ISSN : 1410 – 7104

Implementasi Metode *Arnold's Cat Map* dan *Logistic Map* Pada Proses Enkripsi-Dekripsi Untuk Keamanan Pengiriman Citra

Implementasi Metode Arnold's Cat Map dan Logistic Map Pada Proses Enkripsi-Dekripsi Untuk Keamanan Pengiriman Citra

Mohammad Hamdani[1] dan Novia Listiyani[2]
[1]Program Studi Teknik Elektro, FTI-ISTN
E-mail: mhamdani@istn.ac.id
[2]Transmission Planning Alita Praya Mitra
E-mail: Novialistiyani04@gmail.com

Abstrak --- *Pada makalah ini dibahas tentang implementasi metode Arnold's Cat Map dan metode Logistic Map pada proses enkripsi dan dekripsi untuk keamanan pengiriman citra. Dalam penelitian sebelumnya, dilakukan proses enkripsi dengan hanya menggunakan metode Arnold's Cat Map yang memberikan hasil bahwa proses dekripsinya masih sangat mudah dilacak dengan mengulangi iterasi untuk mendapatkan citra asli. Oleh karena itu dalam penelitian ini pada proses enkripsi dan dekripsi digunakan dua metode, yaitu metode Arnold's Cat Map dan metode Logistic Map untuk menjamin keamanan pengiriman citra. Implementasi metode Arnold's Cat Map dan metode Logistic Map membutuhkan waktu komputasi yang tergantung pada jumlah iterasi yang digunakan saat proses enkripsi dan dekripsi, sedangkan pada pengujian histogram didapatkan hasil histogram yang sempurna dikarenakan pendistribusian pixel gambar hasil enkripsi dilakukan secara merata atau lebih homogen. Dari pengujian didapatkan bahwa koefisien korelasi dari implementasi metode Arnold's Cat Map dan metode logistic mendekati nol, yang berarti bahwa seluruh pixel telah teracak. Pada pengujian sensitivitas kunci didapatkan hasil bahwa dengan perbedaan sebesar 0,00001 antara kunci yang digunakan saat enkripsi dan dekripsi tidak akan dapat menampilkan citra sesuai aslinya yang dikirimkan. Dari pengujian pengiriman data dapat diketahui bahwa waktu yang diperlukan untuk pengiriman data citra yang telah melalui proses enkripsi-dekripsi adalah sama dengan waktu pengiriman data tanpa melalui proses enkripsi, yang berarti bahwa waktu tambahan yang digunakan tidaklah signifikan, yaitu hanya diperlukan untuk proses enkripsi dan dekripsi.*

Kata kunci : *Enkripsi, Metode Arnold's Cat Map, Metode Logistic Map, Dekripsi*

Abstract *--- This paper has discussed the implementation of the Arnold's Cat Map method and the Logistic Map method in the encryption and decryption process for image sending security. In a previous study, an encryption process was carried out by using only the Arnold's Cat Map method which gave results that the decryption process was still very easy to trace by repeating the iteration to get the original image. Therefore in this study the process of encryption and decryption used two methods, namely the Arnold's Cat Map method and the Logistic Map method to ensure the security of image delivery. The implementation of the Arnold's Cat Map method and the logistic map method requires the computation time to depend on the number of iterations used during the encryption and decryption process, while the histogram test shows that the histogram results are perfect because the image distribution of the encrypted image is done evenly or more homogeneously. From the test, it was found that the correlation coefficient of the implementation of the arnold's cat map method and logistic method is close to zero, which means that all pixels have been randomized. In the key sensitivity test results obtained that with a difference of 0.00001 between the keys used when encryption and decryption will not be able to display the image according to the original sent. From the testing of data transmission it can be seen that the time needed for sending image data that has been through the encryption-decryption process is the same as the time of data transmission without going through the encryption process, which means that the additional time used is insignificant, that is only required for the encryption and decryption process.*

Keyword **:** *Encryption, Arnold's Cat Map Method, Logistic Map Method, Decryption*

1. PENDAHULUAN

Dengan adanya perkembangan teknologi maka penyimpanan dan pengiriman media digital seperti citra, audio, dan video menjadi lebih mudah dan efisien. Persoalan yang timbul adalah terdapatnya celah keamanan bagi pihak-pihak yang tidak ber tanggungjawab untuk melakukan pencurian ter hadap data, baik yang tersimpan dalam *hard drive* atau yang ditransmisikan. Salah satu tipe file yang banyak digunakan dan biasanya berisi informasi penting adalah data bertipe gambar atau citra digital.

Salah satu cara untuk menjaga kerahasiaan pengiriman data citra adalah dengan kriptografi, dimana data sumber yang dikirim (plaintext) diubah menjadi bentuk data sandi (ciphertext), dan hanya dapat dikembalikan ke bentuk data sebenar nya dengan menggunakan kunci (key) yang dike tahui oleh pihak tertentu. Pada kriptografi terdapat banyak algoritma diantaranya adalah Chaos yang menggambarkan kebiasaan dari suatu sistem yang terus berubah sehingga memiliki sifat untuk muncul secara acak. Beberapa contoh metode algoritma *Chaos* adalah *Arnold's Cat Map dan Logistic Map*.

