Seminar Nasional Sains Data 2024 (SENADA 2024)
UPN "Veteran" Jawa Timur
E-ISSN 2808-5841
P-ISSN 2808-7283

# Pengamanan Pesan Teks Pada Citra Digital Menggunakan Kombinasi Algoritma RSA-Quasigroup Cipher dan Metode LSB Pola Zig-zag

Yedija A Lesnussa[1], Berny Pebo Tomasouw[2]
[1, 2]Prodi Matematika, FMIPA Universitas Pattimura
[2]bptomasouw@gamil.com
Corresponding author: [1]lesnussayedija@gmail.com

***Abstract:*** Data security is very important in maintaining the confidentiality of information, especially sensitive information that should only be known by authorized parties. one of the data security techniques that are often used is RSA and LSB. RSA is done to hide the content of information by converting the information into a cipher using a key. Quasigroup Cipher is one of the cryptographic algorithms that is very difficult to crack, so when combined with the RSA algorithm will increase the security level of the message or information. Steganography aims to hide a secret message (hiding message) or secret writing (covered writing) so that the message is not detected by others. one method in steganography is Least Significant Bit (LSB). LSB is a steganography method that works by changing the least significant bit. the combination of RSA-Quasigroup cipher and LSB method is expected to increase message security for the better. This can be seen from the PSNR value from the comparison of the image before and after the message is inserted. based on the results of image quality analysis, it is obtained that the smallest PSNR value is 68.61 and the average PSNR of the entire image is 62.89. these results show that the combination of RSA-Quasigroup Cipher and zig-zag pattern LSB method has a very good level of security
***Keywords:*** Message, Security Least Significant Bit, RSA, Quasigroup Cipher, Zig-zag Pattern.

**Abstrak:** Keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi terutama informasi sensitif yang hanya boleh diketahui oleh pihak yang berhak saja. salah satu teknik pengamanan data yang sering digunakan adalah rsa dan LSB. RSA dilakukan untuk menyembunyikan konten dari suatu informasi dengan mengubah informasi tersebut menjadi sandi dengan menggunakan kunci. quasigroup cipher adalah salah satu algoritma kriptografi yang untuk memecahkannya sangat sulit, sehingga bila dikombinasikan dengan algoritma RSA akan meningkatkan tingkat keamanan pesan atau informasi. Steganografi bertujuan untuk menyembunyikan pesan rahasia (hiding message) atau tulisan rahasia (covered writing) sehingga pesan tersebut tidak terdeteksi oleh orang lain. salah satu metode dalam steganografi adalah Least Significant Bit (LSB). LSB merupakan metode steganografi yang bekerja dengan mengubah bit yang paling tidak signifikan. kombinasi RSA-quasigroup cipher dan metode LSB diharapkan dapat meningkatkan keamanan pesan menjadi lebih baik. Hal ini bisa terlihat dari nilai psn dari hasil perbandingan citra sebelum dan sesudah disisipi pesan. berdasarkan hasil analisis kualitas citra, diperoleh bahwa nilai psnr terkecil adalah 68.61 dan psnr rata-rata seluruh citra yaitu 62,89. hasil ini memperlihatkan bahwa kombinasi RSA-quasigroup cipher dan metode LSB pola zig-zag memiliki tingkat kemanan yang sangat baik.
**Kata kunci:** Pesan, Pengamanan, LSB, RSA-Quasigroup Cipher, Pola Zig-zag

## I. PENDAHULUAN

Keamanan informasi adalah aspek paling krusial dalam proses pertukaran informasi, terutama melalui media elektronik. Tanpa keamanan yang memadai, informasi dapat disalahgunakan dengan cara yang merugikan. Oleh karena itu, menjaga keamanan informasi agar hanya dapat diakses oleh pihak yang berhak sangat penting [1].

Berbagai upaya dilakukan untuk memastikan bahwa data penting tidak dapat diakses oleh pihak yang tidak berwenang. Salah satu teknik yang umum digunakan untuk mengamankan pesan penting dan rahasia adalah kriptografi. Kriptografi adalah proses mengubah pesan dari bentuk asli (plaintext) menjadi bentuk rahasia (ciphertext). Algoritma RSA adalah salah satu teknik kriptografi yang sering digunakan. Selain kriptografi, steganografi juga merupakan teknik yang efektif untuk mengamankan informasi. Steganografi adalah seni dan ilmu menyembunyikan pesan sehingga keberadaannya tidak

disadari kecuali oleh pengirim dan penerima pesan. Metode yang umum digunakan dalam steganografi adalah Least Significant Bit (LSB), karena kesederhanaannya dan tidak memerlukan komputasi yang rumit. LSB bekerja dengan mengubah bit terakhir pada data cover dengan bit-bit pesan yang disembunyikan. Karena kesederhanaan LSB, diperlukan metode tambahan untuk memastikan pesan yang disembunyikan tidak mudah diketahui. Penelitian ini menggunakan teknik enkripsi sebelum menyembunyikan pesan dan menerapkan pola zig-zag saat proses penyembunyian [2].

