central Dogma of Molecular Biology

Arjav Gupta, Srinivas Sampalli, *Member, IEEE*

***Abstract*— With the continued integration of technology in medicine, large amounts of patient data are often vulnerable to cyber-attacks. Medical data must be secured, however traditional cryptographic algorithms are inapplicable to medical images due to factors such as bulk data capacity, strong correlation among adjacent pixels, and high redundancy. To address the need for new medical image encryption algorithms, a novel approach based on the central dogma of molecular biology is proposed. The resulting algorithm has a linear runtime complexity, and is resistant to brute force, differential and statistical attacks. The algorithm advances the state-of-the-art in DNA-based image encryption and surpasses recent approaches in medical image encryption in its defence against cyber-attacks.**

***Clinical Relevance*— Secure data transmission and storage is critical for patient privacy. This algorithm increases the security of patient imaging when compared to image encryption algorithms in literature.**

I. Introduction

With the continued progression of medical technology, patient data is more frequently stored and transferred digitally [1]. This makes patient information vulnerable to a variety of threats [2], and an attack on such data can often lead to severe consequences for the healthcare system. Medical data must be encrypted in order to prevent malicious use. For medical images specifically, traditional cryptographic algorithms are inappropriate due to their inherent features such as bulk data capacity, strong correlation among adjacent pixels, and high redundancy [3]. To address the need for new medical image encryption algorithms, we propose a novel approach based on the central dogma of molecular biology, which is the theory that deoxyribonucleic acid (DNA) is transcribed into ribonucleic acid (RNA) and then translated into a protein consisting of amino acids in living organisms [4]. This is illustrated in Figure 1.

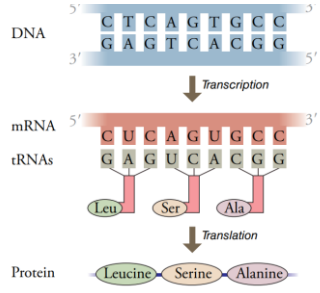

Figure 1. The central dogma of molecular biology [4].

The transcription of DNA into RNA is one-to-one, but the translation of RNA into protein uses degenerate triplet "codons", where three RNA nucleotides translate to one amino acid in a protein. The degenerate codon table is shown in Figure 2. Cryptography researchers have been utilizing the inherent features of DNA in order to create new encryption algorithms which learn from the evolved processes of the natural world [3], [5], [6], [7], [8], [9].

The novelty of our approach is our usage of a degenerate genetic code with a dynamic codon table in order to increase security. Common cryptography operations such as substitutions and permutations are used [10]. Our algorithm also uses pseudorandom number generators (PRNGs) for several cryptographic steps. PRNGs use mathematical formulae to produce seemingly random sequences of numbers, with the ability to re-generate the same sequence using an identical "seed" value [11].

The rest of the paper is organized as follows. Section II presents our literature survey. Section III outlines our encryption method. Section IV demonstrates our encryption performance and results. Section V presents our discussion, future work and conclusion.

II. Literature Survey

Previous literature in this field is varied in its biological inspiration. Ning [6] proposed one of the earliest pseudo-DNA cryptography approaches, converting plaintext binary values into a protein sequence form. The algorithm was strong against brute-force attacks, but weak against chosen plaintext attacks [6]. Hossain et al. [12] proposed a method which converts plaintext into a DNA form using a dynamic DNA sequence table. The method is inefficient, but the concept of a dynamic DNA conversion table was novel.

FIRST POSITION (5′ END)	SECOND POSITION								THIRD POSITION (3′ END)
	U		C		A		G		
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC	Phe	UCC	Ser	UAC	Tyr	UGC	Cys	C
	UUA	Leu	UCA	Ser	UAA	Stop	UGA	Stop	A
	UUG	Leu	UCG	Ser	UAG	Stop	UGG	Trp	G
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg	U
	CUC	Leu	CCC	Pro	CAC	His	CGC	Arg	C
	CUA	Leu	CCA	Pro	CAA	Gln	CGA	Arg	A
	CUG	Leu	CCG	Pro	CAG	Gln	CGG	Arg	G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC	Ile	ACC	Thr	AAC	Asn	AGC	Ser	C
	AUA	Ile	ACA	Thr	AAA	Lys	AGA	Arg	A
	AUG	Met	ACG	Thr	AAG	Lys	AGG	Arg	G
G	GUU	Vae	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC	Val	GCC	Ala	GAC	Asp	GGC	Gly	C
	GUA	Val	GCA	Ala	GAA	Glu	GGA	Gly	A
	GUG	Val	GCG	Ala	GAG	Glu	GGG	Gly	G

Figure 2. The standard genetic codon table for mRNA translation into amino acids [4].

