Data and Metadata. 2025; 4:690
doi: 10.56294/dm2025690

ORIGINAL

Image encryption based on simple shift, permutation and transformation operations on bit layers

Cifrado de imágenes basado en operaciones sencillas de desplazamiento, permutación y transformación de capas de bits

Saleem Alzoubi[1]

[1]Jadara University, Department of Robotics and artificial intelligence. Irbid, Jordan.

Cite as: Alzoubi S. Image encryption based on simple shift, permutation and transformation operations on bit layers. Data and Metadata. 2025; 4:690. https://doi.org/10.56294/dm2025690

Submitted: 22-02-2024 **Revised:** 27-07-2024 **Accepted:** 11-02-2025 **Published:** 12-02-2025

Editor: Dr. Adrián Alejandro Vitón Castillo

ABSTRACT

Introduction: this paper explores image encryption techniques that leverage transformation, shifting, and permutation operations. The primary focus is on enhancing the security and quality of encrypted raster images by manipulating the individual bit layers of color images.
Method: to encrypt a raster image, the color image is decomposed into binary layers, each representing pixel bits at varying levels of significance. The least significant bits are placed in the least significant layers, while the most significant bits are positioned in the most significant layers. Transformation operations are performed on the bits or their arrays, reconfiguring them into different bit arrangements. Shifting operations are applied to bits across rows and columns within each bit layer, with shifts between layers carried out separately. Additionally, permutation operations are used to further rearrange bit arrays both within individual layers and between layers themselves.
Results: through experimentation, two encryption scenarios have been identified that provide high-quality results for images with different structures. These scenarios produce distinct encrypted images based on the combinations of operations and their sequence, yet maintain a high standard of encryption quality.
Conclusions: the proposed method demonstrates an effective approach to encrypting raster images without relying on external encryption tools, minimizing the risk of information loss during decryption. The combination of transformation, shifting, and permutation operations ensures robust encryption, making the technique suitable for a wide range of image types.

Keywords: Image; Image Encryption; Transformation Operations; Bit Layer; Shift; Bit Permutations.

RESUMEN

Introducción: este artículo explora las técnicas de cifrado de imágenes que aprovechan las operaciones de transformación, desplazamiento y permutación. El objetivo principal es mejorar la seguridad y la calidad de las imágenes raster cifradas manipulando las capas de bits individuales de las imágenes en color.
Método: para cifrar una imagen rasterizada, la imagen en color se descompone en capas binarias, cada una de las cuales representa bits de píxel con distintos niveles de significación. Los bits menos significativos se colocan en las capas menos significativas, mientras que los bits más significativos se colocan en las capas más significativas. Las operaciones de transformación se realizan sobre los bits o sus matrices, reconfigurándolos en diferentes disposiciones de bits. Las operaciones de desplazamiento se aplican a los bits a través de filas y columnas dentro de cada capa de bits, y los desplazamientos entre capas se realizan por separado. Además, se utilizan operaciones de permutación para reorganizar aún más las matrices de bits, tanto dentro de cada capa como entre las propias capas.
Resultados: mediante la experimentación, se han identificado dos escenarios de cifrado que proporcionan resultados de alta calidad para imágenes con estructuras diferentes. Estos escenarios producen imágenes

cifradas distintas en función de las combinaciones de operaciones y su secuencia, pero mantienen un alto nivel de calidad de cifrado.
Conclusiones: el método propuesto demuestra un enfoque eficaz para cifrar imágenes ráster sin depender de herramientas de cifrado externas, minimizando el riesgo de pérdida de información durante el descifrado. La combinación de operaciones de transformación, desplazamiento y permutación garantiza un cifrado robusto, lo que hace que la técnica sea adecuada para una amplia gama de tipos de imágenes.

Palabras clave: Imagen; Cifrado de Imágenes; Operaciones de Transformación; Capa de Bits; Desplazamiento; Permutaciones de Bits.

INTRODUCTION

The information security that is transmitted and stored on various media is becoming increasingly important in the modern world. In modern digital systems, almost all information is presented in digital form, as separate files of various formats. There are a large number of files that are received and processed by various software applications. The most commonly used file formats are text and graphic. The information displayed by such files is the most understandable for humans. People store and transmit public and confidential information in text and graphic files. In this regard, a large number of methods and means of protecting such information have been developed (Bertaccini, 2024; Cimato & Yang, 2017; Uhl & Pommer, 2004). One approach to protecting information presented in the form of images is image encryption. At the present time, there are a large number of methods and tools that are widely presented in various sources (Fang, Liu, Wu, & Liu, 2023; Alghamdi & Munir, 2024). Basically, all these methods are based on the use of external additional tools, with the help of which the encryption of graphic files is carried out. Most often, such external means are pseudo-random number generators (PRNG), which form the key range for encryption (Bilan, Bilan, & Motornyuk, 2020; Bilan, 2022). However, this approach contains some hidden risks, which are that the PRNG may malfunction, which may lead to the loss of fairly large volumes of information. The use of additional means is already a disadvantage. Each encryption method requires additional resources, which are not always effective. Therefore, the task of efficient and reliable encryption of images is relevant. At the same time, there are practically no methods of encrypting images that operate only with the information that the image consists of.

