SECTION 8. INFORMATICS, COMPUTING AND AUTOMATION

DOI: 10.46299/ISG.2025.MONO.TECH.1.8.1

8.1 Accelerated encryption of graphic images

Improving the performance of encryption [190-195] of graphic information is an important aspect of modern cybersecurity. With the ever-increasing volume of graphic data, such as images and videos, requiring storage and transmission over the network, the need to ensure their confidentiality and integrity is becoming increasingly important. Effective encryption helps protect graphic data from unauthorized access and manipulation, which is critical for many industries, such as healthcare, financial services, and government. For example, in the medical industry, images obtained from scanners and other diagnostic tools contain sensitive information about the health of patients that must be protected in accordance with privacy regulations.

Improving encryption performance not only ensures data security, but also optimizes storage and transmission processes. Fast and efficient encryption helps reduce data processing time and reduce the load on network resources, which is especially important for systems with a large amount of graphic content, such as video surveillance and remote medical consultation.

One way to improve encryption efficiency is to use hardware accelerators, such as GPUs, which allow for parallel processing of data. There are also special algorithms, such as AES in CBC or ECB modes, that work effectively with large amounts of images. In addition, data compression can reduce their size before encryption, which also helps speed up the process. Parallel encryption of multiple image segments simultaneously on different processors or machines can also significantly reduce processing time. In addition, cloud resources can be used to process large amounts of data, providing scalability and flexibility in resource management. All of these approaches can be integrated to create an effective encryption system that can protect visual data from unauthorized access.

To improve the efficiency of image encryption, techniques such as selective encryption, which encrypts only sensitive parts of the image, can be used. This reduces processing time and optimizes resource usage. Encryption can also be integrated with image file formats such as JPEG or PNG, which allows the encrypted data to be embedded without significantly increasing the file size. Using a distributed architecture, which allows the encryption process to be distributed across multiple systems on a network, can reduce the load on individual servers and increase scalability for processing large data sets. Real-time encryption, which encrypts images at the time they are created or transmitted, provides a high level of security, especially in live video surveillance systems or in direct communications. These strategies can be used individually or in combination to create a more efficient and secure encryption system, which also helps improve the overall performance of the data processing system.

In addition to the above approaches, it is important to consider the use of modern cryptographic protocols that provide both encryption and authentication of data, for example, protocols based on elliptic curves or RSA. This helps not only to encrypt images, but also to ensure their immutability during transmission or storage. Additionally, to ensure a high level of security, steganography methods can be used, which allow you to embed secret messages in images in such a way that external changes are almost imperceptible. These methods can be especially useful in cases where it is necessary to hide the fact of transmitting encrypted information. Taking into account all these aspects will allow you to create a comprehensive system for protecting graphic data that can withstand modern challenges in the field of cybersecurity and provide reliable protection against unauthorized access and manipulation of important data.

Implementing and updating robust key management protocols and using multi-level authentication systems can significantly improve the security of encrypted graphics data. Completing a holistic security strategy requires consideration of both the technical aspects of encryption and organizational measures aimed at ensuring the integrity, availability, and confidentiality of information over the long term.

The use of graphics processing units (GPUs) for encrypting graphics images is becoming increasingly popular due to the GPU's ability to efficiently process large amounts of data and perform parallel computations.

GPUs consist of a large number of cores that can process data simultaneously. This allows many image pixels to be encrypted at once, greatly speeding up the overall encryption process.

NVIDIA's GeForce gaming lineup includes models like the RTX 3090 with around 10,496 CUDA cores, as well as more affordable models like the RTX 3060 with around 3,584 CUDA cores. NVIDIA also offers professional-grade Quadro and Tesla series GPUs that can have even more cores for specialized tasks. AMD's Radeon series graphics cards also have a large core count, with the Radeon RX 6900 XT having around 5,120 stream processors.

GPUs have high-speed memory with high bandwidth, which is optimal for processing large graphics files without significant latency. Many modern encryption algorithms can be optimized for execution on GPUs. For example, algorithms that use large matrix operations or can be broken down into small independent tasks are ideally suited for GPUs. There are numerous libraries and frameworks that simplify the use of GPUs for encryption, such as CUDA and OpenCL. These technologies allow developers to directly use the hardware capabilities of the GPU to accelerate cryptographic algorithms.