Pada penelitian ini dilakukan implementasi algoritma chaos yaitu menggunakan metode *Arnold's Cat Map* dan metode *Logistic Map* pada proses enkripsi dan dekripsi dalam pengiriman citra.

2. TINJAUAN PUSTAKA

2.1. Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai suatu keluaran dari sistem perekaman data yang dapat bersifat analog ataupun bersifat digital. Citra dapat dikelompokan menjadi citra tampak dan citra tak tampak. Foto keluarga, lukisan peman-dangan, hologram (citra optis), dan apa yang tampak di layar monitor dan televisi merupa kan citra tampak, sedangkan data gambar dalam file citra digital, merupakan citra tak tampak.

2.1.1 Citra Analog

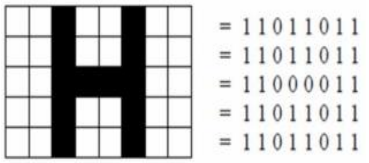
Citra analog adalah citra yang bersifat kontinyu seperti plat nomor kendaraan, gambar pada monitor televisi, foto sinar X, foto yang ter cetak di kertas foto, lukisan, pemandangan alam, hasil CT scan, gambar-gambar yang terekam pada pita kaset dan lain sebagainya. Citra analog tidak dapat direpresentasikan dalam komputer sehingga tidak bisa diproses komputer secara langsung.

2.1.2 Citra Digital

Citra digital adalah citra yang dapat diolah oleh komputer. Umumnya citra digital berbentuk persegi panjang atau bujur sangkar yang biasanya dinyatakan dalam banyaknya titik atau pixel sehingga ukuran citra selalu bernilai bulat. Citra digital dapat didefinisikan sebagai fungsi dua variable f(x,y), dimana x dan y adalah koordinat spasial dan nilai f(x,y) yang merupakan intensitas citra pada koordinat tersebut.

a. Citra Monokrom

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai pixel yaitu hitam dan putih. Citra biner juga disebut sebagai citra B&W (black and white) atau citra monokrom. Pada citra biner setiap titik (pixel) bernilai 0 atau 1, masing-masing mempresentasikan warna tertentu. warna hitam bernilai 0 dan warna putih bernilai 1.



Gambar 1. Citra biner

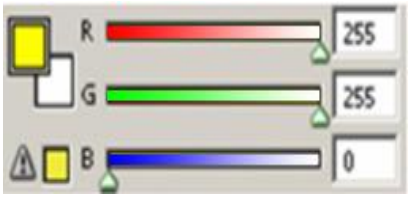
b. Citra *Greyscale*

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap piksel nya, dengan kata lain nilai bagian red, green dan blue memiliki warna yang sama, yaitu warna dari hitam, keabuan, dan putih.

Gambar 2. *Palet grayscale* pada nilai bagian *Red, Green dan Blue*

c. Citra Warna

Pada citra warna, setiap titik mempunyai warna yang spesifik yang merupakan kombinasi dari 3 warna dasar, yaitu merah, hijau, dan biru. Format citra ini sering disebut sebagai citra RGB (red-green-blue). Misalnya warna kuning merupa kan kombinasi warna merah dan hijau sehingga nilai RGB-nya adalah (255 255 0). Dengan demikian setiap titik (pixel) pada citra warna mem butuhkan data 3 byte.



Gambar 3. Citra warna Kuning

2.2 Metode Arnold's Cat Map

Algoritma *Arnold's Cat Map* (ACM) ditemu kan oleh Vladimir Arnold pada tahun 1960. ACM adalah algoritma yang melakukan enkripsi dengan mentransformasikan koordinat pixel (x, y) pada ukuran citra yang berukuran N x N ke koordinat baru (*x',y'*), persamaan matematis yang digunakan:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} mod(N) \quad \text{..... (2.1)}$$

Dimana:

X_{i+1} dan Y_{i+1}: Posisi pixel baru

b dan c : Kunci rahasia dengan semua bilangan bulat positif

X_i dan Y_i : Posisi pixel semula

N : Ukuran citra N x N

Citra yang sudah teracak oleh ACM dapat direkonstruksi menjadi citra semula dengan menggunakan kunci yang sama (a, b, dan c). Persamaan iterasinya adalah :

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix}^{-1} \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} mod(N) \text{ ..... } (2.2)$$

2.3 Metode Logistic

Untuk melakukan enkripsi menggunakan logistic map, terdapat beberapa proses antara lain adalah :