Relative works

Among all existing approaches to image encryption, the simplest and most understandable is the method based on the principles of bitwise encryption similar to the construction of a stream cipher (Bilan & Bilan, 2020; Bilan, 2017). Here, an external key generation generator is used for encryption. In this case, the same key scale generator must be present on the receiving side. As generators for forming the key gamma, PRNGs are used, which can have different implementations (Bilan, 2020). However, using PRNG as an additional external tool complicates the encryption process, and bit-by-bit encryption often leads to partial loss of information.

Along with bit-by-bit encryption of images, there are methods based on block encryption algorithms (Bani Younes & Jantan, 2021; Patel & Vaish, 2023) and based on the RSA algorithm (Sahoo, Mohanty, & Sethi, 2021). Such algorithms provide high resistance to attacks, which corresponds to known symmetric and asymmetric encryption algorithms. However, the implementation of these algorithms requires large computational and time costs. In block algorithms, each block is encrypted with a block algorithm, and in the RSA algorithm, an encryption algorithm is also implemented for each allocated number.

Much attention in modern scientific publications is paid to image encryption, which is based on the use of chaotic transformation (Gong, Luo, Wu, & Zhou, 2022; Hua, Zhou, & Huang, 2019). The use of chaos systems and chaotic maps uses additional external means, as well as complex combined encryption algorithms.

There are image encryption methods that use the Fourier transform, Wavelet transform, Cosine transform and other well-known image transforms (Ben Farah, Guesmi, Kachouri, & Samet, 2020; Mehra & Nishchal, 2015). These transformations are not used to encrypt images as a single operation. Typically, they are complementary to other algorithms, which complicates the encryption process. This results in the use of complex computing resources and time costs. In addition, the use of such transformations produces distortions of the original images.

Among the variety of image encryption methods, there are methods based on operations with the DNA sequence (Vaish & Patel, 2022), which are implemented in combination with other transformations and encryption algorithms, which leads to partial distortion of information and the implementation of complex calculations.

The literature (Singh, Singh, Singh, & Agrawal, 2025; Panwar, Kukreja, Singh, & Singh, 2023) describes methods for image encryption based on deep learning. Such methods are promising, but require significant

computing resources for training and encryption. Due to the use of additional tools and methods, such methods are quite complex, which leads to the loss of information during recovery. Although many studies have shown high resistance to attacks, elements of additional randomization are still used.

Specialists pay attention to the development of image encryption methods based on the division of a raster image into binary layers (Bilan, 2021; Bilan, 2022). The division of a raster image into binary layers is described in detail in (Bilan, 2022; Al-Bdour & Mansour, 2021).

Paper (Bilan & Demash, 2016) describes a method for encrypting an image by encrypting the four most significant binary layers of each of the three bytes using a PRNG. It is shown that it is enough to transform four bit layers of all three RGB bytes to obtain high quality encryption of a raster image. In this case, to implement the method, a PRNG was used, which should generate at the output a key range of high quality and length corresponding to the number of bits equal to the number of bits constituting the twelve bit layers. Formation of a key gamma of large length can lead to failures and partial loss of information during decryption.

Representing an image as binary layers is a promising approach for image encryption. From this perspective, this paper describes a process of encrypting a bitmap image based on the use of simple shift, permutation and transformation operations that are applied to the bit layers of the original image.

Structure of a raster image

The structure of a raster image consists of an array of pixels that represent color and brightness characteristics. These characteristics are encoded using binary code. Most often, the code consists of 24 bits (3 bytes). Each of the three bytes is responsible for red, green and blue colors, as well as their brightness characteristics. If the image is 10×10 in size, it contains 100 pixels, and accordingly 2400 bits are used to form the color and brightness characteristics of the image.

With 24-bit encoding, such an image can be represented as a three-dimensional bit structure with dimensions of 10×10×24 bits. Most often, an image is viewed as a set of numbers represented by binary codes. However, in certain situations, an image can be considered as a multi-layer bit structure (Kirtee et al,2023; Nashat and Ayman, 2022). For the example under consideration, the image consists of 24 layers (figure 1). Each bit layer is a two-dimensional array of bits Pij that encode a certain weight in the binary code of each pixel. For example, the zero layer consists of bits that encode the zero digits in the codes of each pixel, the first layer contains the bits of the first digits of the pixel codes, and so on.

