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

Arjav Gupta
Faculty of Computer Science
Dalhousie University
Halifax, Canada
ar237808@dal.ca

Srinivas Sampalli
Faculty of Computer Science
Dalhousie University
Halifax, Canada
srini@cs.dal.ca

***Abstract*—With the continued integration of technology in medicine, patient data is often vulnerable to cyber-attacks. Medical data must be secured, however traditional cryptography algorithms are inapplicable to medical images. 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 complexity of O(n) and is resistant to brute force, differential and statistical attacks. The algorithm meets the standards of literature in DNA-based image encryption and surpasses recent approaches in medical image encryption in its defence against cyber-attacks.**

***Index Terms*—image, encryption, medical, DNA, central, dogma, molecular, biology, DICOM, cryptography**

I. Introduction

With the continued progression of medical technology, patient data is more frequently stored and transferred digitally. This makes patient information vulnerable to cyber-attacks, which can often lead to severe consequences for patient well-being. Medical data must be encrypted to prevent malicious use. For medical images specifically, traditional cryptography algorithms are inappropriate due to their inherent features such as bulk data capacity, strong correlation among adjacent pixels, and high redundancy [1]. To address the need for new medical image encryption algorithms, we propose a novel approach based on the central dogma of molecular biology.

The central dogma of molecular biology 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 [2]. 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. Cryptography researchers have been utilizing the inherent features of DNA to create new encryption algorithms which mimic the processes of the natural world [1], [3], [4], [5], [6].

Our algorithm also uses pseudorandom number generators (PRNGs) for several cryptographic steps. PRNGs use mathematical formulae to produce seemingly random sequences of numbers, and re-generate the same sequence using an identical “seed” value [7]. Substitution and permutation operations are also used [8].

II. Literature Survey

Previous literature in this field varies in its biological inspiration. Ning 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 [4]. Hossain et al. [9] 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.

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. The method proposed by Zhang et al. [3] was inspired by the translation step of the central dogma and demonstrated improved correlation coefficients in their cipher images. Kaur and Kumar summarized many DNA-based image encryption methods in their review paper. The review demonstrated a need for faster DNA-based encryption algorithms [10], with the fastest algorithm in the review being proposed by Mondal and Mandal [6]. Recent DNA-based medical image encryption approaches include Akkasaligar and Biradar [11] and Belazi et al. [12]. These methods do not use the translation step of the central dogma or a degenerate genetic code. Larobina and Murino [13] concluded that the Digital Imaging and Communication in Medicine (DICOM) format is the most popular file format in medical imaging.

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

III. Methods

The idea behind this algorithm is to simulate the central dogma of molecular biology by converting data in a “nu-

cleotide" form into a "protein" form, thereby encrypting it. The first step is the conversion of digital information into a nucleotide sequence. Since actual DNA has four nucleotides, this system was adopted here as well. 2-bit values can be used to represent nucleotides (A, G, C or T). This conversion can be seen in most DNA-based cryptography literature as well [4], [5], [6].

Our algorithm is novel in its approach to several steps of central dogma-based encryption. In the binary-to-nucleotide conversion step, others 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 we use all 24 using a PRNG to determine which conversion to use for each 2-bit value. PRNG seed values are used as keys in order to reduce key size.

The next step is permutation of the nucleotide sequence. Permutation is critical as it reduces statistical patterns in the ciphertext.

Conversion of the nucleotide sequence into protein is another novel component of our algorithm. We use a dynamic conversion table, similar to Hossain et al. [9]. There are 64 possible 3-letter codons, therefore our table dynamically permutes all possibilities for each plaintext. For simplicity, 16 amino acid values are used instead of the 20 biological amino acids. 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 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. Decryption is a similar process, using the same seed values for PRNGs in reverse.

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. Seed sequences were generated and used as keys. The permutation steps in the algorithm used the Fisher-Yates shuffle algorithm 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 [14]. Five DICOM format images were found online from three different sources [15][16][17]. These files ranged in their content and file size. DICOM images were converted into 8-bit grayscale pixel data for testing, as this removes all non-image metadata. This metadata can still be encrypted by the algorithm, but using grayscale representations makes testing and comparisons more intuitive.

B. Performance

The encryption and decryption processes are both O(n) for the proposed algorithm. Execution time vs. file size was tested for the Java implementation of the algorithm, for both encryption and decryption of the five DICOM images. Linear time complexity was demonstrated with increasing file sizes. The algorithm has a similar speed to Mondal and Mandal [6], 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 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 language such as C could be used to lower this memory overhead.

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 2320 when 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 measures the sensitivity of a decryption process to a slight changes in key value [10]. The Lena image and three of the DICOM images were encrypted using an original key, then decrypted using the altered key (incremented by 1). Pearson correlation values between the original images and key-sensitive decryptions were very low, ranging from 0.04107 to 0.08792 for the various images. This demonstrates a high key sensitivity for our algorithm.

D. Differential and Statistical 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. The ideal value for NPCR is 100%. The NPCR values for Lena and three of the DICOM test images ranged between 99.607 and 99.620% for our algorithm.

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. Higher UACI values are ideal. Our UACI values ranged between 33.428 and 33.588% for the Lena and three DICOM test images.

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

Histograms of pixel intensity values show the distribution of pixel intensities present in an image. Non-encrypted images generally show patterns in their intensities, while encrypted

images should have uniform distributions. A numerical representation of this is information entropy (IE). 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 measures this probabilistic aspect, and has an ideal value of 8 for an 8-bit image [10]. Our IE values ranged between 7.996 and 7.999. Kaur and Kumar's review paper reported IE values between 7.3419 and 7.9996 [10]. This study out-performs one of the most recent medical image encryption papers by Akkasaligar et al., as their IE values had a mean of 7.846 [11]. Our algorithm produces IE values similar to some literature values and exceeding many others.

A correlation coefficient measures similarity between adjacent pixels. An original image would have strong correlations in the horizontal, vertical and diagonal directions. Encrypted images should have values close to 0. Zhang et al. achieved superior correlation coefficient values compared to most other encryption schemes [4]. Our algorithm produced improved correlation coefficient values (especially along the horizontal and diagonal directions) compared to Zhang et al. Their algorithm produced values of -0.0023, 0.0105 and 0.0031 (vertical, horizontal, diagonal) and ours produced 0.00222, -0.00212 and -0.0007. Our correlation coefficients surpassed the literature for image encryption, indicating strong statistical attack defence.

E. Bit Correct Ratio (BCR)

BCR estimates the difference between an original image and a decrypted image. 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.

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. [3] 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 for 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] 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.

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

[3] 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.

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

[5] 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.

[6] 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.

[7] 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.

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

[9] 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.

[10] 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.

[11] 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.

[12] 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.

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

[14] 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].

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

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

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