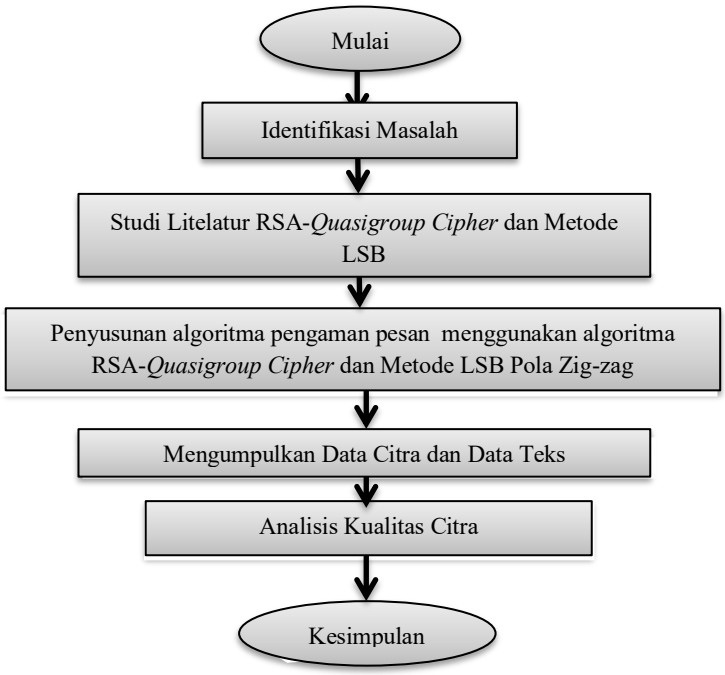
Salah satu teknik kriptografi yang sering digunakan adalah algoritma RSA yang dikombinasikan dengan Quasigroup Cipher menjadi algoritma RSA-Quasigroup Cipher. Dalam teknik ini, kunci dihasilkan menggunakan algoritma RSA dan pesan dienkripsi menggunakan Quasigroup Cipher dengan kunci sesi yang dihasilkan dari algoritma RSA. Dengan cara ini, kelemahan enkripsi yang mudah menimbulkan kecurigaan dapat diatasi dengan steganografi karena pesan tersembunyi dan tidak menarik perhatian. Selain itu, penggunaan pola zig-zag memberikan pengamanan tambahan agar pesan tidak mudah diekstraksi.

Penelitian ini bertujuan untuk mengembangkan algoritma pengamanan pesan teks dalam citra digital dengan mengombinasikan algoritma RSA-Quasigroup Cipher untuk enkripsi dan metode LSB dengan pola zig-zag untuk steganografi. Penggunaan pola zig-zag dalam metode LSB bertujuan untuk menyebarkan data yang disisipkan secara merata ke seluruh citra, sehingga meningkatkan ketahanan terhadap serangan analisis statistik dan membuat deteksi lebih sulit.

## II. METODE PENELITIAN

### 2.1. Data dan alur penelitian

Dalam proses analisis kualitas citra, data yang digunakan sebanyak 20 dan variasi jumlah karakter pada teks yang disisipkan berkisaran antara 100-500 karakter. Alur penelitian sebagai berikut:




**Gambar 1.** Diagram Alur Penelitian

## 2.2. Algoritma RSA

Algoritma RSA merupakan algoritma asimetri yang banyak digunakan saat ini untuk mengamankan data. Algoritma RSA dianggap paling aman dari serangan kriptanalis karena RSA menggunakan algoritma pemfaktoran bilangan yang sangat besar yang sulit difaktorkan menjadi faktor prima.