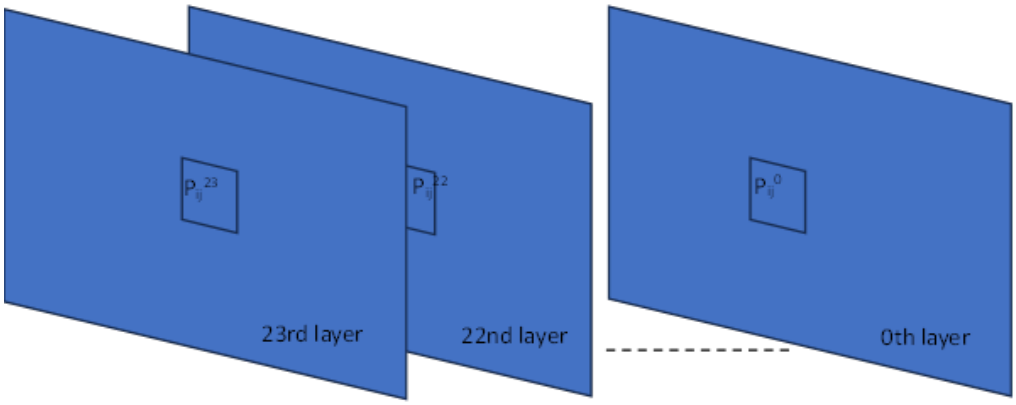

Figure 1. Multilayer bit structure of the image

The first eight bit layers (from 0 to 7 bit layers) are responsible for shades of red, the second eight bit layers (from 8 to 15 bit layers) form green, and the third eight bit layers (from 16 to 23 bit layers) form blue. Changes in bits in each bit layer result in changes in the color and brightness characteristics in the corresponding area of the image, which corresponds to the organization of the RGB model. If the layers are swapped, the colors change significantly and the original image changes its color structure. Breaking an RGB image into layers allows for various kinds of operations that can lead to useful results, such as 3D color shifting (Bilan, 2021), edge pixel extraction (Bilan, 2022), image encryption (Bilan & Demash, 2016), and other image pre-processing operations.

This paper discusses an image encryption method based on various bit layer manipulations.

Image encryption method

Most existing methods of image encryption are based on the use of additional arrays of numbers that are formed pseudo-randomly, as well as preliminary analysis of the image, which can lead to unplanned loss of information. As a rule, encryption is carried out bit by bit or over blocks of bits.

This paper proposes to use the entire information bit array of an image for its own encryption. By implementing a sequence of transformations and permutations of elements of a binary array, image encryption is performed.

https://doi.org/10.56294/dm2025690

As transformations, operations on bits can be implemented, which, using the same operations, can be restored without involving additional bits that do not belong to the image. Such bit conversion operations include the inversion and XOR operations. Several bits of an image can be used to implement the XOR operation. For example, bits of other layers can be used to transform bits of one layer.

The permutations of bits and bit arrays are performed using shift operations within each binary layer, and bits can also be shifted from one bit layer to another, which corresponds to shifts of codes to the right (towards the most significant bits) or to the left (towards the least significant bits) in the codes of the corresponding pixels. The operation of transferring individual bits or bit arrays to different bit layers, as well as within the own bit layer, can also be used. Figure 2 shows a fragment of a raster RGB image measuring 10×10 pixels. From left to right are the original image, the image obtained by using the inversion operation, the image obtained by using shifts, and the image after using bit permutations.

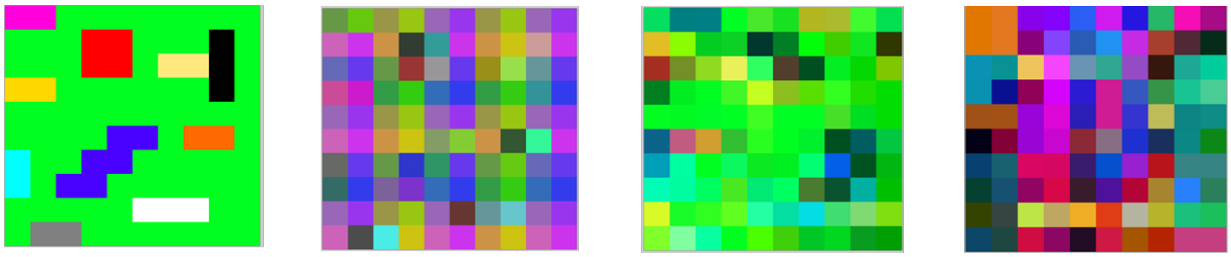

Figure 2. Examples of images obtained as a result of using inversion, shift and permutation operations

As can be seen from figure 2, the resulting images correspond to high encryption quality, but have different color distribution across the entire image field. The shift operation (third image from the left) was performed only in two-dimensional space, since it is impossible to simultaneously shift the code bits of one pixel in three-dimensional space. Therefore, first a shift is performed along the rows and columns in each bit layer, and then a shift is performed from layer to layer or vice versa. For the last image (far right image), bitmap swapping was done between layers. Within a single layer, bitmaps were not swapped.