While GPUs are incredibly powerful for processing data, they also have some security limitations, including storing encryption keys in vulnerable locations. Therefore, it is important to properly manage the security of keys and data when using GPUs.

Encrypting images on the GPU can significantly increase processing speed due to parallel processing. Here are some examples of techniques and approaches for encrypting images using the GPU: AES (Advanced Encryption Standard) on the GPU: AES is a widely used symmetric block cipher. It can be adapted to run on the GPU, where each image block is processed by a separate thread. This provides a significant speedup compared to processing on the CPU. Encryption using CUDA or OpenCL:

Using the CUDA or OpenCL frameworks allows you to develop your own encryption algorithms that are optimized for parallel processing. For example, you can develop encryption based on pixel permutations within image blocks, where each pixel is processed by a separate thread.

Homomorphic encryption allows computations to be performed on encrypted data without the need to decrypt it. On GPUs, this can be implemented for parallel processing of large images, allowing complex computations such as changing brightness or contrast on encrypted data. Fractal encryption: Fractal encryption uses iterative algorithms to encode images that can be efficiently implemented on GPUs. Each thread can process a portion of the image, using fractal transformations to change the structure of the image. Pixel encryption: One approach to encrypting images on GPUs is to use encryption of individual pixels or groups of pixels. Different encryption algorithms can be used, such as encryption by generating pseudo-random numbers that determine how to change the color values of pixels. These approaches can be implemented in different software environments, and the choice of a specific technology depends on the needs of the project and the available hardware. Designing image encryption on GPUs requires a deep understanding of both cryptography and parallel computing architecture.

Using texture shaders on the GPU to encrypt image data by transforming texture coordinates or modifying color channels. This can include permuting or changing textures based on cryptographically strong algorithms, ensuring the security and privacy of the processed images.

Dividing the image into blocks and encrypting each block separately using an encryption algorithm, allowing for maximum GPU power. This method is very efficient for processing large images, as each block can be processed independently, reducing processing time.

Using GPUs to generate cryptographic images, which can be useful for image version control systems or for verifying the integrity of images before encryption and after decryption. This helps ensure that the image has not been altered during transmission or storage. Implementing image encryption on GPUs in a cloud

environment, where GPU-accelerated servers can encrypt and decrypt images on the fly, ensuring the security of data being transmitted or stored in the cloud. Encrypting real-time video streams using GPUs to secure video conferencing or streaming services. The GPU can encrypt the video stream before it is transmitted, ensuring privacy and protection against unauthorized access. These techniques leverage the power of GPU parallel computing to provide efficient and secure image and video encryption, which is essential in today's applications where speed and security are critical.

Transform-based algorithms for encrypting graphics images use mathematical transformations to convert the pixel values of an image into another form that is suitable for cryptographic operations. These techniques can effectively reduce the amount of data that needs to be encrypted and improve the image's protection against attacks based on statistical analysis. Here are some of the main transforms used in image encryption.

The Fourier transform translates an image from the spatial domain to the frequency domain, representing the image as a sum of sine and cosine waves of different frequencies. Encoding in the frequency domain can be more efficient because it allows you to focus on the important frequency components of the image while ignoring the less important ones.

Wavelet transforms are often used for data compression, but they can also be used for encryption. Wavelet transforms an image into components of different sizes and levels of detail, allowing only the important components to be encrypted, which reduces the amount of encryption and improves performance. Using wavelet transforms in GPU-accelerated graphics encryption can improve the efficiency and security of data processing. Wavelet transforms an image into layers containing details from the highest to the lowest frequencies. This transform divides the image into sub-images (approximations and details) that can be processed separately. The resulting wavelet coefficients from each layer can be encrypted separately, allowing different levels of security to be applied to different parts of the image. For example, only high-frequency details that contain important information about edges and textures can be

encrypted. The transformation and encryption of wavelet coefficients can be performed in parallel using the GPU, which significantly increases processing speed. GPUs can efficiently perform the parallel mathematical calculations required for wavelet transforms. Wavelet transforms can be combined with other encryption techniques, such as scrambling or cryptographic hash functions, to increase the overall security of the system. This allows for multi-layered security, where each level of wavelet decomposition can have its own encryption scheme.