A. Gupta is with the Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 1W5 Canada (phone: 1-902-423-9115, e-mail: ar237808@dal.ca).

S. Sampalli is with the Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 1W5 Canada (phone: 1-902-494-1657, e-mail: srini@cs.dal.ca).

Many image-focused DNA encryption methods did not convert plaintext into a protein form, and therefore did not use the degeneracy of the genetic code [13], [14], [15]. The method proposed by Zhang et al. [5] was inspired by the translation step of the central dogma and demonstrated improved correlation coefficients in their cipher images. The DICOM image format was found to be the most popular in medicine [16].

A review of the literature revealed three research gaps: namely, a lack of fast DNA-based encryption methods, a lack of methods employing the degeneracy of the genetic code, and the security flaws in the DICOM image format. We have targeted these gaps to propose a novel encryption method.

III. METHODS

The overall idea behind this algorithm is to simulate the central dogma of molecular biology by converting data in a "nucleotide" form into a "protein" form, thereby encrypting it. A working example is shown in Figure 3. The first step is the conversion of digital information into a nucleotide sequence, so that DNA operations can be performed on it. 2-bit values can be used to represent the four different nucleotides (A, G, C or T). This conversion can be seen in most DNA-based cryptography literature as well [6], [7], [8].

Our algorithm is novel in its approach to several steps of central dogma-based encryption. In the binary-to-nucleotide conversion step, other researchers have generally used a single conversion between 2-bit values and nucleotides (e.g. A=00, C=01, G=10, T=11). However, there are 24 possibilities for this conversion, so our approach uses all 24 using a PRNG to determine which conversion to use for each 2-bit value. PRNG seed values are used as keys throughout the algorithm in order to reduce key size.

The next step in the algorithm is permutation of the nucleotide sequence. Permutation is critical for image encryption, as it aids in removing statistical patterns in the ciphertext.

The conversion of the nucleotide sequence into protein is another novel component of our algorithm. We use a dynamic conversion table, similar to the table used by Hossain et al. [12]. There are 64 possible 3-letter codons, and as such our table dynamically permutes all 64 possible codons for each plaintext. For simplicity and storage efficiency, 16 amino acid values are used instead of the 20 which are commonly present in the natural world. The 16 amino acids are also permuted by a PRNG and assigned randomly to codons in the table. This results in a table which contains 16 amino acids, with 4 codons assigned to each amino acid, in a pseudorandom arrangement.

The result of the translation step is a series of 4-bit amino acid values (for the 16 amino acids) and corresponding 2-bit degeneracy values to indicate which specific codons yielded the amino acids (for the 4 possible codons per amino acid). The amino acid values and degeneracy values are permuted and represented as binary values. This produces the final ciphertext. The decryption of a given ciphertext uses the same seed values for the PRNGs, reversing the overall process.

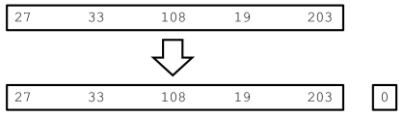

Figure 3a. Padding of file with '0' value bytes (File size must be divisible by 6 for later steps, this step is only executed if needed).

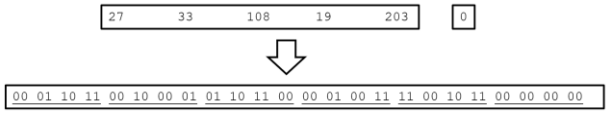

Figure 3b. Conversion of bytes into binary bits.

Figure 3c. Conversion of 2-bit numbers into nucleotides.

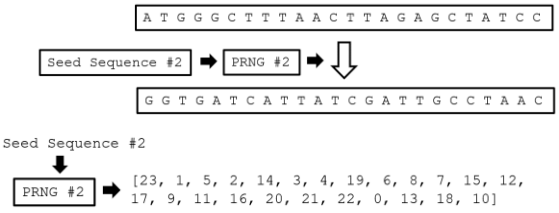

Figure 3d. Permutation of nucleotide sequence.