Adapun algoritma untuk membangkitkan pasangan kunci menggunakan RSA adalah sebagai berikut [3]:

a. Pilih dua bilangan prima sebarang, $p$ dan $q$.
b. Hitung $n = pq$ (sebaiknya $p \neq q$ sebab jika $p = q$ maka $n = p^2$ sehingga $p$ dapat diperoleh dengan menarik akar pangkat dua dari $n$).
c. Hitung $\phi(n) = (p-1)(q-1)$.
d. Pilih kunci publik $e$ yang relatif prima terhadap $\phi(n)$. Dengan demikian, $\big(e, \phi(n)\big) = 1$
e. Bangkitkan kunci privat dengan menggunakan $ed \equiv 1\big(\mathrm{mod}\ \phi(n)\big)$. Perhatikan bahwa $ed \equiv 1\big(mod\ \phi(n)\big)$ ekuivalen dengan persamaan $ed = 1 + k\phi(n)$, sehingga $d = \frac{1+k\phi(n)}{e}$.

Hasil dari algoritma di atas adalah:

- Kunci publik adalah pasangan $e$ dan $n$.
- Kunci privat adalah pasangan $d$ dan $n$.

Selanjutnya kunci publik yang telah dibangkitkan akan digunakan dalam proses enkripsi dan kunci privat digunakan dalam proses dekripsi. Proses enkripsi dan proses dekripsi dengan algoritma RSA menggunakan persamaan

$$c = m^e mod\ n$$

dan

$$m = c^d mod\ n$$

dimana $c = ciphertext$ dan $m = plaintext$

## 2.3. Algoritma Quasigroup Cipher

Algoritma *Quasigroup Cipher* terbagi menjadi proses enkripsi dan proses dekripsi sebagai berikut[4]:

**Algoritma Enkripsi:**

a. Pilih Sebarang bilangan bulat $K, 1 \leq K \leq P-1$ yang mana *quasigroup* $(Q,*)$ terdefinisi untuk elemen $\{1,2,\dots,P-1\}$ dengan persamaan pada lemma 2.2.6.5 dan $P$ adalah sebarang prima yang dipilih.
b. Pilih bilangan bulat acak $a_i, 1 \leq a_i \leq P-2, i = 1,2,\dots,k, k \geq 3$ untuk menjadi $leader$ untuk *quasigroup cipher.*
c. Ubah setiap karakter pada pesan $m_\mu$ menjadi bilangan bulat pada *range* $\{1,2,\dots,P-1\}$, dengan $\mu$ adalah indeks setiap karakter pada pesan. Karakter yang digunakan pada penelitian ini hanyalah 88 karakter karena menggunakan $P = 89$ dan dapat dilihat pada Tabel 1.

**Tabel 1.** Konversi Karakter Menjadi Angka

| **Konversi Karakter menjadi Angka** | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | ! | z | | $ | % | & | ' | ( | ) | * | + | , | - | . | / | 0 | 1 | 2 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | ; | < | = | > | ? | @ | A | B | C | D | E |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
| F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X |
| 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 |

**Konversi Karakter menjadi Angka**

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Z | [ | ] | ^ | a | b | c | d | e | f | g | h | i | j | k | l | m | n |
| 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 |
| o | p | q | r | s | t | u | v | w | x | y | ~ | | | | | | | |
| 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | | | | | | | |

d. Secara berulang hitung $m_\mu^i = a_i * m_\mu^{i-1}$, dimana $m_\mu^0 = m_\mu$, $i = 1,2, \dots, k$ dan $*$ adalah operasi *quasigroup* yang terdefinisi pada Lemma 2.2.6.5
e. $c_\mu = m_\mu^k$ dan *update* nilai *leader* dengan $a_i = m_\mu^i$, $i = 1, \dots, k-1$ dan $a_k = 1 + \left(\sum_{i=1}^{k} m_\mu^i \ mod\ (P-1)\right)$.
f. Diperoleh pesan yang terenkripsi $c_\mu$ *(ciphertext).*

**Algoritma Dekripsi:**

a. Deskripsi *session key* dengan algoritma RSA untuk mendapatkan kembali $K$ dan sejumlah $k$ $leader$ dan membuat $(Q, \backslash)$
b. Secara berulang hitung $c_\mu^k = a_k \backslash c_\mu$, $c_\mu^i = a_i \backslash c_\mu^{i+1}, i = k-1, \dots, 1$ dan $\backslash$ adalah operasi *quasigroup* yang terdefinisi pada Akibat 2.2.6.6
c. $m_\mu = c_\mu^i$ dan *update* nilai *leader* dengan $a_i = c_\mu^{i+1}, i = 1, \dots, k-1$ dan $a_k = 1 + \left(c_\mu + \sum_{i=2}^{k} c_\mu^i \ mod\ (p-1)\right)$
d. Diperoleh *plaintext* $m_\mu$.

