Advanced Encryption Standard (AES)

Khadidja REBAHI Yasmine AIT MIMOUNE
Abderrahmane Yaakoub ALAOUCHICHE
Mohamed Walid KESSOUM

Higher National School of Advanced Technologies (ENSTA), Algeria

January 16, 2025

Abstract

This project implements the Advanced Encryption Standard (AES) algorithm to encrypt and decrypt color images using Python. The aim is to secure image data and evaluate the performance and security of AES. The project explains how AES works, including key transformations and modes of operation. Different AES modes are tested, and their effectiveness is compared using tests such as histogram, correlation, and entropy analysis. The project also checks how well AES resists certain attacks, such as differential and known plaintext attacks. This work demonstrates the practical use of AES for securing images.

Keywords: AES, encryption, cryptography, data security, secure communication

1 Introduction

Cryptography is the practice of encoding information to ensure that only intended recipients can access it. This technique has been used for centuries and is still crucial today in protecting data across various domains, including bank cards, computer passwords, and e-commerce. By transforming data into unreadable formats, cryptography protects privacy, maintains data integrity, and verifies identities. In multimedia security, it prevents unauthorized access to images, videos, and audio files, guarding against tampering, controlling access, and deterring piracy. Common applications include file encryption, digital watermarking to verify authenticity, and securing media content, making cryptography indispensable in the digital age.

2 Overview of AES Algorithm

The Advanced Encryption Standard (AES) is a widely trusted encryption algorithm used to secure electronic data by converting it into an unreadable format without the proper key. Established by the U.S. National Institute of Standards and Technology (NIST) in 2001, AES has become the global standard for data encryption due to its strength and efficiency. Unlike its predecessors, DES and triple DES, AES provides enhanced security while maintaining high performance.

AES operates as a block cipher, encrypting data in fixed-size blocks of 128 bits. It supports key lengths of 128, 192, or 256 bits, with longer keys offering stronger protection. The encryption process uses a substitution-permutation network, applying multiple rounds of transformations to the input data. The number of rounds depends on the key size:

- 128-bit key – 10 rounds/ 192-bit key – 12 rounds/ 256-bit key – 14 rounds

Each round involves byte substitution, row shifting, column mixing, and key addition to progressively obscure the data. AES is widely used to secure internet communication, encrypt sensitive files, and protect data across various platforms, ensuring robust defense against cyber threats [1].

2.1 Creation of Round Keys

The round keys used during encryption are derived from the initial key through a process called the Key Schedule. This algorithm generates multiple round keys by transforming the original key, producing a unique key for each encryption round. These round keys play a crucial role in ensuring the complexity and security of the encryption process.

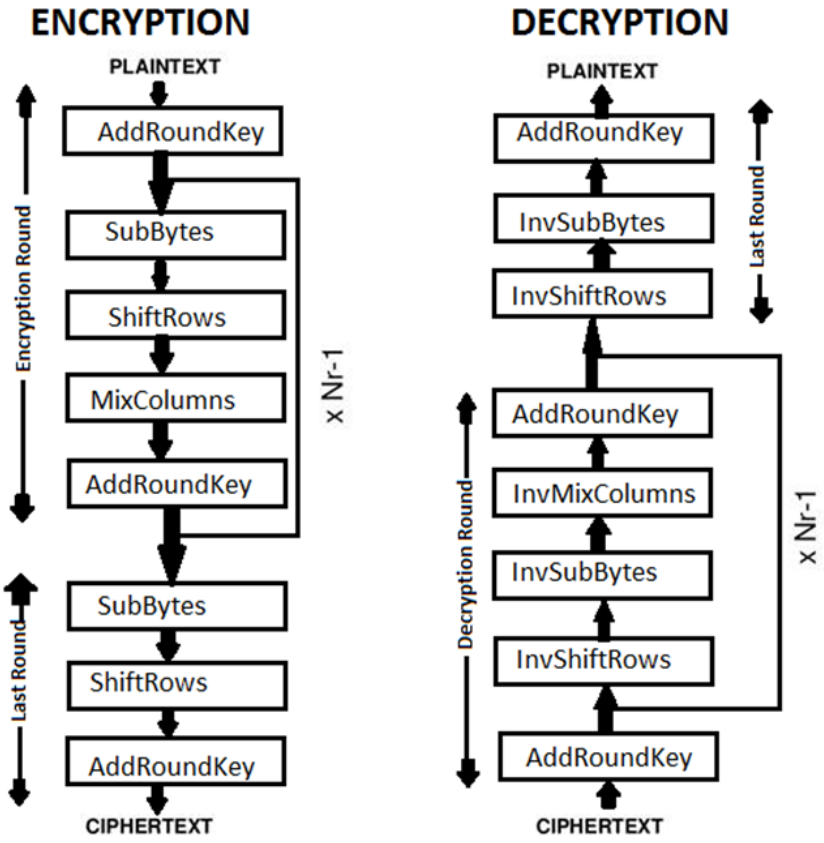


Figure 1: AES Algorithm

2.2 Encryption

The AES encryption process consists of multiple steps applied to data blocks, ensuring that plaintext is transformed into ciphertext securely. The main steps in each round include:

- **SubBytes:** This step implements the substitution, each byte is substituted by another byte. It is performed using a lookup table also called the S-box. This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte. The result of this step is a 16-byte (4 x 4) matrix.
- **ShiftRows:** This step is just as it sounds. Each row is shifted a particular number of times.
  - The first row is not shifted
  - The second row is shifted once to the left.
  - The third row is shifted twice to the left.
  - The fourth row is shifted thrice to the left.
- **MixColumns:** Each column undergoes matrix multiplication with a fixed matrix, altering the byte positions and diffusing the data to increase complexity.
- **Add Round Key:** The output of the previous step is XORed with the corresponding round key. The 16 bytes of data are treated as a 128-bit block during this step.