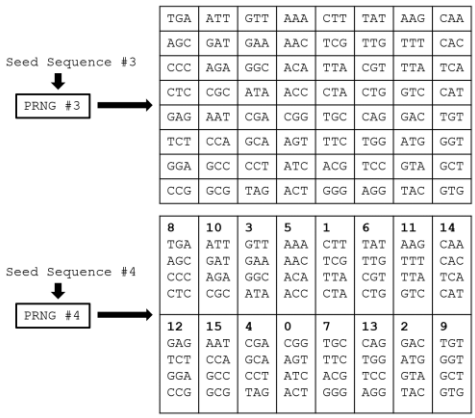

Figure 3e. Generation of codon table.

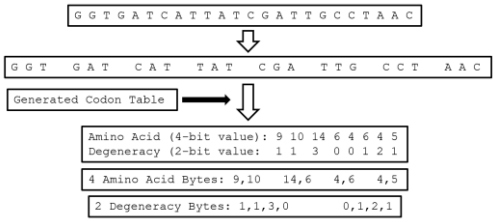

Figure 3f. Translation of nucleotide sequence into amino acid sequence.

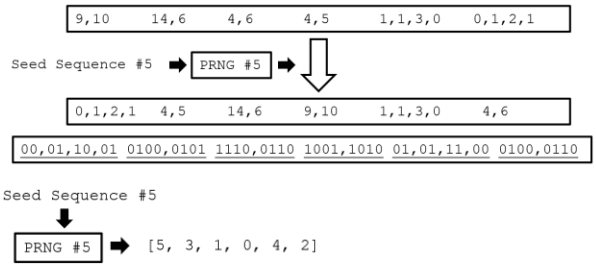

Figure 3g. Permutation of amino acid and degeneracy values, producing final ciphertext.

IV. Results

A. Experimental Setup and Test Dataset

The algorithm was implemented using the Java programming language on a desktop computer running Windows 10. The PRNG used in this implementation was the SecureRandom class in the Java security library, which used the "SHA1PRNG" algorithm [17]. Seed sequences were generated and used as keys. The permutation steps in the algorithm used the Fisher-Yates shuffle algorithm [18] to generate permutations from a PRNG output.

The Lena image, as a 512x512 .tiff file, was used as a benchmark image due to its ubiquitous usage in computer imaging research [19]. Five DICOM format images were found online from three different sources [20][21][22]. These files range in their content and file size. All images were converted into 8-bit grayscale pixel data using ImageJ [23], as this removes all non-image data from the DICOM format. This data can still be encrypted by the algorithm, but using grayscale representations made testing and comparisons more intuitive.

B. Performance

The encryption and decryption processes are both O*(n)* (linear run time complexity) for the proposed algorithm. File size vs. execution time was tested for the Java implementation of the algorithm, for both encryption and decryption of the five DICOM test images. The results of these tests can be found in Figure 4. Linear time complexity was demonstrated with increasing file sizes. The algorithm has a similar speed to Mondal and Mandal [8], showing that this technique is fast when compared to literature standards.

Memory usage vs. file size was also tested using the five DICOM images. The memory usage for encryption and decryption did not change significantly with changes in file size. The implementation reads files in small buffers, which are recycled sections of memory, rather than reading entire files into memory. As a result, the memory usage is constant regardless of file size. Most memory overhead was likely due to the Java virtual machine and automated garbage collector. A more efficient programming language such as C could be used to lower this memory overhead.

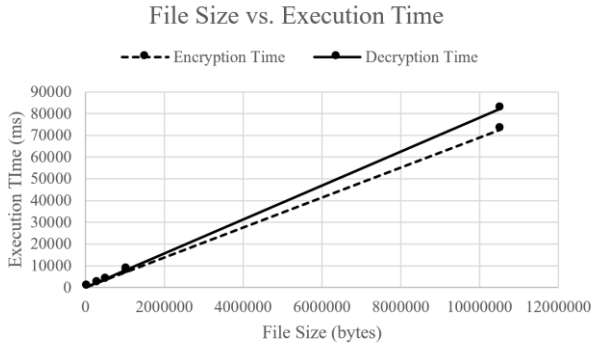

Figure 4. Plot of DICOM image file size (in bytes) vs. encryption and decryption time (in milliseconds)

C. Key Space and Sensitivity

The key for this algorithm includes five 8-byte PRNG seed sequences, as well as a "padding" value between 0 and 5. Overall, the key space is 2^{320} when this key is considered as one 40-byte key. The key space, as well as the permutation and substitution steps, defend this algorithm from brute force attacks.