1) Membangkitkan *keystream* menggunakan *Chaos* Pada proses membangkitkan kunci ini mengguna kan rumus sebagai berikut :
   $x_{i+1} = rx_i(1 - x_i)$ ..... (2.3)
2) Fungsi Pemotongan Enkripsi dan dekripsi beroperasi dalam himpunan bilangan bulat yang nilainya dari 0 sampai 255, sedangkan barisan nilai *chaos* yang digunakan sebagai *keystream* adalah bilangan riil antara 0 dan 1. Agar barisan nilai *chaos* dapat dipakai untuk enkripsi dan dekripsi, maka nilai *chaos* harus dikonversi ke nilai integer menggunakan rumus sebagai berikut:
   $T(x, size) = \|x * 10^{count}\| x \neq 0$ ..... (2.4)
3) Enkripsi Plaintext dengan Kunci .Enkripsi dikerjakan dengan menjumlahkan plainteks p_i dan *keystream* k_i dalam modulo 256, seperti yang dituliskan dalam persamaan 2.5.
   $Enkripsi = (p_i + k_i)\ mod\ 256$..... (2.5)
4) Deskripsi Chipertext menggunakan kunci. Dekripsi dikerjakan dengan mengurangkan cipherteks c_i dengan *keystream* k_i dalam modulo 256, seperti yang dituliskan dalam persamaan 2.6:
   $Dekripsi = (c_i - k_i)\ mod\ 256$ ..... (2.6)

2.4 Parameter Keamanan Enkripsi Citra

2.4.1 Sensitivitas Kunci

Sensitivitas kunci merupakan hal yang sangat penting dalam sistem kriptografi. Pengujian sensitivitas kunci bertujuan untuk melihat hasil dekripsi yang dilakukan menggunakan kunci yang berbeda. Jika gambar asli dapat didekripsi atau dapat terlihat mirip menggunakan kunci yang salah, maka algoritma enkripsi yang diterapkan tidak dapat digunakan.

2.4.2 Histogram

Histogram digunakan untuk analisis dalam mengetahui informasi dari penyebaran nilai pixel, distribusi nilai pixel pada gambar biasanya ber konsentrasi pada sebagian ruang nilai pixel, Enkripsi yang baik menyebabkan nilai pixel menyebar disepanjang ruang nilai pixel, selain itu histogram dari cipher image harus berbeda dari plain image. Jika histogram pada cipher image dan plain image memiliki kemiripan, maka penyerang dapat melakukan analisis statistik untuk mendapat kan beberapa informasi.

2.4.3 Koefisien Korelasi

Koefisien korelasi digunakan untuk analisis dalam penentuan hubungan antara dua variabel untuk mengetahui kualitas enkripsi dari kripto sistem. Enkripsi citra dikatakan bagus, jika algoritma enkripsi yang digunakan mengaburkan hubungan dari plain image, dan cipher image yang dihasilkan benar-benar acak dan tidak memiliki korelasi.

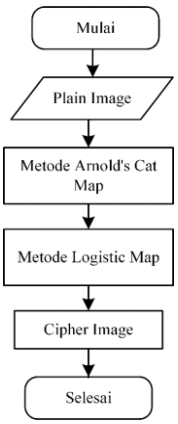
Tabel 1. Interprentasi Nilai Koefisien Relasi

Nilai	Sifat
$\|r\| = 0{,}0$	Tidak ada korelasi
$0{,}0 < \|r\| < 0.2$	Korelasi Sangat Lemah
$0{,}2 \leq \|r\| < 0.4$	Korelasi Lemah
$0{,}4 \leq \|r\| < 0.6$	Korelasi Cukup
$0, 6 \leq \|r\| < 0.8$	Korelasi Kuat
$0{,}8 \leq \|r\| \leq 1{,}0$	Korelasi Sangat Kuat
$\|r\| = 1{,}0$	Korelasi Sempurna

3 METODE

3.1 Perancangan Enkripsi Citra

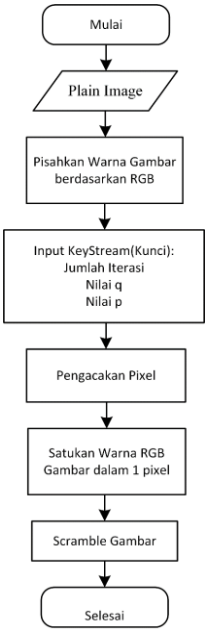
Perancangan sistem pada penelitian ini meliputi 2 proses, yakni untuk enkripsi dan dekripsi pada citra digital. Skema perancangan yang dibuat yakni sebagai berikut :



Gambar 4. Skema Perancangan Enkripsi

3.1.1 Enkripsi Citra Menggunakan Metode Arnold's Cat Map Dan Logistic Map

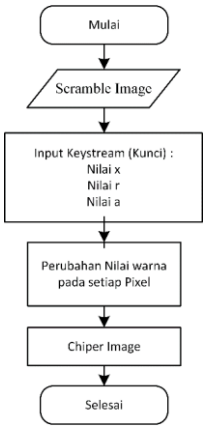
Proses enkripsi citra menggunakan metode *Arnold's Cat Map* dan *Logistic Map* dapat diperlihatkan pada gambar 5.



