Jurnal Ilmiah Matematika dan Pendidikan Matematika (JMP)

Vol. 10 No. 2, Desember 2018

ISSN (Cetak) : 2085-1456; ISSN (Online) : 2550-0422

JURUSAN MATEMATIKA

FMIPA Universitas Jenderal Soedirman

Dewan Redaksi

Dewan Editor

Daftar Isi

Vol. 10 No. 2 Desember 2018

Jurnal Ilmiah Matematika dan Pendidikan Matematika (JMP)
Vol. 10 No. 2, Desember 2018, hal. 49-62
ISSN (Cetak) : 2085-1456; ISSN (*Online*) : 2550-0422

PENYEMBUNYIAN *CIPHERTEXT* ALGORITMA GOST PADA CITRA KE DALAM AUDIO DENGAN METODE *LEAST SIGNIFICANT BIT*

Abduh Riski

Mathematical Optimization and Computation Research Group, Jurusan Matematika, Fakultas MIPA, Universitas Jember
riski.fmipa@unej.ac.id

Heri Purwantoro

Jurusan Matematika, Fakultas MIPA, Universitas Jember

Ahmad Kamsyakawuni

Mathematical Optimization and Computation Research Group, Jurusan Matematika, Fakultas MIPA, Universitas Jember

ABSTRACT. *Government Standard (GOST) is a 64-bit block cipher algorithm with 32 round, use a 256-bit key. The weakness of this algorithm is the keys so simple, than make cryptanalyst easy to break this algorithm. Least Significant Bit (LSB) use to insert message into another form without changing the form of the cover after insertion. This research does by hiding encrypted ciphertext to image and hiding image into audio. This research use grayscale and RBG image with BMP and PNG format. Audio using music with wav format. Security analysis using differential analysis NPCR and UACI. Security analysis aims to calculate percentage from cover after hiding the message. The smaller the NPCR and UACI values, the higher the level of security the message is hidden. The results of the analysis of concealment in the image obtained by the average values of NPCR and UACI were 99.98% and 3.46% respectively. While the results of the analysis of hiding in audio obtained the average value of NPCR and UACI were 83.78% and 12.66% respectively.*

Keywords*: audio, ciphertext, image, GOST cryptography, LSB steganography, security analysis.*

ABSTRAK. *Government Standard* (GOST) merupakan algoritma block cipher 64 bit dengan jumlah putaran 32 round menggunakan 256 bit kunci. Kelemahan algoritma ini adalah kunci yang sederhana, sehingga memudahkan kriptanalis untuk memecahkan algoritma ini. *Least Significant Bit* (LSB) digunakan untuk menyisipkan pesan ke dalam bentuk lain tanpa merubah bentuk media cover setelah penyisipan. Penelitian ini dilakukan dengan menyisipkan ciphertext yang telah dienkripsi pada gambar dan menyembunyikan gambar pada audio. Penelitian ini menggunakan citra grayscale dan RGB dengan format BMP dan PNG. Audio menggunakan musik dengan format wav. Analisis keamanan menggunakan analisis diferensial NPCR dan UACI. Analisis keamanan bertujuan untuk mengetahui persentase dari cover setelah penyisipan pesan. Semakin kecil nilai NPCR dan UACI, maka semakin tinggi tingkat keamanan pesan yang disembunyikan. Hasil analisis penyembunyian pada citra diperoleh rata-rata nilai NPCR dan UACI secara berturut-turut adalah 99,98% dan 3,46%. Sedangkan hasil analisis

penyembunyian pada audio diperoleh rata-rata nilai NPCR dan UACI secara berturut-turut adalah 83,78% dan 12,66%.

Kata Kunci: analisis keamanan, audio, ciphertext, gambar, kriptografi GOST, steganografi LSB.

1. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi mempunyai dampak yang besar bagi persebaran pesan, baik bersifat umum maupun rahasia. Pesan rahasia harus sampai kepada penerima dengan aman tanpa diketahui pihak yang tidak berkepentingan. Pada penelitian ini akan dilakukan pengamanan pesan menggunakan *Government Standard* (GOST) *cipher* (Sah, 2012; Iqbal 2016) dengan merubah pesan awal menjadi sandi yang sulit dimengerti orang lain, kemudian disembunyikan ke dalam bentuk lain (Wirawan, 2011; Wahyuningsih, 2016) agar keamanan pesan lebih terjamin menggunakan algoritma *Least Significant Bit* (LSB). Pengamanan pesan dilakukan dengan merubah pesan menjadi sandi (*ciphertext*) kemudian disembunyikan pada gambar. Gambar hasil penyembunyian *ciphertext* disembunyikan kedalam audio dengan tujuan pesan yang telah dienkripsi menjadi *ciphertext* dan disembunyikan pada gambar lebih terjamin keamanannya. Analisis keamanan dilakukan guna mengetahui tingkat perubahan dari media *cover* setelah penyisipan.

2. TINJAUAN PUSTAKA

2.1 Government Sandard (GOST)

GOST merupakan singkatan dari *Gosudarstvennyi Standard* atau *Government Standard*, merupakan suatu algoritma *block cipher* yang dikembangkan oleh warga kebangsaan Uni Soviet. Algoritma GOST dikembangkan oleh pemerintah Uni Soviet pada masa perang dingin tahun 1970, digunakan untuk menyembunyikan data maupun informasi yang bersifat rahasia. Algoritma GOST merupakan sebuah algoritma enkripsi dengan proses 32 *round* (putaran) yang menggunakan 64 bit *block cipher* dan 256 bit *key* (Munir, 2006).

GOST menggunakan 8 buah *S-Box* yang berbeda dan menggunakan operasi *XOR*. GOST menggunakan kunci 256 bit dengan rincian

$k_1, k_2, k_3, \dots, k_{256}$. Kunci tersebut dimasukkan ke dalam 8 kelompok yaitu, $K_1, K_2, K_3, K_4, K_5, K_6, K_7$, dan K_8 (Iqbal, 2016).

Terdapat 8 buah blok kunci, K_1 sampai dengan K_8. $K_1 = (k_{32}, \dots, k_1)$, $K_2 = (k_{64}, \dots, k_{33})$, K_3 = (k_{96}, ..., k_{65}), K_4 = (k_{128}, ..., k_{97}), K_5 = (k_{160}, ..., k_{129}), K_6 = (k_{192}, ..., k_{161}), K_7 = (k_{224}, ..., k_{193}), K_8 = (k_{256}, ..., k_{225}). Karena terdapat 32 putaran, maka penggunaan kunci dilakukan penjadwalan.

Tabel *S-Box* dibuat pemerintah Rusia dan digunakan oleh Federasi Bank Sentral Rusia (*Central Bank of Rusian Federation*). Tabel *S-Box* dapat dilihat pada Tabel 1.

Tabel 1. *S-Box*

S-Box	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	4	10	9	2	13	8	0	14	6	11	1	12	7	15	5	3
2	14	11	4	12	6	13	15	10	2	3	8	1	0	7	5	9
3	5	8	1	13	10	3	4	2	14	15	12	7	6	0	9	11
4	7	13	10	1	0	8	9	15	14	4	6	12	11	2	5	3
5	6	12	7	1	5	15	13	8	4	10	9	14	0	3	11	2
6	4	11	10	0	7	2	1	1	3	6	8	5	9	12	14	14
7	13	11	4	1	3	15	5	9	0	10	14	14	6	8	2	12
8	1	15	13	0	5	7	10	4	9	2	3	7	6	11	8	12

a. Proses Enkripsi

Proses enkripsi dari algoritma GOST untuk satu *round* (putaran) dijabarkan sebagai berikut:

i. 64 bit *plaintext* dibagi menjadi dua buah bagian, L_i dan R_i masing-masing bagian terdapat 32 bit.