Wavelet transforms allow for precise image analysis at different levels of detail, which makes it possible to optimize the encryption process. The ability to choose which coefficients to encrypt allows for a balance between security and computing power requirements. GPUs are ideally suited for performing wavelet transforms and encryption due to their ability to perform parallel operations, reducing overall processing time. These characteristics make wavelet transforms an ideal choice for implementation in GPU-accelerated graphics encryption systems, providing high speed and flexibility in protecting digital data.

The cosine transform is popular in JPEG compression, while the DCT transforms blocks of an image from the spatial domain to the frequency domain, similar to Fourier transform but using only cosines. Encrypting the DCT coefficients allows the visual integrity of the image to be preserved after decryption, while leaving less important components unencrypted, which facilitates faster encryption.

Scrambling, or permutation, is an important element in the process of encrypting graphics images, especially when it comes to GPU-accelerated encryption. This technique is used to change the position of pixels or blocks of pixels in an image in order to obfuscate the information before the actual encryption is applied. This makes the image less readable or recognizable before it is fully encrypted. Consider GPU scrambling. The image can be divided into blocks (e.g., 8x8 pixels), and each block can be independently scrambled. This allows the GPU to process each block in parallel, which greatly increases processing speed. The permutation algorithm can be applied to each block or pixel within a block. These can be random permutations or defined by a mathematical function that provides a high degree of distribution and difficulty of

cracking. Application of cryptographic algorithms: After the blocks or pixels are permuted, standard encryption such as AES or other AI algorithms can be applied. The GPU allows these operations to be performed quickly due to its parallel processing capabilities. Scrambling can be optimized on the GPU using various techniques, including the use of texture shaders or computational shaders, which can perform complex mathematical transformations on large data sets at high speed.

GPUs can process large amounts of data in parallel, making them ideal for tasks that require fast processing of large amounts of graphical information. Scalability: Algorithms can easily scale to handle very large images or even video streams. GPUs are suitable not only for encryption, but also for implementing various graphical and computational tasks, making them a valuable resource in many applications. These scrambling methods can effectively provide an additional layer of security over the actual cryptographic encryption.

Homomorphic encryption allows computations to be performed on encrypted data without decrypting it, which is important for privacy protection in areas where data must be kept secret, such as cloud computing. The use of homomorphic encryption for image processing can include image processing for medical or security purposes. The processing process begins by converting the image to a numerical format, which is then encrypted. After this, various computations can be performed, such as image enhancement or edge detection, without decrypting the data. The process ends with decrypting the results to obtain the processed image. However, the application of homomorphic encryption can be difficult due to the high computational cost and the need to develop specialized algorithms, but research in this area is actively ongoing.

Accelerated selective encryption of graphics images is a technology that allows you to encrypt only important parts of an image, significantly reducing processing time and computing resource requirements compared to full encryption. This can be useful for large image datasets or real-time video streams where processing speed is critical. This approach typically involves analyzing images to identify "regions of interest" that may contain sensitive or important information and then encrypting them, while other less important areas of the image can be left unencrypted. This not only saves time and

resources, but can also help to ensure a balance between security and data processing efficiency.

The technique can use various encryption methods, including symmetric and asymmetric ciphers, depending on the security needs and implementation. The application of this approach is in areas where fast processing of large amounts of data is required, such as video surveillance, medical images, or when working with confidential documents containing graphic data.

Selective encryption can also have applications in the mobile technology industry, where resource constraints make it impossible to fully encrypt large amounts of data. In such a context, selective encryption algorithms allow for a balance between security and processing costs by encrypting only those parts of images that contain important information, such as faces, license plates, or other AI identifying markers.

There are several approaches to implementing selective encryption. One is a mask-based method, where a mask is created for each image to define the areas to encrypt. Another approach uses significance analysis to dynamically determine the parts of the image that need to be protected.