Key sensitivity is the measurement of how sensitive a decryption process is to a slight change in key value [13]. The Lena image and three of the DICOM test images were used in this test. The images were encrypted using an original key, then decrypted using an altered key (incremented by 1). Pearson correlation values between the original images and key-sensitive decrypted images were taken. These correlations were very low, ranging from 0.04107 to 0.08792 for the various images. This demonstrates a high key sensitivity for our algorithm. The test image "Knee MRI 1" is shown in Figure 5. The histograms for the test image "Knee MRI 1" can be found in Figure 6, demonstrating key sensitivity.

D. Differential Attack Analysis

The Number of Pixel Change Rate (NPCR) value is defined as the percentage of different pixels between two encrypted images, where the two original images differ by only one pixel [13]. The ideal NPCR value is 100%. The NPCR values for Lena and three of the DICOM test images can be found in Table 1.

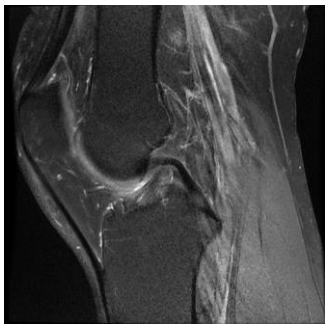

Figure 5. "Knee MRI 1" test image, converted from DICOM to TIFF format for figure [19]

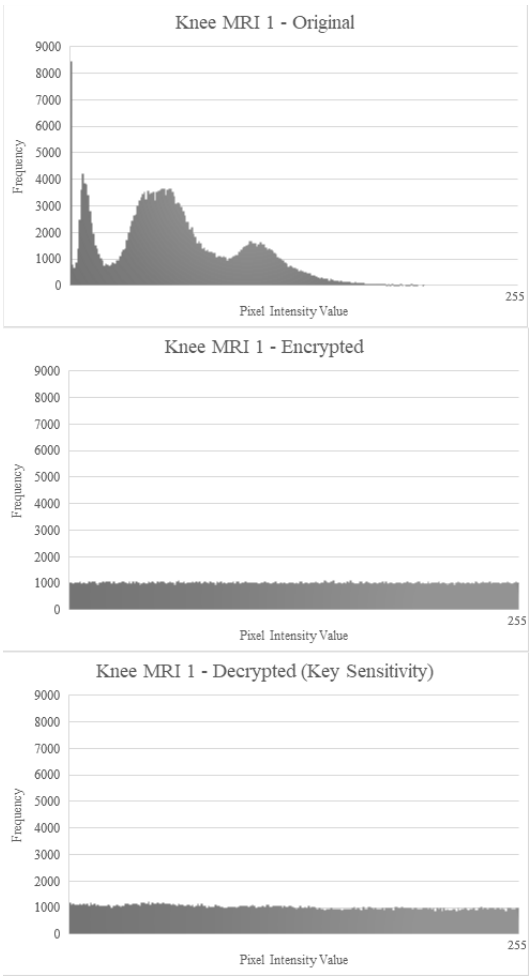

Figure 6. Original image, encrypted image and key-sensitive decrypted image pixel value histograms for Knee MRI 1 test image.

TABLE I. NPCR AND UACI VALUES FOR TEST IMAGES

Value	Test Image			
	Lena	***Knee MRI 1***	***Knee MRI 2***	***Brain MRI***
NPCR (%)	99.620	99.607	99.614	99.617
UACI (%)	33.428	33.510	33.588	33.520

The Uniform Average Changing Intensity (UACI) value is defined as the average intensity difference between two encrypted images, where the original two images differ by only one pixel [13]. Higher UACI values are ideal. Our UACI values for the Lena and three DICOM test images can be found in Table 1.

Belazi et al. [15] demonstrated mean NPCR and UACI values of 99.617 % and 33.475 % accordingly for their medical image encryption algorithm. This indicates our algorithm matches the standards of current medical image encryption literature, or exceeds it, in its defence against differential attacks.