It is advisable to use combinations of the proposed operations on bit arrays. The results of using various combinations of operations of transformation and transfer of bit arrays are presented in figure 3.

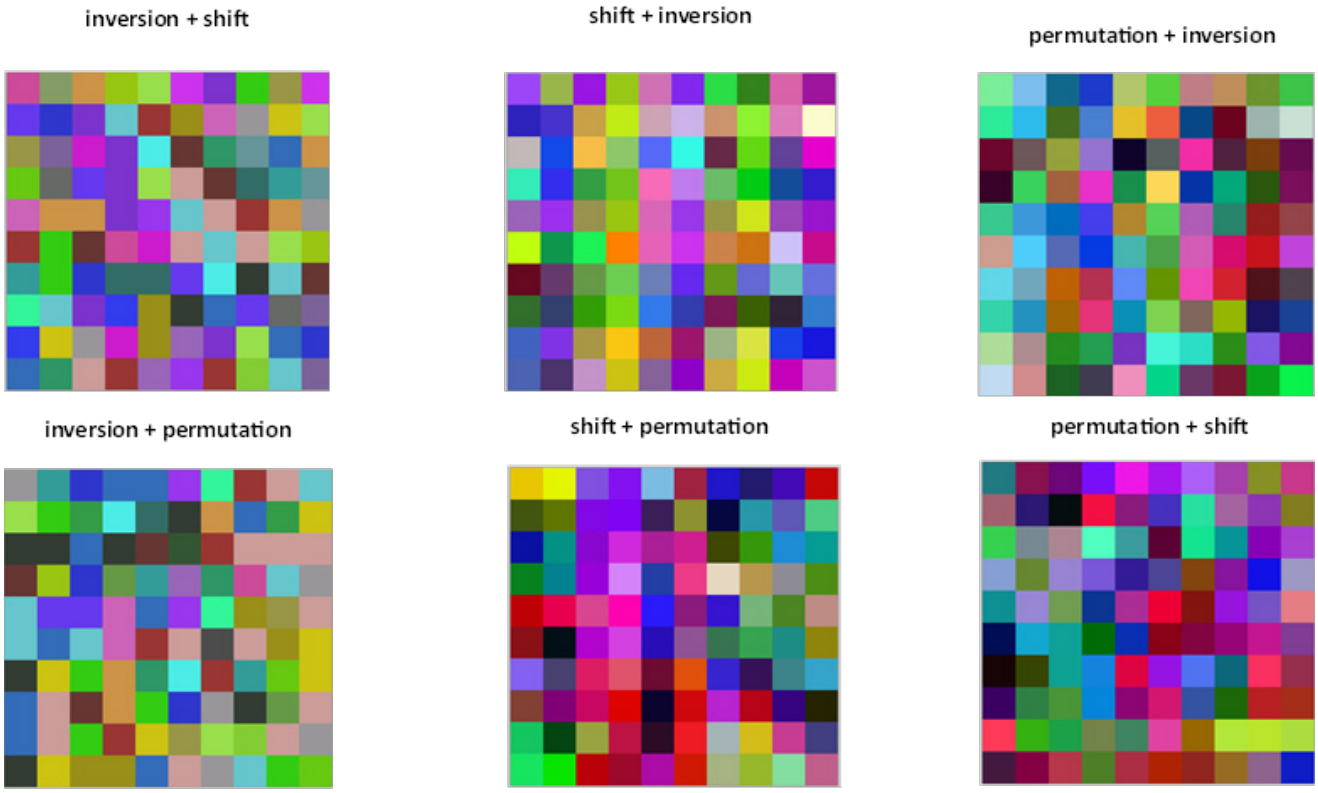

Figure 3. Results of using combinations of transformation and transfer of bit arrays for the example shown in figure 2

Figure 3 shows six combinations of using different operations. All of them give a good result. More combinations of three operations can be used, which also gives a good result.

https://doi.org/10.56294/dm2025690

To encrypt images based on the proposed approach, it is necessary to create a key sequence of operations, which must be strictly ordered, since the last operation in the encryption sequence is the first operation in the decryption sequence. Operations used for encryption must have inverse operations. For example, for the inversion encryption operation, the inverse operation is the inversion operation. For the operation of shifting a bit array to the right by n bits, the inverse operation is the operation of shifting a bit array to the left by n bits. There are a large number of such operations that do not lead to loss of information when using inverse operations.

Since images may have different structures and different color distributions, it is recommended to perform a preliminary image analysis. The result of the preliminary analysis of images is the selection of a scenario for encryption. Each encryption scenario consists of choosing the necessary sequence of operations that give the best effect. Depending on the characteristics of the image, two main scenarios are possible.