## III. HASIL DAN PEMBAHASAAN

### 3.1. Algoritma Pengamanan Pesan Kombinasi RSA-Quasigroup Cipher dan Metode LSB Pola Zig-zag

Algoritma pengamanan pesan menggunakan kombinasi RSA-*Quasigroup Cipher* dan metode LSB pola zig-zag terbagi menjadi 3 proses yaitu proses membangkitkan kunci dimana kunci akan dibangkitkan untuk proses enkripsi dan dekripsi, proses enkripsi-penyisipan dimana pesan akan dienkripsi menjadi *ciphertext* kemudian disisipkan ke dalam citra, serta proses ekstraksi-dekripsi yaitu proses pengambilan *ciphertext* dari citra stego dan kemudian didekripsi untuk mendapatkan pesan asli.

**Proses Bangkitkan Kunci**

Proses bangkitkan kunci adalah proses dimana kunci publik dan kunci privat akan dibangkitkan dengan menggunakan algoritma RSA. Setelah kunci publik dan kunci privat dibangkitkan kunci publik akan dikirim kepada pengirim pesan untuk melakukan enkripsi pesan pada proses enkripsi-penyisipan untuk mendapatkan citra stego. Kunci privat yang telah dibangkitkan akan tetap disimpan untuk proses ekstraksi-dekripsi.

**Proses Enkripsi-Penyisipan**

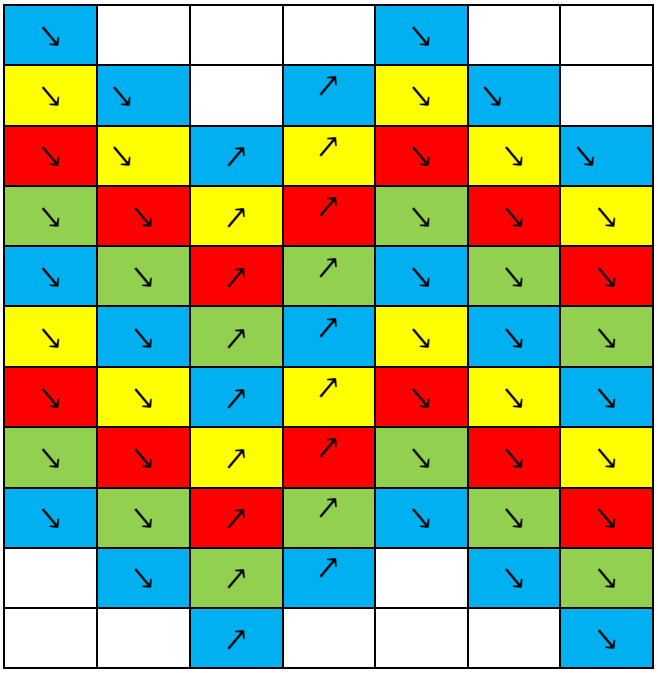
Pada proses ini enkripsi pesan akan dilakukan menggunakan algoritma RSA-Quasigroup Cipher dan penyisipan pesan menggunakan metode LSB pola zig-zag. Adapun langkah-langkah yang dilakukan sebagai berikut:

1. Masukkan gambar, pesan, password, K, dan leader
2. Cek panjang pesan dan ukuran citra penampung. Jika panjang pesan melebihi ukuran citra penampung maka masukkan kembali citra yang berukuran lebih besar atau kurangi panjang pesan

3. Lakukan enkripsi pesan menggunakan Quasigroup Cipher untuk mendapatkan ciphertext
4. Lakukan enkripsi K dan leader menggunakan algoritma RSA
5. Konversi ciphertext dan kunci sesi yang terenkripsi menjadi byte ciphertext dan byte kunci sesi terenkripsi
6. Konversikan citra menjadi matriks citra yang setiap elemennya mewakili piksel citra.
7. Setiap karakter password dikonversikan menjadi angkasehingga bisa diketahui posisi awal penyisipan ciphertext.
8. Byte ciphertext dan byte kunci sesi yang telah terenkripsi disisipkan ke dalam matriks citra mengikuti pola zig-zag dengan ketentuan jika angka konversi karakter pertama dari *password* yang akan digunakan lebih besar dari angka konversi karakter terkahir maka pola zig-zag untuk proses penyesipan pesan akan dimulai dari tepi arah kiri ke arah tepi kanan. Jika tidak, maka pola zig-zag untuk proses penyesipan pesan akan dimulai dari tepi arah kanan ke arah tepi kiri
9. Jika proses penyisipan selesai maka matriks citra dikonversikan kembali menjadi citra stego.