E. Statistical Attack Analysis

Histograms can be used to analyze the distribution of pixel intensities present in an image. Non-encrypted images generally show patterns in their intensities, while encrypted images ideally have uniform distributions. Assuming an ideal encrypted image has a uniform distribution of pixel values, the probability of a given value occurring should be near equal between pixel values. Information entropy (IE) measures this probabilistic aspect, and has an ideal value of 8 for an 8-bit image [13]. Our IE values ranged between 7.996 and 7.999. The numerous results in Kaur and Kumar [13] had IE values which ranged between 7.3419 and 7.9996. With regards to medical image encryption literature, this study out-performs one of the most recent papers in this field by Akkasaligar and Biradar [14], as their IE values had a mean of 7.846. Our algorithm produces IE values which are similar to some literature values while exceeding many others.

A correlation coefficient value measures the similarity between adjacent pixels in an image. An original image would have strong correlations in the horizontal, vertical and diagonal directions. Encrypted images should have correlation values close to 0. Zhang et al. [5] stated that their encryption scheme achieved superior correlation coefficient values compared to most other schemes mentioned in their paper. Our algorithm produced improved correlation coefficient values when compared to their paper, as shown in Table 2. The corresponding p-values can be found in Table 3. Our correlation coefficient values surpassed image encryption literature, indicating strong statistical attack defence.

F. Bit Correct Ratio (BCR)

Bit Correct Ratio (BCR) calculates the difference between an original image and a decrypted image [13]. This ratio should be 1.0, indicating the images are identical and that the decryption algorithm is lossless. BCR values were 1.0 for all of our test images.

TABLE II. LENA 512X512 CORRELATION COEFFICIENT COMPARISON WITH ZHANG ET AL.

Encryption Approach	Correlation Coefficient Direction		
	Vertical	***Horizontal***	***Diagonal***
Our algorithm	0.002221943	-0.002121632	-0.000733059
Zhang et al. [5]	-0.0023	0.0105	0.0031

TABLE III. LENA 512X512 CORRELATION COEFFICIENT P-VALUES

Encryption Approach	Correlation Coefficient P-Value Direction		
	Vertical	***Horizontal***	***Diagonal***
Our algorithm	0.260464	0.260464	1.0
Zhang et al. [5]	0.30631	<0.00001	0.113172

V. Discussion and Conclusions

The algorithm was successfully implemented in the Java programming language and was able to encrypt image files. The execution time was linear in its relation to file size, and memory usage was relatively constant. Using a more memory-efficient language such as C for implementation could improve performance.

The key space was sufficient for withstanding brute-force attacks. However, the key size is relatively large, and as such it could be possible to reduce this key size while maintaining security. Future work could explore the use of smaller keys with fewer PRNG seed sequences and how this parameter affects security.

Mathematical results indicated that the algorithm is secure against differential and statistical attacks. However, the algorithm may be vulnerable to "noise" attacks, where noisy data is generated by a malicious agent and added to an encrypted image. Future work could test the algorithm against noise attacks. The correlation coefficients in the vertical, horizontal and diagonal directions were superior to that of Zhang et al. [5] for the Lena image. This demonstrates the cryptographic security improvements of the algorithm. The algorithm was lossless in decryption, as demonstrated by BCR values for test images.

Some limitations of this study include the original goal of implementing simultaneous compression, the lack of exploration of different applicable PRNGs, and several molecular biology aspects which could have been implemented and explored.

The final result of this study was a medical image encryption algorithm which is able to withstand brute force, differential and statistical attacks. It is superior to existing medical encryption algorithms in some ways, while meeting the standard of most literature in other ways. Future work may improve this algorithm further, preventing malicious entities from accessing confidential patient information.

References

[1] A. Agrawal and A. Kalyanpur, "Synchronizing computer clocks: The challenge of multiple time zones in teleradiology," Indian Journal of Radiology and Imaging, vol. 22, no. 3, p. 240, 2012.

[2] B. Desjardins, Y. Mirsky, M. P. Ortiz, Z. Glozman, L. Tarbox, R. Horn, and S. C. Horii, "DICOM Images Have Been Hacked! Now What?," American Journal of Roentgenology, vol. 214, no. 4, pp. 727–735, 2020.

[3] X. Wu, J. Kurths, and H. Kan, "A robust and lossless DNA encryption scheme for color images," Multimedia Tools and Applications, vol. 77, no. 10, pp. 12349–12376, 2017.