ii. $(R_i + K_n)$ mod 2^{32} dimana K_n adalah kunci ke-*n*. Hasil penjumlahan dan modulo 2^{32} menghasilkan 32 bit.

iii. Hasil penjumlahan modulo 2^{32} dibagi menjadi 8 bagian, masing-masing bagian terdiri dari 4 bit. Setiap bagian dimasukkan ke dalam tabel *S-Box* yang berbeda, 4 bit pertama menjadi *input* dari *S-Box* pertama, 4 bit kedua menjadi *input S-Box* kedua dan seterusnya.

iv. Setelah mendapatkan hasil subtitusi dari *S-Box*, digabungkan kembali menjadi 32 bit, kemudian dilakukan operasi rotasi *left shift* sebanyak 11 bit.

v. Hasil dari rotasi *left shift* dilakukan operasi $\oplus$ dengan L_i, menghasilkan R_{i+1}.

vi. Sementara R_1 yang tidak dilakukan operasi apapun adalah hasil dari L_{i+1} .

vii. Untuk putaran selanjutnya dilakukan dengan cara yang sama dengan menggunakan R_{i+1} dan L_{i+1} sebagai biner enkripsi.

b. Proses Dekripsi

Proses dekripsi merupakan kebalikan dari proses enkripsi, urutan proses dekripsi tidak berubah dari proses enkripsi. Perbedaan yang terjadi terdapat pada penjadwalan kunci setiap putaran. Penjadwalan kunci setiap putaran untuk proses dekripsi sebagai berikut (Heryawan, 2010).

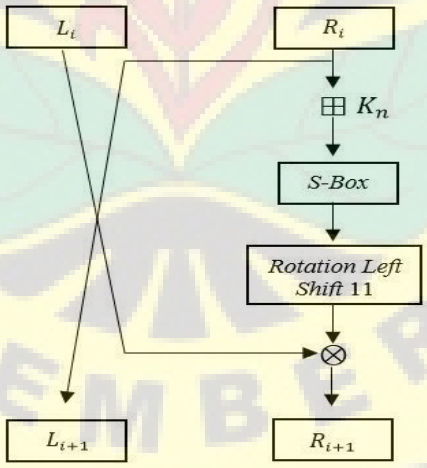
Putaran 1-8 : $K_1, K_2, K_3, \ldots, K_8$

Putaran 9-16 : $K_8, K_7, K_6, \ldots, K_1$

Putaran 17-24 : $K_8, K_7, K_6, \ldots, K_1$

Putaran 25-32 : $K_8, K_7, K_6, \ldots, K_1$

Proses enkripsi dan dekripsi algoritma GOST ditunjukkan pada Gambar 3.



Gambar 3. Proses Algoritma GOST

2.2 Least Significant Bit (LSB)

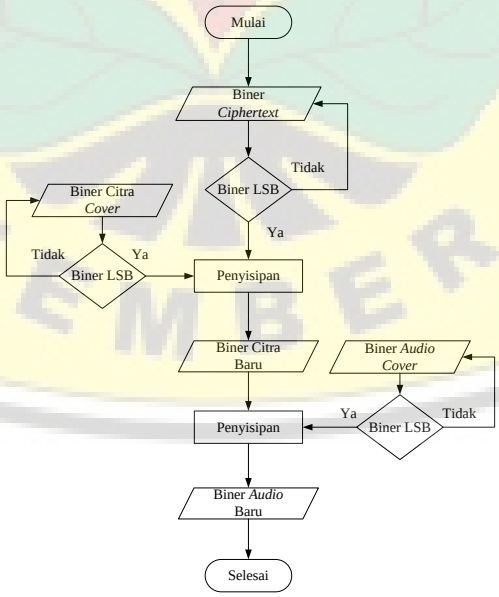
Metode *Least Significant Bit* (LSB) merupakan salah satu metode dalam steganografi yang digunakan untuk menyisipkan pesan atau data ke dalam format yang berbeda. Metode *Least Significant Bit* (LSB) berfungsi mengganti bit terakhir dari media *cover* dengan bit pesan atau data yang akan disisipkan. Pesan atau data yang disisipkan ke dalam media lain harus disesuaikan dengan ukuran

cover tersebut. *Least Significant Bit* (LSB) adalah bit yang mempunyai nilai terendah dalam barisan biner. Sedangkan bit yang memiliki nilai tertinggi disebut *Most Significant Bit* (MSB).