Consider accelerated selective encryption techniques

The image is analyzed to identify areas with high information content or importance. Only these areas are encrypted, while other parts remain unencrypted or are encrypted with a lower level of security.

In the case of a face, you can encrypt only the eyes or other important features. This approach reduces the amount of data to encrypt and speeds up the entire process.

Machine learning algorithms can be used to automate ROI selection or to determine which parts of an image contain sensitive information that needs to be encrypted.

Encryption can be adapted depending on network speed or bandwidth requirements. On low-bandwidth networks, you can encrypt smaller parts of the image to save bandwidth.

Data encryption can be accelerated using specialized hardware, such as GPUs or FPGAs, which can quickly process cryptographic algorithms.

Using artificial intelligence (AI) to accelerate image encryption can significantly improve the efficiency and speed of this process. AI, especially machine learning and deep learning, can be used to optimize image encryption and processing algorithms. For example, neural networks can adapt the encryption process to the specific characteristics of images, providing greater encryption efficiency and reducing processing time.

One approach is to use convolutional neural networks (CNNs), which efficiently process images for classification and analysis. These networks can be trained to perform encryption tasks, where they learn to recognize and optimize patterns in image data before encryption. This may involve reducing the size of the data or pre-processing the images to improve the efficiency of encryption algorithms.

AI can also help automate the selection and configuration of encryption settings based on image type and security requirements. This is especially useful in multi-user systems where encryption settings need to be dynamically changed based on specific context or user requirements.

Additionally, using AI to optimize encryption can include developing intelligent systems that detect and adapt to external threats, providing higher levels of security through predictive models and adaptive encryption. This opens up new possibilities for data protection in today's technology ecosystems, where security and privacy demands are rapidly increasing.

AI can analyze large amounts of data to identify potential breaches or weaknesses in encryption, allowing for rapid adjustments to strengthen protection. This is especially important in systems where image processing is performed at high speed, such as in video surveillance systems or telemedicine.

Adaptive AI-based encryption systems can automatically select the level of encryption depending on the context of the image's use. For example, images transmitted over less secure channels can be encrypted with a higher level of protection, while data stored in a more secure environment may require a lower level of encryption. This allows for an effective balance between security needs and processing efficiency.

In addition, the use of AI allows for the integration of more sophisticated encryption algorithms that can be dynamically optimized during processing.

AI can be used to analyze data access patterns and determine the optimal time to encrypt or decrypt images based on anticipated need. This allows the system to stay one step ahead of potential requests, reducing response times and improving overall system performance.

AI can learn the context of image usage and automatically adapt the level of encryption according to the risk level and importance of a given image. For example, images transmitted over insecure channels can be encrypted with a higher level of security.

Integrating data compression with encryption can be automated using AI to achieve maximum efficiency. AI can determine which parts of an image can be compressed more without losing important information, while encrypting them for security.

GANs can be used to create encrypted images that look like regular images when viewed normally, but contain hidden information that is only accessible with the appropriate keys. This opens up new possibilities for steganography and secure data exchange.

AI can analyze existing cryptographic protocols for vulnerabilities and automatically suggest improvements or warn of potential attacks, increasing overall encryption security.

Modification of existing image encryption methods can significantly improve their efficiency and security.

The fusion of symmetric and asymmetric encryption can provide a balance between security and speed. Symmetric encryption can be used to encrypt the image itself due to its speed, while asymmetric encryption can protect the encryption keys used for the symmetric method.

Modify selective encryption so that different areas of the image are encrypted with different levels of intensity depending on their importance and sensitivity. This

can be implemented using machine vision algorithms to automatically identify areas of interest.

Modification of encryption methods using wavelet transform for deeper analysis of images and their subsequent encryption at the level of wavelet coefficients. This allows encrypting only those parts of the image that contain the most important information.

Quantum encryption opens up new possibilities for securing graphics images, thanks to the features of quantum mechanics, such as unpredictability and the impossibility of copying states. This makes quantum encryption suitable for protecting data from attempts at interception or modification.

Accelerated quantum image encryption uses quantum properties to generate encryption keys and protect transmitted data. One key technology is quantum cryptography, specifically the quantum key distribution protocol known as BB84, which allows two parties to securely exchange encryption keys using quantum bits, or qubits, which have the property of entanglement.