The first scenario is to use only bitmap move and permutation operations. Such scenarios apply to images in which most of the colors are distributed throughout the image and there are no dominant colors in the image array. For example, there is no clear background of one color on which objects are displayed in another color and occupy a small area. An example of such an image in figure 4 is shown.

Figure 4. An example of an image to which the first scenario can be applied

In this example, shifts within layers, shifts between layers, and one operation of replacing the right part of the bit array with the left in the third layer were applied. As can be seen from figure 4, it is visually impossible to determine the original elements of the image and their contours. To encrypt such an image, a shift operation was applied. In this case, the encryption scenario can be described by a sequence of numbers that indicate the bit layer number, the row number, and the column number of the image array, the direction of the shift in the row and the number of bits by which the row bits are shifted, as well as the direction of the shift in the column and the number of bits by which the column bits are shifted. For example, the sequence of numbers 2, 3, 0, 1, 20; 2, 0. 5, 0, 30 indicates that in the second bit layer it is necessary to shift the bits of the third row to the right by twenty bits, and then in the second bit layer to shift the contents of the fifth column down by thirty bits. The zero in the second position of each cyclic group indicates that no shifting occurs in the rows. The number in this position indicates the row number. The same applies to the third position, which indicates the column number. The zero in the third position indicates that no shifting of columns is performed at this time iteration. Each shift iteration in the example shown is separated by the symbol “;”.

All shifts are performed cyclically. This means that during a shift, the bits of the last or first outer cells of a row and column are moved to the corresponding cells of the last cells of the row and column.

In this example, the shift is specified in only one plane. In this case, depth-shifting can be used, i.e. bits at one location in each layer can be shifted from one bit layer to another bit layer. In this case, four more numbers n, m, d, s are added to the key sequence (where n is the row number in each bit layer, m is the column number, d is the direction of the shift, s is the number of bits by which the shift is performed). The value d can have the value 1 (shift towards higher bit layers) or 0 (shift towards lower bit layers). The value of $S \leq 24$ since only 24 bit layers are used. The n and m values define the pixel location for all bit layers in which the shift is performed. If no shift is performed across layers, these numbers are equal to 0.

The second scenario is realized for images that have a low density of color distribution throughout the image. In such images, shifting and rearranging often results in minor changes in the visual picture. For example, if the background is predominantly white, then when shifts are made, some units move to the places of others. The reason for this is that the binary code of white contains only ones. The same applies to black. In this case, the zeros are moved.

https://doi.org/10.56294/dm2025690

In this case, it is necessary to use bitmap transformation operations in each bit layer. One of the most effective transformation operations is the inversion operation. This operation can be used for small bit arrays initially. After that, shift and permutation operations are performed. The inversion operation reduces the number of dominant bits and converts them to their inverse value. The shift and permutation operations distribute the bits after inversion over the entire image area. The order in which the operations are applied is of great importance. Different sequences of operations yield different results. Figure 5 shows examples of applying different shift and permutation operations to an image with a saturated white background.

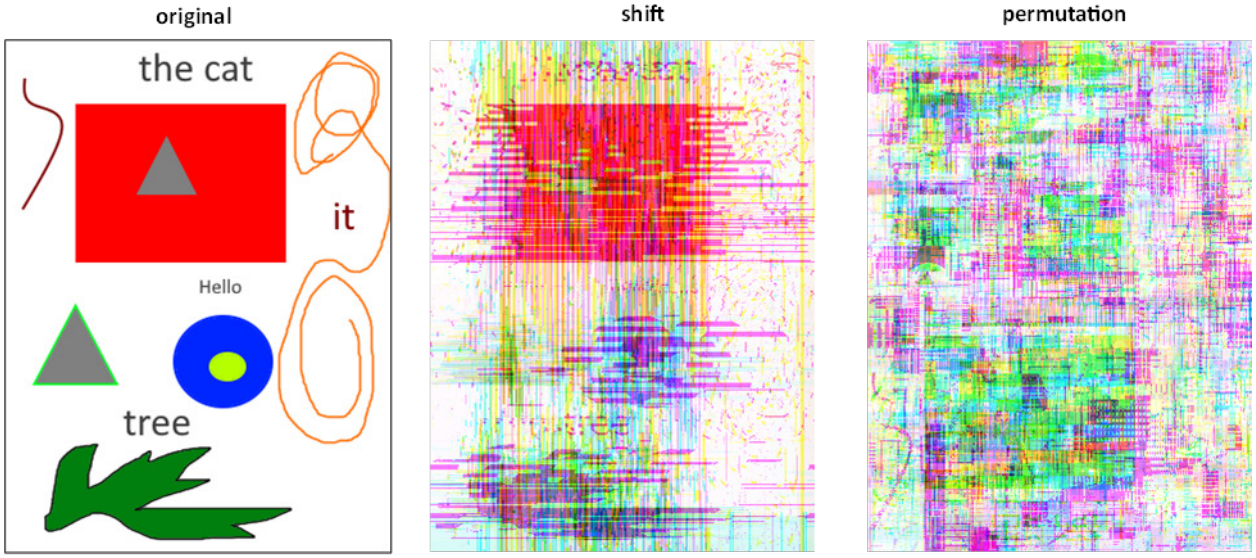

Figure 5. An example of applying shift and permutation operations to the same image