2.3 Decryption

Decryption in AES follows the reverse process of encryption, undoing each transformation step by step. Depending on the key size, data passes through 10, 12, or 14 decryption rounds. The main stages in each round of decryption are:

- **Add Round Key:** The ciphertext is XORed with the round key.
- **Inverse MixColumns:** Matrix multiplication is applied using the inverse matrix to revert the MixColumns step.
- **ShiftRows (Inverse):** Rows are shifted to the right by varying offsets to reverse the row shifting from encryption.
- **Inverse SubBytes:** Each byte is substituted using the inverse S-box to revert the original byte substitution.By performing these steps, AES decryption successfully restores the original plaintext from ciphertext, ensuring secure and reliable data recovery.

3 AES Modes of Operation

AES can operate in different modes to achieve various cryptographic goals. These modes affect how data blocks are encrypted and how errors propagate during encryption or decryption [2]. The most common AES modes of operation include:

- **Electronic Codebook (ECB):** ECB encrypts each block independently. This simplicity makes it fast, but it is highly insecure for large data because identical plaintext blocks produce identical ciphertext blocks, revealing patterns.

- **Advantage:** Fast and straightforward.
- **Disadvantage:** Vulnerable to pattern analysis and block replay attacks.

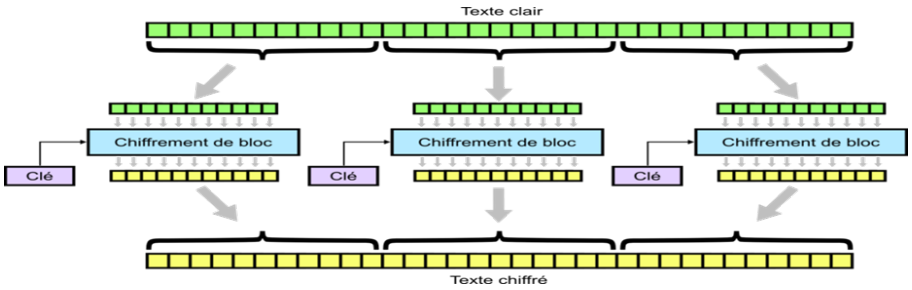


Figure 2: ECB Mode

- **Cipher Block Chaining (CBC):** CBC links each block with the previous one by XORing it with the ciphertext of the preceding block. An Initialization Vector (IV) ensures that even if plaintext blocks are identical, their ciphertexts differ.
  - **Advantage:** Provides better confidentiality by eliminating patterns.
  - **Disadvantage:** Requires IV management, and errors in one block affect subsequent blocks.

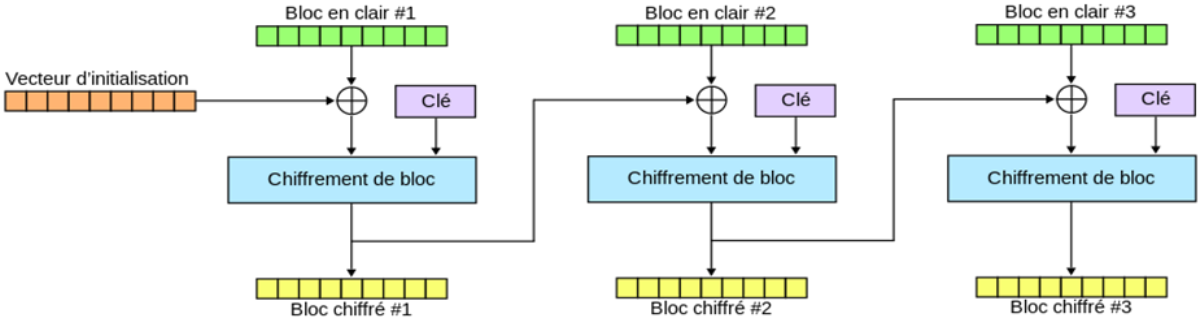


Figure 3: CBC Mode

- **Counter Mode (CTR):** CTR transforms blocks into a stream cipher by encrypting a counter value that is incremented for each block. This allows parallel processing and random access to encrypted data.
  - **Advantage:** Highly efficient and parallelizable.
  - **Disadvantage:** Requires unique counters and careful management to avoid reuse.

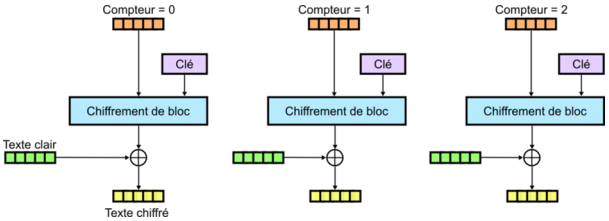


Figure 4: CTR Mode

- **Cipher Feedback (CFB):** CFB turns a block cipher into a self-synchronizing stream cipher. The IV is encrypted, and the output is XORed with the plaintext to produce the ciphertext.
  - **Advantage:** Can process data in smaller chunks (byte-level encryption). And Self-synchronizing if ciphertext gets corrupted.
  - **Disadvantage:** Requires a unique IV and Slower than stream ciphers designed for streaming.