The essence of quantum image encryption is to use the unique properties of quantum particles, such as entanglement and superposition, to create secure communication channels. The main method of quantum encryption, quantum key distribution (QKD), provides a way to securely exchange cryptographic keys between two parties without the risk of their interception. In QKD, each bit of the key is encoded in a quantum state (qubit), which can be represented by the polarization of a photon or its direction of rotation.

One of the key properties of quantum particles is that measuring a quantum state necessarily changes that state. This means that any attempt to intercept the qubits during transmission will destroy the information, making the interception attempt detectable. If a distortion is detected during key transmission, it signals possible tampering, and the transmission can be aborted or repeated, providing a high level of security.

Additional techniques, such as qubit entanglement, allow the use of links between particles to further increase security. Entangled qubits react to each other

regardless of distance, so any tampering with one qubit will immediately affect its entangled partner, which can also be detected.

Additionally, quantum algorithms, such as Grover's quantum algorithm, can be used to search large databases much faster than classical algorithms. This has the potential to significantly improve the search and extraction of encryption keys for images, which can be integrated into image processing systems to provide more efficient and secure processing.

Thus, quantum encryption opens up new prospects for ensuring extreme security in the field of processing and transmitting graphic images, and the development of quantum technologies and their integration into modern communication systems can radically change approaches to ensuring confidentiality and data protection.

Chaos theory [193], which studies complex and unpredictable behavior of systems, can be applied to improve the efficiency and security of image encryption. The application of chaotic algorithms to image encryption has several advantages, including high processing speed and complexity, which makes it difficult to break the cipher.

The basic idea is to use chaotic maps, such as the logistic map or Bernoulli map, to generate pseudo-random keys that are used to encrypt image pixels. Chaotic systems are sensitive to initial conditions, so even small changes in the input image or key result in a significant difference in the encrypted output, which increases the security of the encryption.

Encryption occurs in several stages. The first stage involves transforming the image pixels using random values generated using the encryption key. The next stage involves permuting the pixels using another random algorithm, which provides an additional level of security. The result is an image that is significantly different from the original and has a high level of randomness.

This method is effective for real-world applications due to its speed and high resistance to brute-force attacks, making it an attractive choice for modern encryption systems, especially in the field of digital image security.

Among the methods of chaotic encryption of graphic images, several approaches stand out. The first method is to transform pixel values using chaotic maps, such as the logistic map or the Bernoulli map. These maps generate pseudo-random values that are used to modify pixels, thus providing data encryption.

The second method involves chaotic permutation of image pixels, where the order of pixels is changed randomly according to generated chaotic sequences. This increases the unpredictability of the encrypted image and makes it more difficult to analyze.

A third popular method is to combine chaotic transformations with conventional encryption algorithms, such as symmetric or asymmetric encryption algorithms. This approach combines the high speed and efficiency of chaotic encryption with the strength of standard cryptographic methods.

There is a method based on the use of chaotic neural networks, which allows you to adapt encryption parameters depending on the characteristics of the input image. Neural networks can learn optimal encryption methods based on previous experience, thereby increasing the efficiency and security of the process.

There are also several other chaotic image encryption methods worth mentioning. One such method is the use of multidimensional chaotic maps, which are capable of generating more complex chaotic sequences and are applicable to encrypting large, high-resolution images. These maps can effectively scale up encryption without significantly sacrificing processing speed.

Another method is to use inverse chaotic maps for encryption, which involve the reverse decoding process, thus allowing for more reliable and robust ciphers. This is especially important in environments where images need to be protected from high risks of tampering.

Cascading encryption, which uses successive layers of chaotic encryption, each using different keys or algorithms, is also being considered. This approach provides a deeper level of protection and increases the complexity of the cipher, making it more resistant to attacks.

Image encryption protects the confidentiality of visual information by converting it into a form that can only be decrypted with the appropriate key. This method is used in a variety of areas, from personal security to commercial applications, ensuring the security of image transmission over insecure networks. The effectiveness of image encryption depends on the encryption algorithm and the size of the key, which affects the difficulty of decryption without the appropriate key.