As can be seen from figure 5, the encryption results do not provide high quality, since when using both standard operations, there is a possibility of examining some elements of the original image. Reducing the bit arrays for permutations gives higher quality. The permutation operation can be specified by ten numbers L_1, n_1, m_1, n_2 , m_2, L_2, n_1', m_1', n_2', m_2'. Here the first number (L_1) indicates the number of the bit layer from which the bit array is moved, which is described by the coordinates (n_1, m_1) of the upper left pixel and the lower right pixel (n_2 , m_2). The sixth number (L_2) indicates the number of the bit layer, which is described by the coordinates (n_1', m_1') of the upper left pixel and the lower right pixel (n_2', m_2'). Arrays of both layers must have the same dimensions, but can have different placement in their own layers. In this case, the bit array of the bit layer to which the move is made must move to the place of the first, but can be moved to another place in any bit layer. Arrays must be stored in bit layers because when restoring the original image, pixels with erroneous values appear. Figure 6 shows the results of different sequences of operations for the original image shown in figure 5.

Figure 6. Examples of using combinations of shift, permutation and inversion operations

https://doi.org/10.56294/dm2025690

Figure 6 shows high encryption quality for all combinations of operations. However, the visual results are different for the same initial image. Experiments have shown that reducing the areas of bit arrays that are moved provides greater resistance to attacks, since the number of possible attempts to try all combinations of operations increases. Especially for large-sized images it is almost impossible to enumerate.

To implement encryption, software can be developed in advance that initially splits the raster image into layers and applies various encryption and decryption scenarios to the image. According to the described method, the program can implement a large number of shift, permutation and bit array transformation operations, which leads to large expenditures of time spent on encryption and decryption. To increase the speed of encryption and reduce the volume of encryption and decryption programs, you can use a sequence of functions that generate the necessary numbers based on the initial data. For example, to implement encryption based on shift operations, the following sequence of functions can be used:

$F_L(t)$, $F_R(t)$, $F_C(t)$, $F_d(t)$, $F_N(t)$

Where:

- $F_L(t)$: function that generates the bit layer number at the t-th time step.
- $F_R(t)$, $F_C(t)$: functions that generate the row and column numbers at the t-th time step, respectively.
- $F_d(t)$: a function that specifies the direction of the shift, which can have two possible values 1 (shift to the right or up) or 0 (shift to the left or down).
- $F_N(t)$: generates a number corresponding to the number of bits to be shifted.

The functions $F_R(t)$ and $F_C(t)$ operate in such a way that if one of the functions produces a number greater than 0, then the second function at that moment in time produces 0. If functions $F_R(t)$ and $F_C(t)$, are used that simultaneously generate numbers greater than zero, then priority can be set for the larger number. The function that produces the smaller of the two functions at the corresponding time t takes the value 0. For example, if $F_R(t)=5$, and $F_C(t)=3$, then the corresponding positions of the key code will contain 5, 0. The values of the function $F_d(t)$ indicate the direction of the shift. If $F_R(t)>0$ and $F_C(t)=0$, then when $F_d(t)=1$ a shift to the right is performed, and when $F_d(t)=0$ a shift to the left is performed. If $F_R(t)=0$ and $F_C(t)>0$, then when $F_d(t)=1$ an upward shift is performed, and when $F_d(t)=0$ a downward shift is performed. Each new iteration of shifts begins after the number of shifts corresponding to the number formed by $F_N(t)$ has been implemented.

Since the number of layers is limited, the Amod24 function can be used to implement $F_L(t)$ which forms the remainder of division by 24. The output number will always be less than 24. The output number will always be less than 24. The same function can be applied to $F_R(t)$, $F_C(t)$ and $F_N(t)$. The described functions do not generate numbers that enter as arguments to other functions that implement the transformation of image codes. All bits of the image are preserved and can be distributed differently during encryption.

This method allowed to reduce the time spent on encryption and decryption, since it does not implement bit-by-bit encryption. In addition, the method also increases the encryption speed compared to block encryption algorithms, since keys are used to encrypt each block, and an additional encryption algorithm is implemented.

CONCLUSIONS

This paper presents a method for encrypting images based on various bitmap shift and permutation operations, as well as based on bitmap transformation operations that can have an inverse transformation. This method does not require the use of additional external key arrays involved in the implementation of bit encryption functions or image bit arrays. By dividing the raster image into bit layers, the quality of image encryption has been improved. High quality encryption is achieved by shifting and permuting bit arrays in three planes. The method can work with images of any complexity and size using various scenarios for encryption. The method is characterized by the fact that encryption is carried out without introducing other information. The encrypted image contains only those bits that belong to the original image. Experiments have shown that shifting and permutation are best performed on small-dimensional bit arrays. Also, experiments have shown that different sequences of the same operations lead to different encryption results for homogeneous operations when encrypting not all images, it gives high encryption quality. However, combinations of such operations allow obtaining high quality encryption for any images.