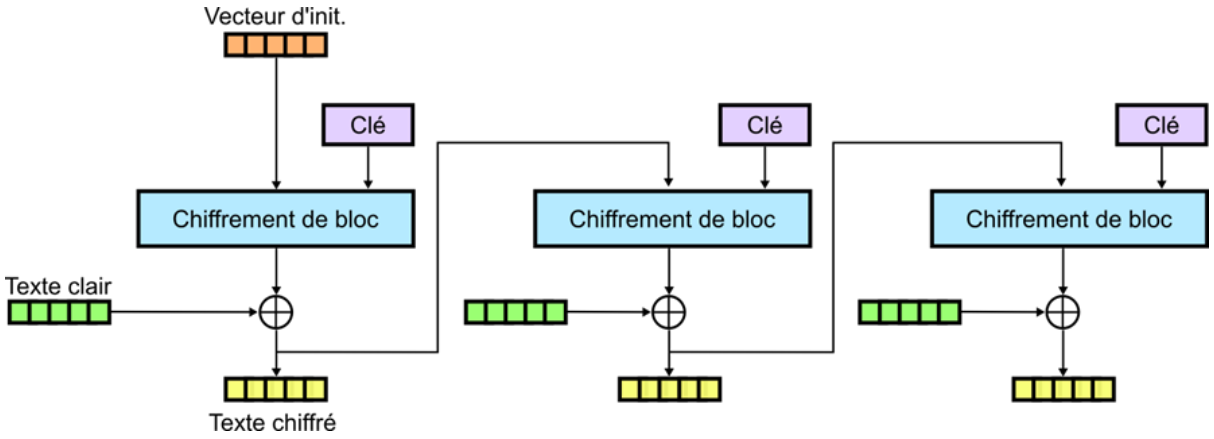


Figure 5: CFB Mode

- **Output Feedback (OFB):** OFB also converts a block cipher into a stream cipher but differs from CFB by pre-generating a keystream independently of the plaintext and ciphertext.
  - **Advantage:** Errors do not propagate (one error only affects one block) and Pre-generation of the keystream allows for parallel encryption.
  - **Disadvantage:** Sensitive to IV reuse and and Slower than dedicated stream ciphers.

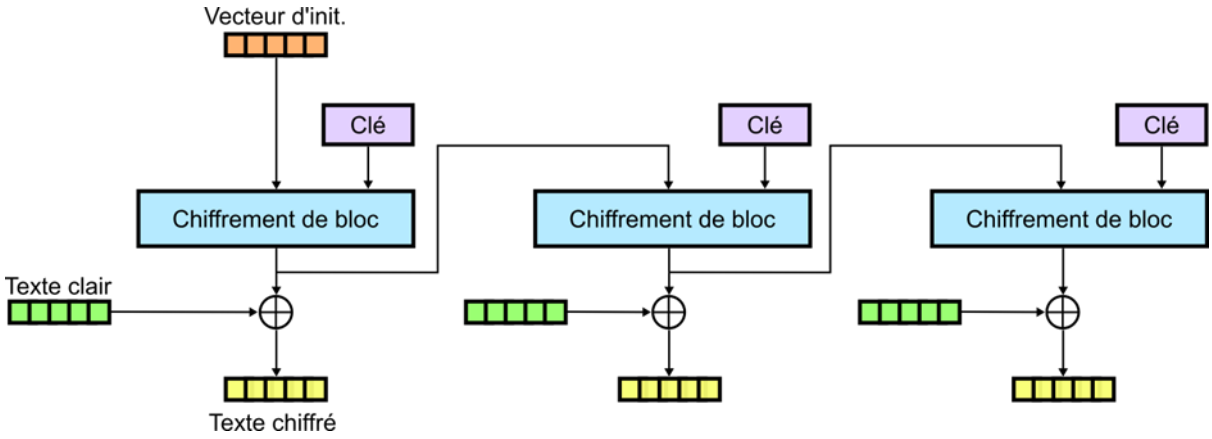


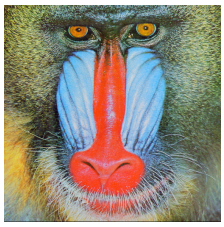
Figure 6: OFB Mode

Each mode serves specific purposes depending on the security needs, performance requirements, and application scenarios, providing flexibility in AES implementations [3].

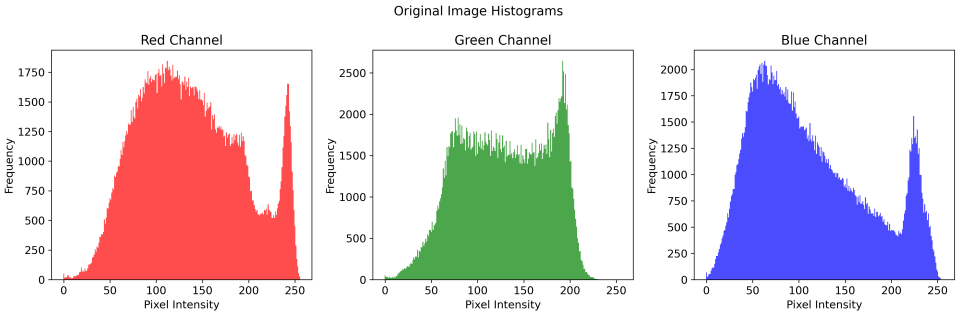
4 Experimental Evaluation and Security Analysis

4.1 Statistical Analyses

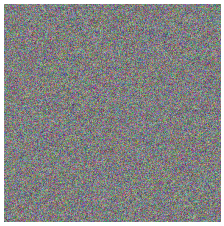
- **Histogram Analysis** A histogram represents the frequency distribution of pixel intensity values across different color channels (red, green, and blue) of an image. An effective encryption algorithm should produce a histogram for the encrypted image that is uniformly distributed, showing no visible patterns making cryptanalysis significantly more challenging.