**Ilustrasi Penyisipan Dengan Pola Zig-Zag**

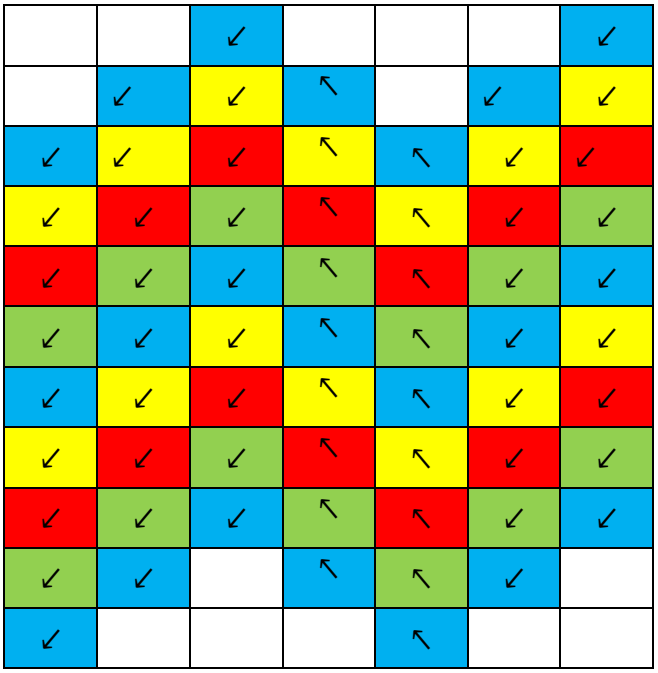
Misalkan matriks citra yang akan disisipi berukuran 11x7 dan diketahui angka konversi karakter pertama dari *password* yang akan digunakan lebih besar dari angka konversi karakter terkahir maka pola zig-zag untuk proses penyesipan pesan akan dimulai dari tepi arah kiri ke arah tepi kanan dan cara penyisipan dimulai dari baris pertama kolom pertama atau titik (1,1) ke arah kanan sampai titik (3,7) selanjutnya proses penyisipan akan berjalan ke arah bawah hingga mencapai titik (11,7)



**Gambar 2.** Ilustrasi proses penyisipan pesan dengan pola zig-zag dari tepi arah kiri ke arah tepi kanan

Jika diketahui angka konversi karakter pertama dari *password* yang akan digunakan lebih kecil dari angka konversi karakter terkahir maka pola zig-zag untuk proses penyesipan pesan akan dimulai dari tepi arah kanan ke arah tepi kiri dan cara penyisipan dimulai dari baris pertama kolom ketujuh

atau titik (1,7) ke arah kiri sampai titik (3,1) selanjutnya proses penyisipan akan berjalan ke arah bawah hingga mencapai titik (11,1).



**Gambar 3.** Ilustrasi proses penyisipan pesan dengan pola zig-zag dari tepi arah kanan ke arah tepi kiri

**Proses Ekstraksi-Dekripsi**

Setelah melakukan proses enkripsi-penyisipan untuk mendapatkan kembali pesan dari citra stego perlu dilakukan proses ekstraksi-dekripsi sebagai berikut:

1. Masukkan citra stego, password, dan kunci private
2. Konversikan citra stego menjadi matriks citra stego yang setiap elemennya mewakili piksel citra.
3. Setiap karakter password dikonversikan menjadi angka sehingga bisa diketahui posisi awal ekstraksi ciphertext.
4. Byte ciphertext dan byte kunci sesi yang telah terenkripsi dapat diekstraksi dari dalam matriks citra stego mengikuti pola zig-zag dengan ketentuan jika angka konversi karakter pertama dari *password* yang akan digunakan lebih besar dari angka konversi karakter terkahir maka pola zig-zag untuk proses ekstraksi pesan akan dimulai dari tepi kiri ke arah tepi kanan. Jika tidak, maka pola zig-zag yang dibentuk dimulai dari tepi kanan ke arah tepi kiri
5. Byte ciphertext dan byte kunci sesi terenkripsi dikonversi menjadi angka
6. Dekripsi kunci sesi menggunakan algoritma RSA
7. Dekripsi ciphertext menggunakan Quasigroup Cipher
8. Konverikan hasil dekripsi ciphertext menjadi plaintext

Berikut ini merupakan ilustrasi pengamanan pesan menggunakan kombinasi RSA-Quasigroup Cipher dan metode LSB pola zig-zag. Misalkan Bob akan mengririmkan pesan kepada Alice maka Alice akan memilih bilangan p=5 dan q=11 serta menghitung kunci public dan kunci privat dengan menggunakan algoritma RSA. Alice memberikan kunci public e = 7 dan n = 55 kepada Bob sedangkan kunci privat d= 23 dirahasiakan.