Gambar 5. Diagram Alir Enkripsi Citra Metode Arnold's Cat Map

3.1.2 Enkripsi Citra menggunakan Metode Logistic Map

Scrambled image menjadi masukan pada proses enkripsi menggunakan Metode *Logistic Map*. Pengubahan nilai warna pada *scrambled image* dilakukan dengan memasukkan *keystream* (Kunci) yang meliputi nilai x, nilai r, dan nilai a. Nilai acak yang dibangkitkan *Logistic Map* meng hasilkan bilangan riil. Untuk menghasilkan bilang an *integer*, maka nilai acak tersebut diproses dengan mengubah menjadi bilangan desimal yang kemudian dilakukan proses modulo dengan 256 dikarenakan rentang *integer pixel* yakni [0, 255]. Hasil dari modulo inilah yang disebut *key stream*,hal ini dapat diperlihatkan pada gambar 6.

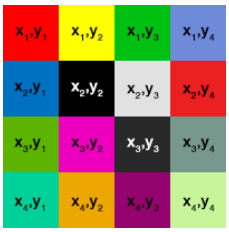


Gambar 6. Diagram Alir Proses Enkripsi Citra dengan Metode *Logistic Map*

3.1.3 Proses Enkripsi Menggunakan Metode Arnold's Cat Map Dan Logistic Map

Perhitungan matematis pada proses enkripsi dapat disimulasikan seperti terlihat pada gambar 7 s.d gambar 11. Terdapat citra RGB berukuran 4x4 piksel, proses yang pertama dilakukan adalah proses pengacakan menggunakan metode *Arnold's Cat Map*.

Gambar 7. Citra Awal 4x4 piksel

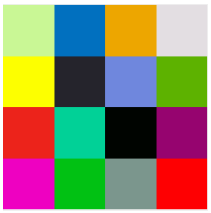


Gambar 8. Citra Awal dengan koordinat ACM

Proses yang dilakukan menggunakan rumus 2.1. dengan Iterasi = 1, Banyaknya percobaan yang dilakukan adalah 1

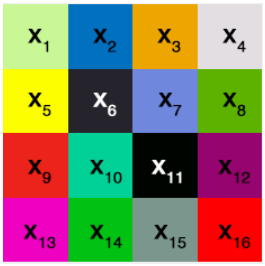
b = 2, Key untuk metode ACM
c = 2, Key untuk metode ACM
N = 4, Jumlah Pixel

Dari perhitungan didapat hasil seperti terlihat pada gambar 9.



Gambar 9. Hasil dari proses enkripsi *Arnold's Cat Map*

Proses selanjutnya adalah melanjutkan pengacakan terhadap hasil ACM tersebut menggunakan algoritma *Logistic Map*.



Gambar 10. Pixel 4x4 dengan asumsi Xn

Proses yang dilakukan menggunakan rumus 2.3, 2,4 dan 2,5.

r = 4, $0 \leq r \leq 4$, r adalah konstanta yang menentukan perubahan warna, semakin kecil nilai r maka perubahan warna tidak signifikan, begitu sebaliknya semakin besar nilai r maka perubahan warna akan terjadi secara signifikan.

x_0 = 0,87, $0 \leq r \leq 4$, x_0 adalah *key* awal yang ditentukan. Setelah menentukan *key* awal yang digunakan maka akan didapat untuk nilai *key* selanjutnya x_1, x_2, x_3,.....dst.

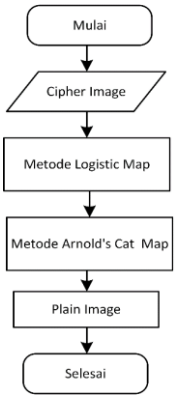
a = 10^4, a adalah bilangan yang digunakan untuk mengkonversi nilai x yang berupa bilangan riil men jadi bilangan integer.

Dari perhitungan didapat hasil seperti terlihat pada gambar 11.

Gambar 11. Hasil dari proses enkripsi *Logistic Map*

3.2 Perancangan Dekripsi Citra

Proses dekripsi merupakan proses pengubahan *cipher image* menjadi citra semula yakni *plain image* dengan menggunakan nilai parameter masukan yang sama dengan proses enkripsi yang telah dilakukan. Proses dekripsi seperti terlihat pada gambar 12 dilakukan untuk mengubah citra terenkripsi atau *cipher image* menjadi citra asli atau *plain image*.