[4] C. W. Pratt and K. Cornely, Essential biochemistry. Hoboken, NJ: Wiley, 2018.

[5] S. Zhang, L. Yang, Y. Zhang, and T. Gao, "A Bit Level Encryption Scheme Based on Hyper-chaotic System Combing with the Ideology of Central Dogma," Chinese Journal of Electronics, vol. 27, no. 3, pp. 595–602, 2018.

[6] K. Ning, "A pseudo DNA cryptography method," arXiv:0903.2693. 2009.

[7] U. Hussain, T. Chithralekha, G. A. Raj, A. A.dharani, and G. G.sathish, "A Hybrid DNA Algorithm for DES using Central Dogma of Molecular Biology (CDMB)," International Journal of Computer Applications, vol. 42, no. 20, pp. 1–4, 2012.

[8] B. Mondal and T. Mandal, "A light weight secure image encryption scheme based on chaos & DNA computing," Journal of King Saud University - Computer and Information Sciences, vol. 29, no. 4, pp. 499–504, 2017.

[9] S. Das, S. N. Mondal, and M. Sanyal, "A Novel Approach of Image Encryption Using Chaos and Dynamic DNA Sequence," 2019 Amity International Conference on Artificial Intelligence (AICAI), 2019.

[10] A. Biryukov, "Substitution–Permutation (SP) Network," Encyclopedia of Cryptography and Security, pp. 602–602, 2011.

[11] L. E. Bassham, A. L. Rukhin, J. R. Soto, J. E. Nechvatal, M. B. Smid, E. D. Barker, S. L. Leigh, M. A. Levenson, M. F. Vangel, D. undefined Banks, N. undefined Heckert, J. undefined Dray, and S. undefined Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST, 2010.

[12] E. M. S. Hossain, K. M. R. Alam, M. R. Biswas, and Y. Morimoto, "A DNA cryptographic technique based on dynamic DNA sequence table," 2016 19th International Conference on Computer and Information Technology (ICCIT), 2016.

[13] M. Kaur and V. Kumar, "A Comprehensive Review on Image Encryption Techniques," Archives of Computational Methods in Engineering, vol. 27, no. 1, pp. 15–43, 2018.

[14] P. T. Akkasaligar and S. Biradar, "Selective medical image encryption using DNA cryptography," Information Security Journal: A Global Perspective, vol. 29, no. 2, pp. 91–101, 2020.

[15] A. Belazi, M. Talha, S. Kharbech, and W. Xiang, "Novel Medical Image Encryption Scheme Based on Chaos and DNA Encoding," IEEE Access, vol. 7, pp. 36667–36681, 2019.

[16] M. Larobina and L. Murino, "Medical Image File Formats," Journal of Digital Imaging, vol. 27, no. 2, pp. 200–206, 2013.

[17] SecureRandom (Java Platform SE 8), 09-Jul-2020. [Online]. Available: https://docs.oracle.com/javase/8/docs/api/java/security/SecureRandom.html. [Accessed: 29-Jul-2020].

[18] A. Chauhan, "Generate a random permutation of 1 to N", GeeksforGeeks. [Online]. Available: https://www.geeksforgeeks.org/generate-a-random-permutation-of-1-to-n/. [Accessed: 29-Jul-2020]

[19] L. Kinstler, "Finding Lena, the Patron Saint of JPEGs," Wired. [Online]. Available: https://www.wired.com/story/finding-lena-the-patron-saint-of-jpegs/. [Accessed: 29-Jul-2020].

[20] Softneta, "DICOM Library - Anonymize, Share, View DICOM files ONLINE," DICOMLibrary. [Online]. Available: https://www.dicomlibrary.com/. [Accessed: 29-Jul-2020].

[21] Sample DICOM files. [Online]. Available: http://www.rubomedical.com/dicom_files/index.html. [Accessed: 29-Jul-2020].

[22] D. Vaughan, "DICOM Sample Images," Dean Vaughan, 11-Jul-2013. [Online]. Available: https://deanvaughan.org/wordpress/2013/07/dicom-sample-images/. [Accessed: 29-July-2020]

[23] "ImageJ," National Institutes of Health. [Online]. Available: https://imagej.nih.gov/ij/. [Accessed: 29-Jul-2020].