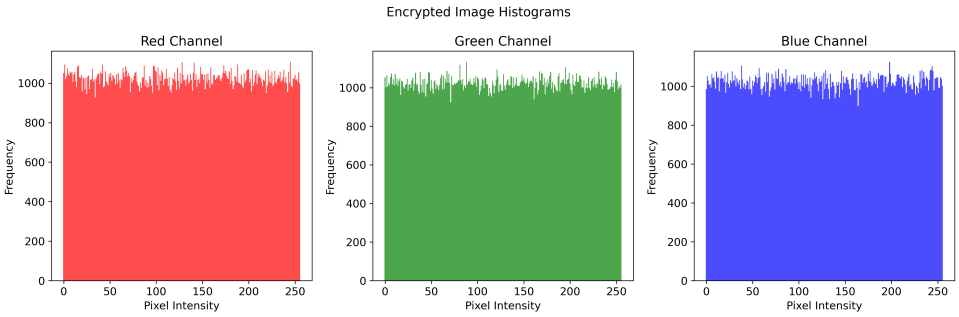
(a) Original Image



(b) RGB Histogram of Original Image



(c) Encrypted Image



(d) RGB Histogram of Encrypted Image

Figure 7: Comparison of RGB Histograms Before and After Encryption

- The histograms of the original image display distinct peaks, indicating the non-uniform distribution of pixel intensities across the red, green, and blue channels. These patterns are characteristic of the original image's content and structure.
- After encryption, the histograms of all three channels (red, green, and blue) become uniformly distributed, with no visible patterns. This uniformity demonstrates that the encryption algorithm successfully randomizes the pixel intensities, effectively concealing the original image's statistical properties.

- **Correlation Analysis**
  - In this section, we perform a correlation analysis between the original image and its encrypted counterpart across the three color channels (R, G, and B). The correlation will be evaluated horizontally, vertically, and diagonally to observe how the encryption process affects the correlation in different directions. This analysis will also highlight the impact of encryption on the correlation values, considering both the individual color channels and the spatial orientations (H, V, and D).