Bob menggunakan kunci e dan n untuk mengenkripi pesan dengan Quasigroup Cipher sebagai berikut:

1. Pilih sebarang bilangan bulat K dengan $1 \leq K \leq P-1$, dan $P$ adalah sebarang bilangan prima yang dipilih ($P = 89$ sesuai dengan karakter yang dipakai pada tabel konversi). Misalkan $K = 17$
2. Pilih bilangan bulat $a_i, i = 1,2,\cdots,k$ dengan syarat $1 \leq a_i \leq P-2$ dan $k \geq 3$ untuk menjadi *leader*. Misalkan pilih $k = 3$ atau dengan kata lain tentukan nilai $a_1, a_2, a_3$ . Misalkan $a_1 = 36, a_2 = 49, a_3 = 54$
3. Enkripsi $K$ dan masing-masing *leader* $a_i$ dengan algoritma RSA

$$c = m^e mod\ n$$

$$K = 17^7 mod\ 55 = 8, \qquad a_1 = 36^7 mod\ 55 = 31, \qquad a_2 = 49^7 mod\ 55 = 14$$

$$a_3 = 54^7 mod\ 55 = 54$$

4. Mulai enkripsi pesan, misalkan pesan yang akan dikirim adalah "PELAKU". ubah pesan menjadi angka $(m_\mu)$ dalam range $\{1, 2, \dots, P-1\}$. Berdasarkan tabel diperoleh $m_1 = 49, m_2 = 38, m_3 = 45, m_4 = 34, m_5 = 44, m_6 = 54$.
5. Secara berulang hitung $m_\mu^i = a_i * m_\mu^{(i-1)}$ dimana $m_\mu^0 = m_\mu$ , $i = 1,2,3$ dengan $*$ merupakan operasi yang didefinisikan sebagai:

$$i * j = \left(a_i \times \left(1 + \left(K + j\ mod\ (P-1)\right)\right)^{-1} mod\ P\right) mod\ P$$

Enkripsi $m_1 = \mathrm{P}$

- $m_1^1 = a_1 * m_1^0 = \left(a_1 \times \left(1 + \left(K + m_1^0\ mod\ (P-1)\right)\right)^{-1} mod\ P\right) mod\ P$
$= (36 \times (1 + (17 + 49) mod\ 88)^{-1} mod\ 89) mod\ 89)$
$= 144\ mod\ 89 = 55$
- $m_1^2 = a_2 * m_1^1 = \left(a_2 \times \left(1 + \left(K + m_1^1\ mod\ (P-1)\right)\right)^{-1} mod\ P\right) mod\ P$
$= (49 \times (1 + (17 + 55) mod\ 88)^{-1} mod\ 89) mod\ 89)$
$= 2450\ mod\ 89 = 47$
- $m_1^3 = a_3 * m_1^2 = \left(a_3 \times \left(1 + \left(K + m_1^2\ mod\ (P-1)\right)\right)^{-1} mod\ P\right) mod\ P$
$= (54 \times (1 + (17 + 47) mod\ 88)^{-1} mod\ 89) mod\ 89)$
$= 3528\ mod\ 89 = 20$

Diperoleh $c_1 = m_1^3 = 20$

*update leader* baru untuk perhitungan karakter selanjutnya

$$a_1 = m_1^1 = 55, \qquad a_2 = m_1^2 = 47$$

$$a_3 = 1 + \left(\sum_{i=1}^{3} m_1^i\right) mod\ (P-1)$$
$$= 1 + \left((55 + 47 + 20) mod\ 88\right)$$
$$= 35$$

Dengan melakukan langkah yang sama sampai semua karakter pada $m_1$ sampai $m_6$ terenkripsi diperoleh ciphertext $c_\mu = [\,20\ \ 41\ \ 65\ \ 44\ \ 80\ \ 23] = [\,3\ H\ c\ \ K\ r\ 6]$. Hasil ciphertext, hasil

enkripsi k dan hasil enkirpsi leader akan disisipkan kedalam citra menggunakan metode LSB pola zig-zag sehingga menghasilkan citra stego, Bob akan mengirimkan citra stego ini kepada Alice.