Gambar 12. Diagram Alir Perancangan Dekripsi

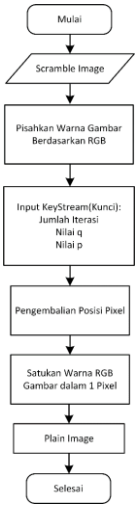
3.2.1 Dekripsi Gambar Metode *Logistic Map*

Proses dekripsi merupakan proses kebalikan dari enkripsi. Pada proses dekripsi diawali dengan masukan yang berupa *cipher image*, langkah selanjutnya adalah input nilai *keystream* (kunci). Pada proses dekripsi untuk nilai *keystream* (kunci) bernilai sama dengan proses enkripsi dikarenakan algoritma ini termasuk dalam algoritma simetri. Selanjutnya setelah dilakukan *input keystream* (kunci) *cipher image* tersebut akan melakukan proses pengembalian nilai *pixel*. Pada proses ini menghasilkan *scrambled image*.

Gambar 13. Diagram Alir Proses Dekripsi Citra Dengan Metode *Logistic Map*

3.2.2 Dekripsi Citra Metode *Arnold's Cat Map*

Pada proses dekripsi citra menggunakan metode *arnold's cat map* ini nilai masukannya adalah *scramble image* dari hasil dekripsi meng gunakan metode *logistics map*, selanjutnya *scramble image* dipisahkan warnanya berdasarkan RGB, kemudian input nilai *keystream* (kunci), untuk nilai *keystream* (kunci) bernilai sama dengan proses enkripsi, setelah dilakukan input *keystream* (kunci) maka akan dilakukan pengembalian posisi pixel sesuai dengan posisi awal, selajutnya satukan warna RGB dalam satu pixel maka akan dihasilkan plain image.



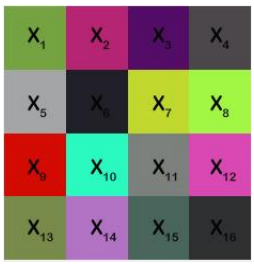
Gambar 14. Diagram Alir Proses Dekripsi Citra Metode *Arnold's Cat Map*

3.2.3 Proses Dekkripsi Menggunakan Metode *Arnold's Cat Map* dan *Logistic Map*

Citra yang digunakan pada proses ini adalah citra hasil proses enkripsi yang sudah dilakukan

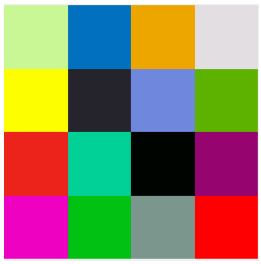
sebelumnya, proses yang pertama dilakukan adalah proses pengembalian warna pada setiap pixel citra menggunakan metode *logistic Map*.

Gambar 15. Gambar enkripsi logistic



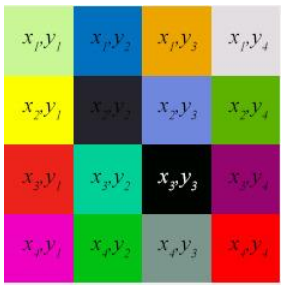
Gambar 16. Gambar dengan asumsi nilai Xn

Pada proses dekripsi ini asumsi perhitungan yang dipakai sama dengan pada saat proses enkripsi yaitu r = 4; x_0 = 0,87; a = 10^4, perhitungan menggunakan rumus 2.3, 2.4, 2.6.
Dari perhitungan didapat hasil seperti terlihat pada gambar 17.



Gambar 17.Hasil Dekripsi menggunakan *Logistic Map*

Proses dekripsi selanjutnya adalah mengembali kan posisi pixel citra awal, proses dekripsi ini dilakukan dengan menggunakan metode *arnold's cat map.*

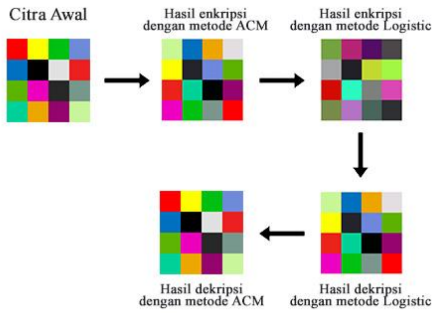


Gambar 18. Gambar Hasil Dekripsi dengan koordinat ACM

Pada proses dekripsi ini asumsi perhitungan yang dipakai sama dengan pada saat proses enkripsi yaitu Iterasi = 1; b = 2, c = 2; N = 4 dan menggunakan rumus 2.2.
Dari perhitungan didapat hasil seperti terlihat pada gambar 19.

Gambar 19. Hasil dekripsi dengan metode arnold's cat map

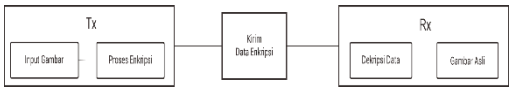
Keseluruhan proses enkripsi dan dekripsi mengguna kan *metode Arnold's Cat Map* dan *Logistic Map* terlihat pada gambar 20.