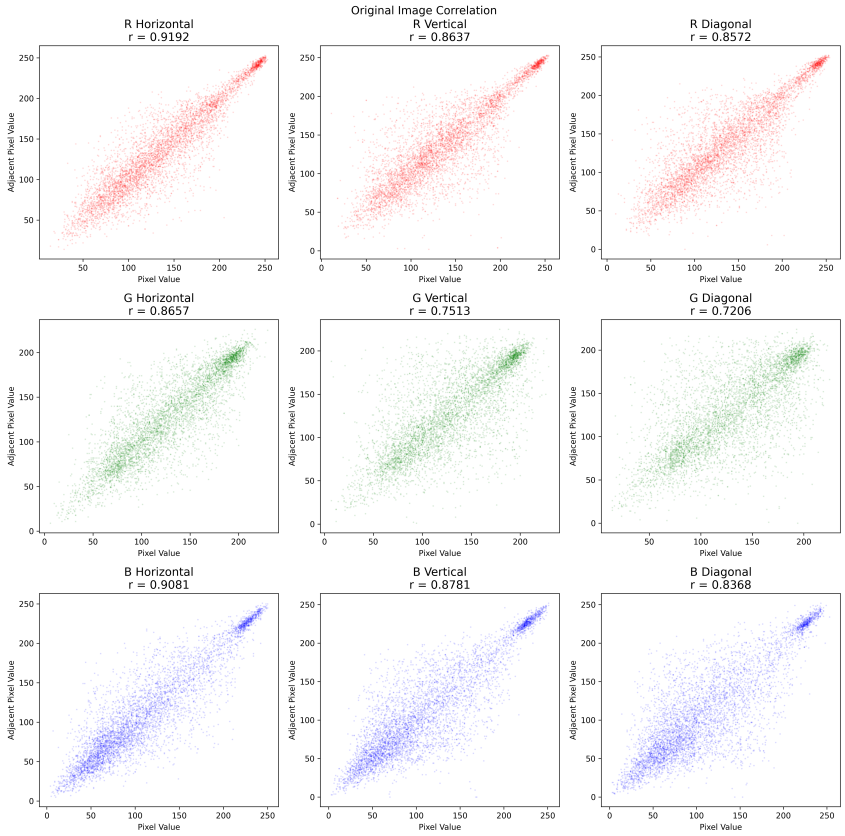


Figure 8: Correlation Analysis of Original Image

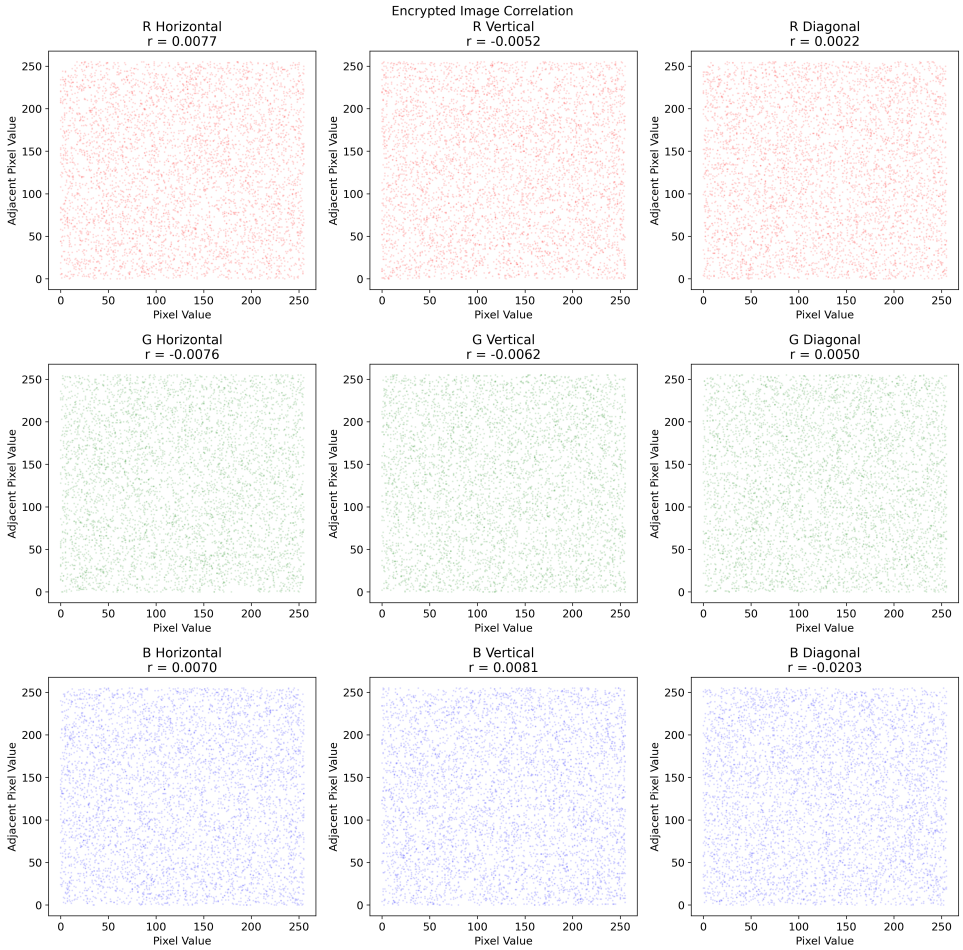


Figure 9: Correlation Analysis of Encrypted Image

- **Observations:**
  - In the correlation analysis of the original image, a clear pattern is observed across all three color channels (R, G, B) and in all directions (H, V, D). This indicates a strong correlation between the pixels and their adjacent counterparts before encryption, forming a positive correlation. However, after encryption, the results change drastically. The correlation values are significantly reduced, and there is no visible pattern. The image essentially transforms into a scattered cloud of points, with no correlation between the pixels, indicating that the encryption process successfully disrupts the pixel relationships.
- **Conclusion:**
  - The encryption process effectively eliminates the correlation between adjacent pixels in the image, as evidenced by the random distribution of points in the correlation analysis post-encryption. This confirms that encryption successfully obscures the original pixel patterns, ensuring data security by preventing any discernible relationships between pixels in the encrypted image.

Table 1: Correlation Coefficients for Original and Encrypted Images Across Different Channels and Directions.

Direction	**Channel**	**Original**	**ECB**	**CBC**	**CFB**	**OFB**	**CTR**
H	R	0.9192	-0.0152	0.0077	0.0233	0.0233	0.0302
	G	0.8657	0.0030	-0.0076	-0.0096	-0.0096	0.0187
	B	0.9081	-0.0050	0.0070	0.0024	0.0024	-0.0128
V	R	0.8637	0.0023	-0.0052	-0.0048	-0.0048	-0.0103
	G	0.7513	0.0014	-0.0062	-0.0092	-0.0092	0.0048
	B	0.8781	0.0357	0.0081	-0.0114	-0.0114	0.0176
D	R	0.8572	-0.0019	0.0022	-0.0222	-0.0222	0.0122
	G	0.7206	-0.0093	0.0050	-0.0061	-0.0061	-0.0125
	B	0.8368	0.0240	-0.0203	0.0024	0.0024	-0.0254

- **Conclusion:**
  - The correlation coefficients across different encryption algorithms (ECB, CBC, CFB, OFB, CTR) reveal that the encryption process significantly disrupts the correlation between adjacent pixels in the image. In the original image, the correlation coefficients for all three color channels (R, G, B) and across all directions (H, V, D) are high, indicating a strong correlation between the pixels. However, after encryption, the correlation coefficients are greatly reduced, with many values near zero or even negative, signaling the loss of the original pixel relationships.
  - Specifically, the ECB algorithm results in the most significant disruption, as the correlation values drop drastically across all color channels and directions. This is followed by the CBC, CFB, OFB, and CTR algorithms, which also reduce the correlation, though not as drastically as ECB. The encryption processes effectively obscure the pixel patterns, ensuring that the encrypted image has no discernible structure or pattern, enhancing the security of the image data.
  - In summary, the correlation analysis confirms that encryption algorithms effectively achieve their goal of obfuscating pixel relationships, with ECB being the most effective in disrupting these correlations.

- **Entropy Analysis**
  - Entropy measures the randomness or unpredictability of an image, with higher values indicating greater randomness and improved security. This section analyzes the entropy values of the original and encrypted images across various encryption algorithms (ECB, CBC, CFB, OFB, CTR) in the R, G, and B color channels to assess the effectiveness of the encryption in enhancing image security.

Table 2: Entropy Values for Original and Encrypted Images.