In further research, the author plans to explore options for forming a universal control key sequence for encrypting images of any complexity.

BIBLIOGRAPHIC REFERENCES

1. Massimo Bertaccini. Cryptography Algorithms: Explore New Algorithms in Zero-knowledge, Homomorphic Encryption, and Quantum Cryptography. Packt Publishing (August 12, 2024). 410 p. DOI: https://doi.org/10.22059/IJMS.2021.319211.674442

2. Stelvio Cimato, Ching-Nung Yang. Visual Cryptography and Secret Image Sharing (Digital Imaging and Computer Vision). CRC Press; 1st edition (December 19, 2017). DOI: https://doi.org/10.22059/IJMS.2021.319211.674442

3. Andreas Uhl, Andreas Pommer. Image and Video Encryption: From Digital Rights Management to Secured Personal Communication (Advances in Information Security, 15). Springer; 2005th edition (November 4, 2004). 178 p. DOI: https://doi.org/10.22059/IJMS.2021.319211.674442

4. Pengfei Fang, Han Liu, Chengmao Wu, Min Liu. (2023) A survey of image encryption algorithms based on chaotic system. The Visual Computer Volume 39, pages 1975–2003. DOI: https://doi.org/10.1007/s00371-023-02458-1

5. Yousef Alghamdi and Arslan Munir. Image Encryption Algorithms: A Survey of Design and Evaluation Metrics. J. Cybersecur. Priv. 2024, 4, 126–152. DOI: https://doi.org/10.22059/jcsp.2024.321097.673568

6. S.M.Bilan, M.M.Bilan, R.L. Motornyuk. (2002). New Methods and Paradigms for Modeling Dynamic Processes Based on Cellular Automata, IGI-Global. 2020. — P. 200. DOI: https://doi.org/10.4018/978-1-7998-1679-2

7. Stepan Bilan. Formation Methods, Models, and Hardware Implementation of Pseudorandom Number Generators: Emerging Research and Opportunities.- (2017).- IGI Global, USA.- P. 301. Journal of Theoretical and Applied Information Technology 15th December 2022. Vol.100. No 23. 6998-7004. DOI: https://doi.org/10.3999/jatit.2022.100.23.6998-7004

8. NASHAT AL BDOUR. IMAGE ENCRYPTION METHODOLOGY BASED ON CELLULAR AUTOMATA. DOI: https://doi.org/10.4108/eai.27-7-2020.163249

9. Stepan Bilan, Andrii Demash. High performance encryption tools of visual information based on cellular automata. - Information Technology and Security. - 2016. - Vol. 4, № 1(6). - C. 62-75. DOI: https://doi.org/10.1007/s00371-016-1267-5

10. Yunling Ma, Chengqing Li, Bo Ou. Cryptanalysis of an image block encryption algorithm based on chaotic maps. Journal of Information Security and Applications. Volume 54, October 2020, 102566. DOI: https://doi.org/10.1016/j.jisa.2020.102566

11. Mohammad Ali Bani Younes and Aman Jantan. Image Encryption Using Block-Based Transformation Algorithm. AENG International Journal of Computer Science, 35:1, IJCS_35_1_03. DOI: https://doi.org/10.2139/ssrn.3486783

12. Saumya Patel, Ankita Vaish. Block based visually secure image encryption algorithm using 2D-Compressive Sensing and nonlinearity. Optik. Volume 272, February 2023, 170341. DOI: https://doi.org/10.1016/j.ijleo.2023.170341

13. Kovalchuk A., Lotoshynska N. ENCRYPTION AND DECRYPTION OF GRAYSCALE AND COLOR IMAGES. CSN. 2018, Number 905: pp. 82 - 87. DOI: https://doi.org/10.1016/j.csn.2018.07.010

14. Aradhana Sahoo, Pratyasha Mohanty, and Purna Chandra Sethi. Image Encryption Using RSA Algorithm. In book: Intelligent Systems, Proceedings of ICMIB 2021. 641-652. DOI: https://doi.org/10.1007/978-3-030-76662-2_63

15. Hossein Movafegh Ghadirli, Ali Nodehi, Rasul Enayatifar. An overview of encryption algorithms in color images. Signal Processing. Volume 164, November 2019, Pages 163-185. DOI: https://doi.org/10.1016/j.sigpro.2019.07.026