Alice akan mengestrak ciphertext enkripsi k dan enkripsi leader dari citra stego dengan menggunakan metode LSB pola zig-zag. Selanjutnya Alice menggunakan kunci privat d dan n untuk melakukan proses dekripsi pesan dengan algoritma quasigroup cipher sebagai berikut:

1. Dekripsi *K* dan masing-masing *leader* yang telah dienkripsi dengan menggunakan algoritma RSA

$$m = c^d mod\ n$$

$$K = 8^{23} mod\ 55 = 17,\ k_1 = 31^{23} mod\ 55 = 36, \qquad k_2 = 14^{23} mod\ 55 = 49$$

$$k_3 = 54^{23} mod\ 55 = 54$$

2. Dekrpisi *cipher* yang telah diperoleh $c_\mu = [\,20\ \ 41\ \ 65\ \ 44\ \ 80\ \ 23]$ dengan menghitung $c_\mu^k = a_k \backslash c_\mu,\ c_\mu^i = a_i \backslash c_\mu^{(i+1)}; i = k-1, \dots, 1$ dengan operasi \ yang didefiniskan oleh persamaan:

$$i \backslash j = \begin{cases} g(i,j,K), & Jika\ g(i,j,K) \neq 0 \\ P-1, & Jika\ g(i,j,K) = 0 \end{cases}$$

Dengan

$$g(i,j,K) = \big((i \times j^{-1} mod\ p) - 1 - K\big) mod\ (P-1)$$

**Dekripsi:**

- Dekripsi $c_1 = 20$

$c_1^3 = a_3 \backslash c_1, \quad c_1^3 = 54 \backslash 20$

Hitung dulu $g(54,20,17)$

$$= \Big(\big((54 \times 20^{-1} mod\ 89) mod\ 89\big) - 1 - 17\Big)\ mod\ 88$$

$$= (65 - 1 - 17) mod\ 88 = 47$$

$c_1^2 = a_2 \backslash c_1^3, \quad c_1^2 = 49 \backslash 47$

Hitung dulu $g(49,47,17)$

$$= \Big(\big((49 \times 47^{-1} mod\ 89) mod\ 89\big) - 1 - 17\Big)\ mod\ 88$$

$$= (73 - 1 - 17) mod\ 88 = 55$$

$c_1^1 = a_1 \backslash c_1^2, \quad c_1^1 = 36 \backslash 55$

Hitung dulu $g(36,55,17)$

$$= \Big(\big((36 \times 55^{-1} mod\ 89) mod\ 89\big) - 1 - 17\Big)\ mod\ 88$$

$$= (67 - 1 - 17) mod\ 88 = 49$$

*Plaintext* diperoleh dari $m_\mu = c_\mu^1$

$$m_1 = c_1^1 = 49$$

*Update leader* baru

$a_i = c_1^{(i+1)}, i = 2{,}1$ dan $a_3 = 1 + \left(c_1 + \sum_{i=2}^{3} c_1^{(i)}\right)\ mod\ (P-1)$

$a_3 = 1 + (20 + 47 + 55) mod\ 88 = 1 + 34 = 35$

$a_2 = 47$, dan $a_1 = 55$

Dengan melakukan perhitungan yang sama diperoleh plaintext sebagai berikut

$m_\mu = [49\ \ 38\ \ 45\ \ 34\ \ 44\ \ 54\,] = [\,P\ \ E\ \ L\ \ A\ \ K\ \ U]$

**Analisis Kualitas Citra**

Untuk melakukan pengujian kualitas citra sebelum dan sesudah penyisipan pesan, digunakan juga MATLAB untuk menghitung PSNR pada masing-masing citra. Berikut ini disajikan hasil pengujian kualitas citra dengan menggunakan 20 citra uji dengan variasi nama citra dimulai dari citra Pohon durian dan laut sampai pada citra Jembatan Merah Putih .

Sedangkan panjang pesan teks memiliki variasi antara 100 karakter hingga 500 karakter. Hasil perhitungan nilai PSNR dari masing-masing citra per masing-masing panjang karakter dapat dilihat pada Tabel 4.1 berikut ini:

**Tabel 2.** Nilai PSNR Hasil Pengujian citra dengan variasi panjang karakter

| Nama Citra | NILAI PSNR (dB) | | | | |
|---|---|---|---|---|---|
| | JUMLAH KARAKTER PESAN | | | | |
| | 100 | 200 | 300 | 400 | 500 |
| Pohon durian | 74,78 | 72,19 | 70,56 | 69,31 | 68,20 |
| Pasir | 82,99 | 79,88 | 78,32 | 77,25 | 76,17 |
| Gedung mipa | 73,64 | 71,02 | 69,00 | 67,92 | 66,82 |
| Labkom | 78,46 | 75,68 | 74,21 | 72,88 | 71,84 |
| Bebatuan | 74,56 | 72,09 | 70,30 | 68,97 | 68,05 |
| Bunga | 76,74 | 74,43 | 72,92 | 71,65 | 70,55 |
| Pohon cengkeh | 73,70 | 70,91 | 69,21 | 68,10 | 67,20 |
| Kucing | 79,71 | 76,46 | 74,85 | 73,77 | 72,75 |
| Masjid | 75,85 | 73,25 | 71,54 | 70,28 | 69,38 |
| Me | 81,54 | 78,75 | 77,14 | 75,93 | 74,98 |
| Ombak | 68,72 | 66,23 | 64,64 | 63,39 | 62,39 |
| Pantai | 74,55 | 71,92 | 70,30 | 68,99 | 68,09 |
| Pemandangan malam | 79,30 | 76,44 | 74,69 | 73,47 | 72,56 |
| Pemandangan pantai | 69,67 | 67,01 | 65,46 | 64,27 | 63,30 |
| Permainan anak | 74,60 | 72,01 | 70,26 | 69,14 | 68,20 |
| Pesawat | 84,52 | 81,86 | 80,18 | 78,88 | 77,99 |
| Pohon kelapa | 79,61 | 76,89 | 75,19 | 73,93 | 72,96 |
| Pohon | 80,78 | 78,24 | 76,57 | 75,31 | 74,30 |
| Rumah makan | 84,21 | 81,36 | 79,69 | 78,41 | 77,31 |
| Jembatan Merah Putih | 85,21 | 82,37 | 80,69 | 79,57 | 78,55 |
| PSNR Terkecil | 68,72 | 66,23 | 64,64 | 63,39 | 62,39 |
| Rata-rata PSNR | | | 73,81 | | |

Berdasarkan Tabel 4.1 terlihat jelas bahwa semakin panjang jumlah karakter pesan yang digunakan, semakin kecil nilai PSNR

## IV. KESIMPULAN

Berdasarkan hasil analisis kulaitas citra dari keseluruhan citra yang digunakan, diperoleh bahwa h nilai PSNR terkecil adalah 62,39 Db dan rata rata nilai PSNR adalah 73,81 dB dengan masing-masing variasi panjang karakter mulai dari 100 karakter hingga 500 karakter. Hasil pengujian memperlihatkan bahawa nilai PSNR seluruh citra dari kombinasi algoritma RSA-Quasigroup Cipher dan metode LSB pola zig-zag bernilai lebih dari 60 dB. Hal ini berarti kualitas citra stego yang dihasilkan sangat baik.

## REFERENSI

1. Mido, A. R., dkk (2022). Analisis Pengaruh Citra Terhadap Kombinasi Kriptografi RSA dan Steganografi LSB. Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK). Vol. 9., No. 2., hlm. 279-286. https://jtiik.ub.ac.id/index.php/jtiik/article/view/4852.

2. Utomo, D.S(n,d)., dkk (2019). Penyembunyian Teks Terenkripsi Pada Citra Rgb Menggunakan Metoda Lsb Dengan Pola Zig-Zag. Jurnal Masyarakat Telematika dan Informasi:Vol.10.,No.2.,hlm.19-29. https://www.researchgate.net/publication/339233235_PENYEMBUNYIAN_TEKS_TERENKRIPSI_PADA_CITRA_RGB_MENGGUNAKAN_METODA_LSB_DENGAN_POLA_ZIG-ZAG
3. Munir, R. (2014). Diktat Kuliah: Sistem Kriptograf Kunci-Publik. Bandung:Departemen Teknik Informatika Institut Teknologi Bandung. https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2020-2021/Algoritma-RSA-2020.pdf&ved=2ahUKEwjw8arKo_6GAxVjRmwGHS6TDbUQFnoECB0QAQ&usg=AOvVaw1dUJO7jOsVA7F_XbFs1iyK
4. Khudzaifah, M. (2014). Aplikasi Quasigroup Dalam Pembentukan Kunci Rahasia Pada Algoritma Hibrida (Rsa-Quasigroup Cipher). *Cauchy, 3*(2), hlm. 55 - 58. https://www.researchgate.net/publication/283932106_Aplikasi_quasigroup_dalam_pembentukan_kunci_rahasia_pada_algoritma_hibrida