Image	**RGB**	**R**	**G**	**B**
Original	7.7624	7.7067	7.4744	7.7522
ECB	7.9998	7.9993	7.9993	7.9992
CBC	7.9998	7.9993	7.9993	7.9993
CFB	7.9997	7.9993	7.9992	7.9993
OFB	7.9998	7.9992	7.9993	7.9994
CTR	7.9998	7.9993	7.9994	7.9993

- **Conclusion:**
  - The original image has relatively lower entropy values, ranging from 7.4744 to 7.7624, indicating some predictability. After encryption, all algorithms (ECB, CBC, CFB, OFB, CTR) result in entropy values close to the maximum of 8, demonstrating that the encryption introduces significant randomness and eliminates discernible patterns.
  - The encryption algorithms effectively increase the entropy of the image, making it more random and secure. All encryption methods produce similar near-maximal entropy values, confirming their success in removing predictability and enhancing image security.

4.2 Differential Analysis

- **NPCR:** NPCR measures the percentage of pixels that change in the encrypted image when a small modification is made to the original image. It assesses how sensitive the encryption algorithm is to minor changes in the plaintext image.
- **UACI:** UACI measures the average intensity of the differences between two encrypted images when a small change is made in the original image. It quantifies how much the pixel values change, on average, due to the encryption.
- **PSNR:** PSNR measures the similarity between the original plaintext image and the encrypted image by treating the original image as a "signal" and the encrypted image as "noise." A low PSNR value indicates high distortion, meaning the encrypted image is very different from the original.

Here are the results for the Differential Analysis NPCR , UACI and PSNR :

Table 3: Encryption Performance Metrics for Different Modes.

Encryption Mode	**NPCR (%)**	**UACI (%)**	**PSNR (dB)**
ECB	99.59	33.4966	8.7753
CBC	99.6035	33.4620	8.7683
CFB	99.6029	33.4322	8.7769
OFB	99.6000	33.4449	8.7613
CTR	99.5991	33.4490	8.7816

- **Observations:**
  - **NPCR:**
    - All modes (ECB, CBC, CFB, OFB, CTR) achieved NPCR values close to 99.6%, indicating that nearly all pixels in the encrypted images change when a small modification is made in the plaintext.
    - **ECB Mode:** While ECB performs well in terms of NPCR, it is insecure for image encryption due to its inability to hide patterns, as identical plaintext blocks produce identical ciphertext blocks.
  - **UACI:**
    - UACI values for all modes are close to 33%, with minor variations (ranging from 33.43% to 33.46%). This is consistent with the theoretical ideal value of 33.46% for strong encryption algorithms.
    - **CFB, OFB, and CTR Modes:** These modes have UACI values slightly below 33.46%, indicating slightly less average pixel intensity change, but still fall within an acceptable range for good diffusion and randomness.
  - **PSNR:**
    - PSNR values are around 8.76–8.78 dB, confirming that the encrypted images are highly distorted compared to the original images.
    - **CBC and OFB Modes:** These modes have the lowest PSNR values (8.7683 and 8.7613 dB), indicating slightly better encryption performance.

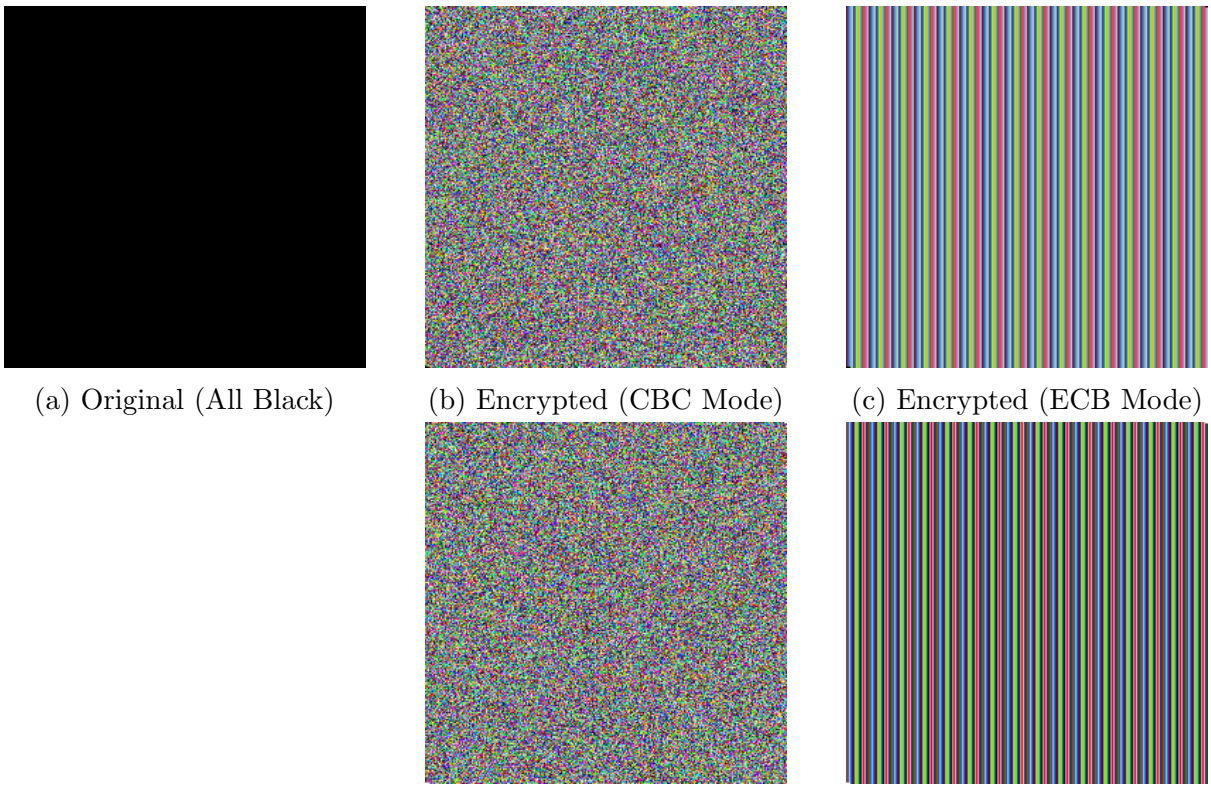
    * **CTR Mode:** With the highest PSNR value (8.7816 dB), CTR mode has slightly less distortion, but the difference is minimal and does not affect encryption strength.