16. Li-Hua Gong, Hui-Xin Luo, Rou-Qing Wu, Nan-Run Zhou. New 4D chaotic system with hidden attractors and self-excited attractors and its application in image encryption based on RNG. Physica A: Statistical Mechanics and its Applications. Volume 591, 1 April 2022, 126793. DOI: https://doi.org/10.1016/j.physa.2022.126793

17. Zhongyun Hua, Yicong Zhou, Hejiao Huang. Cosine-transform-based chaotic system for image encryption. Information Sciences. Volume 480, April 2019, Pages 403-419. DOI: https://doi.org/10.1016/j.ins.2018.12.043

18. Xingyuan Wang, Shengnan Chen, Yingqian Zhang. A chaotic image encryption algorithm based on random dynamic mixing. Optics & Laser Technology. Volume 138, June 2021, 106837. DOI: https://doi.org/10.1016/j.optlastec.2021.106837

19. Ankita Vaish, Saumya Patel. Securing color images using DNA coding and cosine stockwell transformation in wavelet domain. Optik. Volume 266, September 2022, 169606. DOI: https://doi.org/10.1016/j.ijleo.2022.169606

20. Isha Mehra, Naveen K. Nishchal. Wavelet-based image fusion for securing multiple images through asymmetric keys. Optics Communications. Volume 335, 15 January 2015, Pages 153-160. DOI: https://doi.org/10.1016/j.optcom.2014.09.050

21. M.A. Ben Farah, R. Guesmi, A. Kachouri, M. Samet. A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation. Optics & Laser Technology. Volume 121, January 2020, 105777. DOI: https://doi.org/10.1016/j.optlastec.2019.105777

22. Prakash Singh, Kedar Nath Singh, Amit Kumar Singh, Amrit Kumar Agrawal. Deep learning-based image encryption techniques: Fundamentals, current trends, challenges and future directions. Neurocomputing. Volume 612, 7 January 2025, 128714. DOI: https://doi.org/10.1016/j.neucom.2024.10.017

23. Kirtee Panwar, Sonal Kukreja, Akansha Singh, Krishna Kant Singh. Towards Deep Learning for Efficient Image Encryption. Procedia Computer Science. Volume 218, 2023, Pages 644-650. DOI: https://doi.org/10.1016/j.procs.2023.12.091

24. S.M.Bilan, Evolution of two-dimensional cellular automata. New forms of presentation, Ukrainian Journal of Information Technologies, 2021, Vol. 3, No. 1: 85-90. DOI: https://doi.org/10.4108/eai.27-7-2020.163249

25. Stepan Bilan, Operators for Edge Detection in an Image Based on Technologies of Cellular Automata. International Conference “Information Technology and Interactions” (IT&I-2022). Workshops Proceedings Kyiv, Ukraine, 2022. Vol. 3384 P. 142-150. DOI: https://doi.org/10.1109/IT&I52851.2022.9522423

26. Nashat Al-Bdour, Ayman M Mansour. Optimal Steganographic Method Based on Image Encryption. PRZEGLĄD ELEKTROTECHNICZNY, ISSN 0033-2097, R. 97 NR 6/2021. DOI: https://doi.org/10.22059/jcsp.2021.320425.675688

27. Al-Zoubi, et al. (2023). The influence of soft and hard quality management practises on quality improvement and performance in UAE higher education. International Journal of Data Science 11 3. DOI: https://doi.org/10.22059/ijds.2023.321215.674872

28. Al-Zoubi, et al. (2023). The impacts of task technology fit, transparency, and supply chain agility on the blockchain adoption by SMEs in Jordan. International Journal of Data Science 11 3. DOI: https://doi.org/10.1016/j.ijds.2023.311200

29. Al-Zoubi, et al. (2023). Critical Success Factors for Business Intelligence as well as Bank Performance in Jordan. Uncertain Supply Chain Management 11 3. DOI: https://doi.org/10.22059/usc.2023.321520.674982

30. Saleem Issa Al-Zoubi, Mahdi H. Miraz (2024). Enhancing Robot Navigation Efficiency Using Cellular Automata with Active Cells, Annals of Emerging Technologies in Computing (AETiC) Vol. 8, No. 2, 2024.

31. Ghaleb Abu Rummana , Abdel Rahman Alkhazalib, Shemseddine Ethani Barnatc , Saleem Alzoubid. The contemporary management accounting practices adoption in the public industry: Evidence from Jordan, International Journal of Data and Network Science 8 (2024) 1237–1246.

32. Saleem Issa Al-Zoubi (2021). New Architecture of Storage for Network Security and Identify Data Security Issues in Cloud Computing, Multicultural Education Volume 7, Issue 3, 2021.

FINANCING

No financing.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION

Data curation: Saleem Alzoubi.
Methodology: Saleem Alzoubi.
Software: Saleem Alzoubi.
Drafting - original draft: Saleem Alzoubi.
Writing - proofreading and editing: Saleem Alzoubi.

https://doi.org/10.56294/dm2025690