Gambar 20. Proses perubahan Citra pada proses enkripsi dan dekripsi

3.3 Pengiriman Data Citra

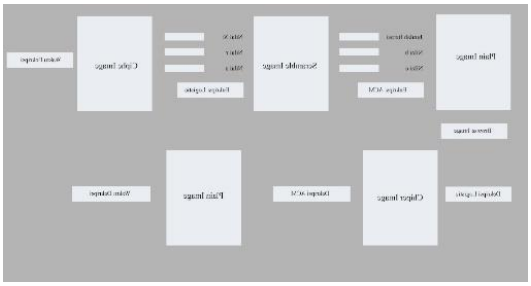
Proses pengiriman data dimulai dengan melakukan input gambar, selanjutnya gambar di enkripsi menggunakan metode *Arnold's Cat Map* dan *Logistic Map*. Citra yang telah di enkripsi ini lah yang akan dikirim ke penerima, selanjutnya setelah gambar sampai di penerima akan dilakukan dekripsi pada citra tersebut menggunakan metode *Arnold's Cat Map* dan *Logistic Map*. Proses dekripsi dilakukan menggunakan *keystream* yang sama pada proses enkripsi.



Gambar 21. Proses Pengiriman Data

3.4 Perancangan Tampilan GUI

GUI ini digunakan untuk membantu proses enkripsi dan dekripsi menggunakan metode *Arnold's Cat Map* dan *Logistic Map* pada Citra Yang berukuran besar. Berikut adalah perancangan untuk tampilan pada GUI matlab.



Gambar 22. Tampilan GUI Matlab

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Fungsionalitas

Pengujian fungsionalitas dilakukan untuk mengetahui apakah sistem GUI yang dibuat untuk membantu proses enkripsi dapat berfungsi dengan baik pada tiap-tiap metode.

Gambar 23. Tampilan Gui Pada Input Gambar

Tabel 2. *Keystream* perhitungan manual

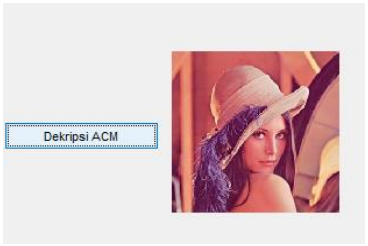
Keystream	Nilai
Iterasi	1
a	2
b	2
x_0	0,87
r	4
a	10000

Gambar 24. Tampilan GUI Menampilkan Hasil Enkripsi ACM

Gambar 25. Tampilan Gui Pada Metode *Logistic Map*

Gambar 26. Tampilan GUI Menampilkan Hasil Enkripsi Logistic

Gambar 27. Tampilan GUI Menampilkan Hasil Dekripsi Logistic

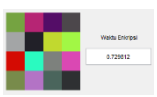
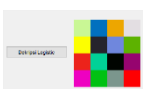


Gambar 28. Tampilan GUI Menampilkan Hasil Dekripsi ACM

4.2 Pengujian Metode *Arnold's Cat Map* dan *Logistic Map*

Pada pengujian ini dilakukan untuk menguji apakah perhitungan manual yang dilakukan pada proses perancangan mendapatkan hasil yang sama saat dilakukan percobaan menggunakan matlab. *Key* yang digunakan pada pengujian ini adalah sebagai berikut :

Tabel 3. Perbandingan perhitungan manual dengan percobaan Matlab

Plain Image	Proses	Hasil Perhitungan Manual	Hasil Percobaan Matlab
	Enkripsi Menggunakan Metode ACM		
	Enkripsi Menggunakan Metode Logistic		
	Dekripsi Menggunakan Metode Logistic		

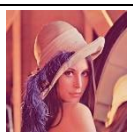
	Dekripsi Menggunakan Metode ACM		

Pada tabel 4 untuk proses enkripsi dan dekripsi menggunakan metode *arnold's cat map* dan metode *logistic map* dengan cara perhitungan manual menghasilkan gambar yang sama dengan proses enkripsi dan dekripsi menggunakan matlab.

4.3 Skenario Pengujian Keamanan

Pada pengujian keamanan ini menggunakan tiga data gambar yang memiliki ukuran berbeda, dan *keystream* yang digunakan adalah sebagai berikut :

Tabel 4. Data Gambar

Gambar	Ukuran Pixel Gambar	Ukuran Gambar
	256x256	11,5 KB
	512x512	307 KB
	1024x1024	273 KB

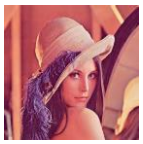
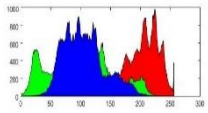
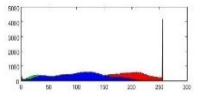
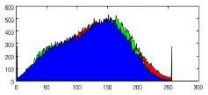
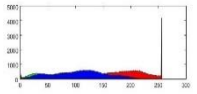
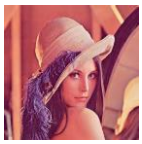
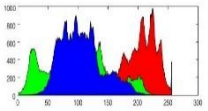
Tabel 5. Data *Keystream*

Key Stream	**Nilai**
Iterasi	Iterasi 5 Iterasi 10 Iterasi 15 Iterasi 20
a	2
b	2
x_0	0,2345
r	4
a	1000

4.3.1 Analisa Histogram

Analisis histogram menunjukkan grafik visual tentang pendistribusian nilai *pixel*. Analisis histogram diuji pada *plain image* dan *cipher image* yang kemudian menghasilkan grafik pada kedua nya. Seperti yang terlihat pada Tabel 5.