- **Conclusion:**
  - The differential analysis results confirm that all tested AES modes provide strong encryption. Metrics such as NPCR (approximately 99.6%), UACI (around 33%), and PSNR (8.76–8.78 dB) demonstrate high sensitivity to plaintext changes and effective concealment of visual patterns.
  - **ECB Mode:** Despite similar NPCR, UACI, and PSNR values, ECB is unsuitable for real-world image encryption due to its failure to hide patterns, making it vulnerable to statistical attacks.
  - **CBC, CFB, OFB, and CTR Modes:** These modes provide stronger encryption by eliminating patterns in the ciphertext, making them better suited for practical image encryption.
  - **Best Performing Modes:** Among these, CBC and OFB modes demonstrated slightly better overall encryption quality.

4.3 Known Plaintext Attack

In this section, we explore the concept of Known Plaintext Attack (KPA) through a statistical analysis of two encrypted images: one all-white and one all-black. The goal of this analysis is to evaluate how well the encryption algorithms can protect against this type of attack, where the attacker has prior knowledge of the plaintext. By testing with two extreme cases—an all-white image and an all-black image—we can examine how the encryption algorithms handle uniform, predictable data. We conduct the analysis using ECB and CBC modes, to assess their ability to obscure such patterns and enhance security.

– **Histogram Analysis**



(a) Original (All Black) (b) Encrypted (CBC Mode) (c) Encrypted (ECB Mode)

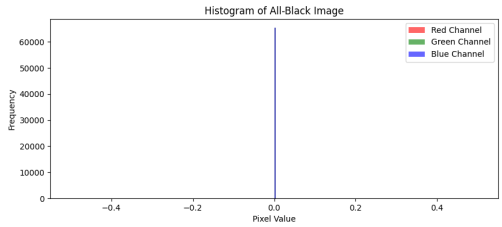
(d) Original (All White) (e) Encrypted (CBC Mode) (f) Encrypted (ECB Mode)

Figure 10: Comparison of Original and Encrypted Images (All Black and All White) in CBC and ECB Modes.

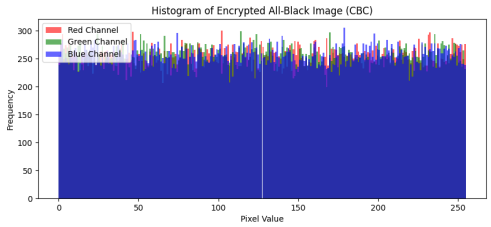
– **Observation :**

* In the CBC mode, both the all-white and all-black encrypted images are well-secured, with the pixel values appearing completely random. There are no discernible patterns in the pixel distribution, indicating that the encryption effectively eliminates any predictability

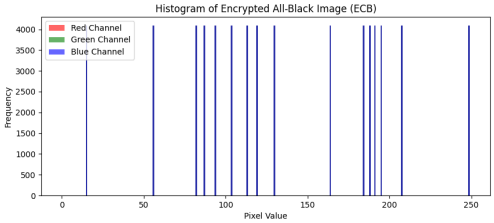
* In contrast, the ECB mode exhibits a significant weakness. Both the all-white and all-black encrypted images display obvious patterns, particularly in the form of alternating columns for each RGB color channel. This pattern is visible across the entire encrypted image, where the columns of pixels alternate between red, green, and blue in a regular pattern. This indicates that ECB does not effectively obscure the structure of the original image, making it more vulnerable to analysis and attack.



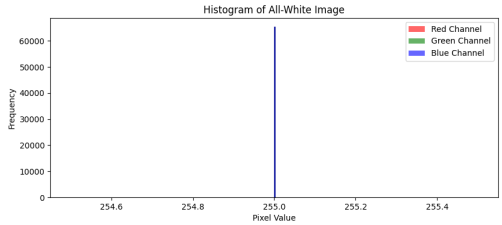
(a) Histogram of Original (All Black)



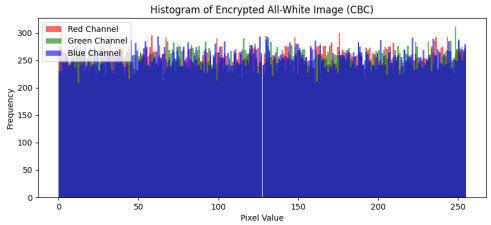
(b) Histogram of Encrypted (CBC Mode)



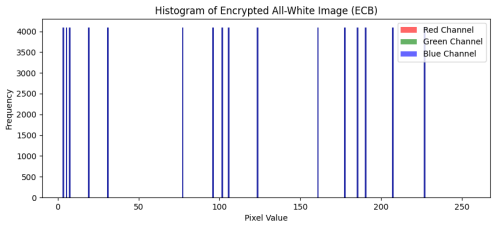
(c) Histogram of Encrypted (ECB Mode)



(d) Histogram of Original (All White)



(e) Histogram of Encrypted (CBC Mode)



(f) Histogram of Encrypted (ECB Mode)

Figure 11: Comparison of RGB Histograms for Original and Encrypted Images in CBC and ECB Modes. Each row displays the histogram analysis for all black (top row) and all white (bottom row) images.