10001110

Angka satu yang bergaris bawah merupakan MSB sementara angka nol yang bergaris bawah adalah LSB.

Proses penyisipan pesan pada citra, dilakukan terhadap bit LSB *cover* dan bit LSB pesan. Pesan yang disisipkan cenderung mempunyi panjang yang dinamis, panjang pesan disesuaikan dengan *cover* penyisipan. Bila bit pesan lebih panjang dari bit *cover*, maka pesan tersebut tidak dapat disisipkan seluruhnya (Maryam, 2013). Pada citra *digital* penggantian bit-bit LSB berpengaruh terhadap warna. Penggantian bit LSB akan menyebabkan perubahan sebesar satu angka lebih tinggi ataupun satu angka lebih rendah dari nilai awal. Perubahan ini tidak menimbulkan perbedaan warna yang signifikan, karena perubahan hanya terjadi satu bit di atas atau di bawah bit awal. Flowchart algoritma LSB ditunjukkan pada Gambar 4.



Gambar 4. *Flowchart* Algoritma LSB

2.3 Citra

Citra adalah suatu gambaran (representasi), kemiripan atau imitasi dari suatu objek. Secara harfiah citra (*image*) adalah suatu gambar yang terletak pada

bidang dua dimensi. Citra terdiri dari dua sifat yaitu citra analog dan citra *digital*. Citra analog adalah citra yang bersifat *continue* seperti gambar pada monitor televisi, foto *sinar-X* dan lain sebagainya. Citra analog tidak dapat direpresentasikan oleh komputer sehingga tidak dapat langsung diolah oleh komputer, akan tetapi harus melalui proses konversi analog ke *digital* terlebih dahulu. Citra *digital* adalah Citra yang dapat diolah langsung oleh komputer secara numerik dan disimpan pada komputer sebagai angka untuk menunjukkan besar intensitas pada setiap *pixel* (Munir, 2004).

Citra *digital* berukuran N x M dinyatakan dengan matriks yang berukuran N baris dan M kolom sebagai berikut :

$$f(x,y) = \begin{bmatrix} f(1,1) & \cdots & f(1,M) \\ \vdots & \ddots & \vdots \\ f(N,1) & \cdots & f(N,M) \end{bmatrix} \quad (1)$$

Setiap elemen pada citra *digital* disebut dengan *pixel* (*picture element*). Jadi citra yang berukuran N x M mempunyai NM buah *pixel* (Munir, 2004).

Citra menjadi dua kelompok berdasarkan pergerakannya, yaitu citra diam dan Citra bergerak. Citra diam adalah citra tunggal yang tidak dapat bergerak. Citra bergerak adalah rangkaian citra diam yang ditampilkan secara beruntun sehingga memberikan kesan bergerak (Dulimarta, 1997). Menurut nilai *pixel*nya citra terbagi menjadi tiga jenis (Putra, 2010).

a. Citra Biner

Citra biner juga disebut dengan citra *B and W* (*Black* dan *White*). Kemungkinan warna yang dimiliki adalah hitam atau putih. Pada waktu penampilan citra, 0 adalah warna putih dan 1 adalah warna hitam, jadi citra biner mempunyai latar belakang putih dan objek berwarna hitam. Contoh citra biner dapat dilihat pada Gambar 5.

Gambar 5. Citra Biner

b. Citra *Grayscale*

Citra *grayscale* adalah suatu citra *digital* yang hanya mempunyai satu nilai kanal pada setiap *pixel*nya. Warna yang terdapat pada citra *grayscale* dapat berupa warna hitam, warna keabuan hingga warna putih. Citra *grayscale* mempunyai *range* 256. Intensitas 0 menyatakan hitam, intensitas 255 menyatakan putih dan nilai antara 0 sampai 255 menyatakan warna keabuan. Contoh gammbar *grayscale* dapat dilihat pada Gambar 6.