Tabel 6. Hasil Pengujian Histogram gambar 256 x 256 Pixel

Gambar	Histogram	Keterangan
		Plain Image
		Enkripsi ACM
		Enkripsi Logistic Map
		Dekripsi Logistic
		Dekripsi ACM

4.3.2 Waktu Komputansi

Pengujian dilakukan untuk mengetahui pengaruh iterasi pada waktu komputasi, yakni pada proses enkripsi dan dekripsi. Pada pengujian ini dilakukan sebanyak 5 kali percobaan pada setiap gambar degan menggunakan nilai iterasi yang sama, gambar yang dilakukan untuk pengujian ber ukuran 256x256 pixel, 512x512 pixel, 1024x1024 pixel.

Tabel 7. Pengujian Waktu Komputasi Enkripsi

Jumlah Iterasi	Ukuran Gambar		
	256x256 Pixel	512x512 Pixel	1024x1024 Pixel
1	2,979158	11,42606	46,08026
5	3,315136	12,57866	49,5774
10	3,931224	14,49742	59,59824
15	4,327934	17,09752	69,64398
20	4,937336	19,55088	79,65174

Tabel 8. Pengujian Waktu Komputasi Dekripsi

Jumlah Iterasi	Ukuran Gambar		
	256x256 Pixel	512x512 Pixel	1024x1024 Pixel
1	2,846906	11,69204	44,14124
5	3,195682	12,42336	49,81314
10	3,78438	14,66444	59,27228
15	4,320854	17,13882	69,3341
20	4,895428	19,75012	80,52168

4.3.3 Koefisien Korelasi

Koefisien korelasi digunakan untuk mengetahui hubungan antara *pixel* yang berdekatan. Parameter ini diuji pada *plain image* dan *cipher image*. Untuk menghasilkan algoritma enkripsi yang baik, maka korelasi antara *pixel* yang berdekatan harus diturunkan menjadi hingga mendekati 0. Pada pengujian ini dilakukan empat kali pada nilai iterasi yang berbeda yaitu 5,10,15,20 untuk parameter p=2, q=2, X_0=0.2345, r=4, a=1000.

Tabel 9. Hasil Pengujian Koefisien Korelasi

Gambar	Ukuran	Koefisien Korelasi Enkripsi			
		Iterasi 5	Iterasi 10	Iterasi 15	Iterasi 20
Lena	256	0,0054	0,0036	0,0020	0,0023
Bunga	512	0,00047	0,0015	0,0017	0,0025
Burung	1024	0,0011	0,0007	0,0013	0,0008

4.3.4 Analisis Sensitivitas Kunci

Sensitivitas kunci merupakan hal yang sangat penting dalam sistem kriptografi. Pengujian sensitivitas kunci bertujuan untuk melihat hasil dekripsi yang dilakukan menggunakan kunci yang berbeda. Teori ini dapat diuji dengan cara melaku kan dekripsi menggunakan kunci berbeda dengan kunci enkripsi. Pada pengujian ini nilai X_0 yang digunakan untuk enkripsi yaitu 0,2345, kemudian dilakukan perubahan penambahan 0,0001 sehingga kunci deksripsi yang digunakan menjadi X_0 = 0.2346. Dengan perubahan ini pada tabel 10 terlihat bahwa gambar yang dihasilkan tidak dapat menampilkan gambar aslinya.

Tabel 10. Hasil Pengujian Sensitivitas Kunci

Plain Image	Cipher Image	Dekripsi
	0,02345	X0 = 0.02346
		
		
		

4.3.5 Pengujian Pengiriman Data

Pengujian ini dilakukan dengan skenario seperti yang ditunjukkan pada gambar 21. Pengujian ini dilakukan untuk mengetahui apakah ada perbedaan ukuran gambar stelah dilakukan enkripsi, dan mengetahui waktu yang dibutuhkan untuk melakukan pengiriman gambar jika gambar harus dilakukan proses enkripsi terlebih dahulu kemudian didekripsi. Setelah itu dibandingkan dengan pengiriman gambar tanpa harus dilakukan enkripsi.