– **Observation :**

  * We plotted histograms of the all-black and all-white images before and after encryption in both ECB and CBC modes. For the original images, the histograms show that the all-white image has pixel values concentrated exclusively at 255, while the all-black image has pixel values concentrated at 0. After encryption, the results vary significantly between ECB and CBC modes. In CBC mode, the histograms display all possible pixel values distributed approximately evenly, resembling white noise, indicating effective randomization. In contrast, the ECB mode histograms reveal a limited range of pixel values with noticeable patterns, failing to fully randomize the pixel distribution. This highlights the weakness of ECB mode in obscuring the original data, as patterns persist in the encrypted images.

– **Correlation Analysis**

Table 4: RGB Correlations for Original, CBC, and ECB Modes

Mode	**Channel**	**Correlations (Horizontal, Vertical, Diagonal)**
Original	R	—
	G	—
	B	—
CBC	R	-0.0212, -0.0398, -0.0462
	G	0.0129, 0.0114, 0.0053
	B	0.0362, 0.0283, 0.0097
ECB	R	-0.0675, 1.0000, -0.0675
	G	-0.0884, 1.0000, -0.0884
	B	-0.0652, 1.0000, -0.0652

– **Observation :**

  * The correlation analysis shows a stark contrast between the CBC and ECB modes. In the original image, the RGB channels exhibit strong correlations (values not provided in the table), reflecting the predictable relationships between adjacent pixels. For CBC mode, the correlation coefficients are near zero across horizontal, vertical, and diagonal directions, indicating that the encryption has effectively eliminated any pixel correlations and randomized the data. On the other hand, the ECB mode displays a significant weakness: while the horizontal and diagonal correlations are disrupted (near zero), the vertical correlation remains exactly 1.0000. This reveals that ECB fails to obscure patterns in the vertical direction, leaving the encrypted image vulnerable to analysis.

– **Entropy Analysis**

Table 5: Image Entropy for Original, CBC, and ECB Modes

Mode	Channel	Entropy
Original	R	-0.0
	G	-0.0
	B	-0.0
CBC	R	7.9972
	G	7.9974
	B	7.9971
ECB	R	4.0008
	G	4.0008
	B	4.0010

– **Observation :**

* The entropy analysis further supports these observations. The original image has an entropy of 0.0 for all channels, as expected for uniform pixel values. In CBC mode, the entropy values approach the ideal value of 8 for all channels, indicating a highly random distribution of pixel intensities. However, the entropy values for ECB mode are significantly lower, around 4 for all channels, confirming that it fails to achieve the same level of randomness and leaves the image more predictable.

– **Conclusion :**

* The CBC mode demonstrates superior encryption performance, effectively randomizing pixel correlations and achieving near-ideal entropy values, making it highly secure. In contrast, the ECB mode exhibits critical flaws, particularly with persistent vertical correlation and lower entropy values, which compromise its ability to secure the image. This analysis underscores the importance of using CBC mode for robust image encryption.

5 Final Conclusion

This project implemented and evaluated the AES algorithm for securing color image data using modes: ECB, CBC, CFB, OFB, and CTR. We assessed their effectiveness with histogram analysis, correlation coefficients, entropy, and differential analysis.

Key Findings

1. **Correlation Analysis:**
   - All modes disrupted pixel relationships, with ECB being the most effective at reducing correlation but unsuitable due to its pattern exposure.
   - CBC, CFB, OFB, and CTR eliminated patterns in ciphertext, providing better security.
2. **Entropy Analysis:**
   - All modes increased randomness, achieving near-maximal entropy (~ 7.9998), confirming effectiveness in obscuring the original image structure.
3. **Differential Analysis:**
   - NPCR ($\sim 99.6\%$) and UACI (close to 33%) showed AES's high sensitivity to plaintext changes, with low PSNR (~ 8.76 dB) confirming robustness.
4. **Known Plaintext Attacks (KPA):**
   - CBC successfully hid patterns in uniform images, while ECB exposed vulnerabilities.

Final Insights

- AES remains a robust, efficient, and versatile choice for image security.
- CBC and OFB provided the best balance of security and performance, while ECB proved insecure due to its pattern exposure.

Broader Implications

This work shows AES's effectiveness in protecting image data against statistical and differential attacks, with potential applications in secure image storage, transmission, and digital forensics.

References

[1] Geeksfor Geeks. Advanced Encryption Standard (AES). https://www.geeksforgeeks.org/advanced-encryption-standard-aes/

[2] Nielson, S. J., & Monson, C. K. Practical Cryptography in Python. Apress, 2021.

[3] Bray, S. W. Implementing Cryptography Using Python. McGraw-Hill Education, 2020.

[4] Yasin.O, Mohammed Yaseen.A. Performance Analyses of AES and 3DES Algorithms for Encryption of Satellite Images. Karabuk University.

Authors and Affiliations

Yasmine Aitmimoune[1], Walid Kessoum[1], Khadija Rebahi[1], Abderrahmane Yaakoub Alaouchiche[1]

[1] *École Nationale Supérieure de Technologie Avancées (ENSTA), Algiers, Algeria*

y_aitmimoune@ensta.edu.dz
m_kessoum@ensta.edu.dz
k_rebahi@ensta.edu.dz
a_alaouchiche@ensta.edu.dz