Tabel 11. Hasil Pengujian Pengiriman Gambar Tanpa Proses Enkripsi

Gambar	Ukuran Gambar (KB)	Waktu Kirim (Sekon)	Ukuran Gambar Disisi Penerima (KB)	Waktu Total (Sekon)
256x256	11,5	0,199836	11,9	0,199836
512x512	307	0,350147	96,2	0,350147
1024x1024	273	1,274818	98,2	1,274818

Tabel 12. Hasil Pengujian Pengiriman Gambar Melalui Proses Enkripsi

Gambar	Waktu Enkripsi (Sekon)	Waktu Kirim (Sekon)	Waktu Dekripsi (Sekon)	Waktu Total (Sekon)
256x256	3,31514	0,214256	3,19568	6,725076
512x512	12,57866	0,319565	12,42336	25,321585
1024x1024	49,5774	0,879935	49,81314	100,270475

Pada tabel 12 terlihat bahwa waktu yang dibutuhkan untuk melakukan pengiriman data gambar lebih lama dikarenakan terdapat waktu enkripsi dan dekripsi pada sisi pengirim dan penerima.

5. SIMPULAN

Berdasarkan hasil penelitian dan percobaan dapat diambil simpulan sebagai berikut :

1. Dengan menggunakan metode *Arnold's Cat Map* dan metode *Logistic Map* pada proses enkripsi citra, dapat menjaga keamanan saat pengiriman citra. Hal ini dikarenakan citra hasil enkripsi akan terlihat acak sehingga citra asli tidak terlihat saat proses pengiriman. Setelah dilakukan proses dekripsi disisi penerima maka akan didapat citra sesuai asli nya tanpa ada komponen yang hilang.
2. Proses enkripsi dan dekripsi menggunakan metode *Arnold's Cat Map* dan *Logistic Map* dengan perhitungan manual menghasilkan data gambar enkripsi yang sama dengan meng gunakan percobaan pada matlab.
3. Waktu enkripsi ditentukan oleh besarnya ukuran citra semakin besar ukuran citra maka semakin lama waktu yang dibutuhkan, demikian pula dengan nilai iterasi yang digunakan semakin besar nilai iterasi yang digunakan maka semakin lama waktu yang dibutuhkan namun hal ini menyebabkan pengiriman citra lebih aman karena dengan nilai iterasi yang besar citra akan semakin teracak, pada percobaan ini nilai iterasi yang

digunakan adalah 20 sudah cukup untuk meng amankan citra. Begitupun untuk proses dekripsi. Selain itu pada saat pengiriman tidak terdapat perbedaan waktu antara pengiriman citra asli dengan chiper image, hanya terdapat penambahan waktu saat proses enkripsi dan dekripsi.

4. Pada pengujian sensitivitas *keystream* dapat terlihat bahwa dengan mengubah satu *keystream* pada proses dekripsi menyebabkan gambar yang dikirimkan tidak telihat sesuai dengan citra asli.
5. Dengan menggunakan metode *Arnold's Cat Map* dan metode *Logistic Map* dapat terlihat bahwa *histogram* citra menjadi lebih homogen warnanya sehingga akan menyulitkan pihak tertentu untuk mengetahui *histogram* asli. Dengan demikian citra yang dikirim menjadi lebih aman.
6. Ditinjau dari koefisien korelasi, hasil percoba an menunjukkan nilai mendekati 0, hal ini memperlihatkan bahwa antar pixel pada citra yang dikirim tidak ada kemiripan warna pada citra awal sehingga citra yang dikirimkan akan lebih aman.

DAFTAR PUSTAKA

[1] Andi. 2003, *Memahami Model Enkripsi dan Security Data*. Yogyakarta: Wahana Komputer.

[2] Deswanti, F. M. 201. *Simulasi dan Analisa Steganografi Citra Digital Menggunakan Enkripsi Berdasarkan Prinsip Kubus Rubik Dan Kode Bch.* Bandung : Universitas Telkom.

[3] Huang, M.-Y., Huang, Y.-M., & Wang, M.-S. 2010. *Image Encryption Algorithm Based on Chaotic Map*. *Computer Symposium (ICS) International IEEE Xplore*, 154-158.

[4] Krisnawati. 2009. *Kompresi Citra RGB Dengan Metode Kuantisasi.* Yogyakarta : STIMIK AMIKOM.

[5] Kurniawan, Y. 2004. *Kriptografi Keamanan Internet dan Jaringan Komunikasi.* Bandung: Informatika.

[6] Mohammad Hamdani, Gloria Natalia Samosir, *Implementasi Steganometri Untuk Pengaman an Citra Dijital Menggunakan Metode DCT (Discrete Cosine Transform)*, ISSN 1411-4593 Sinusoida Vol. XX No. 2, 2018, ejournal.istn.ac.id

[7] Stinson, R. 2002. *Cryptography Theory and Practice 2nd Edition.* London: Boca Raton.

[8] Syah, R. D. 2015. *Aplikasi Enkripsi Citra Digital Berbasis Chaos Dengan Algoritma Arnold's Cat Map Menggunakan Matlab.* Jakarta : Universitas Gunadarma.

[9] T, Sutoyo. 2009. *Teori Pengolahan Citra Digital.* Yogyakarta : Andi.

[10] Wijaya, F. H. 2015. *Analisis Algoritma Cat Map Untuk Keamanan Data Citra Satelit-nano Pada Low Earth Orbit.* Bandung : Telkom University.