Gambar 6. Citra *Grayscale*

c. Citra Warna (RGB)

Citra warna atau citra *spectral* tediri dari tiga warna utama dengan kanal merah (*red*), hijau (*green*) dan biru (*blue*), warna lain dapat diperoleh dengan mencampurkan ketiga warna pokok tersebut. Seperti pada citra *grayscale* citra RGB memiliki *range* 0-255. Contoh citra RGB dapat dilihat pada Gambar 7.

Gambar 7. Citra citra RGB

2.4 Audio

Audio adalah sebuah fenomena yang dihasilkan oleh getaran suatu benda, berupa sinyal analog dengan amplitudo yang berubah secara terus menerus terhadap waktu dan frekuensi. Perbedaan tekanan terjadi selama benda bergetar dengan pola osilasi yang dinamakan gelombang. Perulangan gelombang yang sama pada interval tertentu disebut sebagai periode. Format *file audio* tanpa

kompresi yang paling sering ditemui adalah PCM (*Pulse Code Modulation*), biasanya tersimpan dalam format wav pada Windows dan sebagai aiff dalam Mac.Os. WAV adalah format *file* yang fleksibel untuk menyimpan kombinasi *audio* dengan bit *rates* maupun tanpa bit *rates*. WAV akan mengkoding semua suara, baik suara yang kompleks maupun tanpa suara, dengan jumlah bit yang sama setiap satuan waktunya (Santoso, 2014).

2.5 Analisis Keamanan

Analisis keamanan yang digunakan pada penelitian ini adalah analisis diferensial NPCR dan UACI. Langkah yang dilakukan dengan menghitung nilai dari *Number of Pixels Change Rate* (NPCR) dan *Unified Average Changing Intensity* (UACI).

$$NPCR = \frac{\sum_{i,j,k} D(i,j,k)}{W \times H \times L} \times 100\% \quad (2)$$

$$UACI = \frac{1}{W \times H \times L}\left[\sum_{i,j} \frac{|C(i,j,k) - C'(i,j,k)|}{255}\right] \times 100\% \quad (3)$$

di mana W, H dan L masing-masing adalah lebar, tinggi dan kanal citra dengan $D(i,j)$ ditentukan sebagai berikut:

$$D(i,j,k) = \begin{cases} 1, & C(i,j,k) = C'(i,j,k) \\ 0, & C(i,j,k) \neq C'(i,j,k) \end{cases} \quad (4)$$

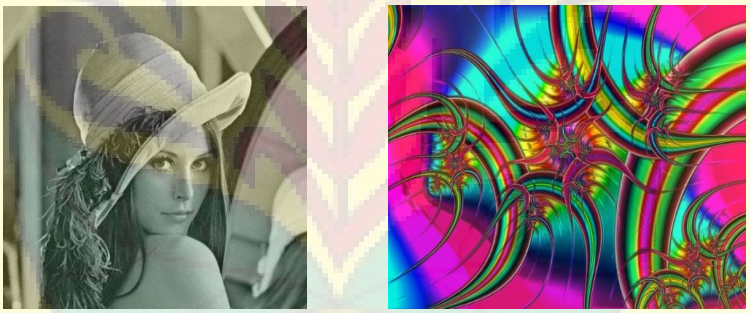
di mana $C(i,j,k)$ dan $C'(i,j,k)$ masing-masing adalah nilai dari derajat keabuan baris ke-i, kolom ke-j dan kanal ke-k dari citra C dan C'. Kriteria agar metode steganografi dikatakan kuat, apabila nilai NPCR dari media *cover* setelah penyembunyian pesan tidak mengalami perubahan 90%, sementara UACI digunakan untuk menghitung rata-rata perubahan intensitas setiap *pixel* dengan perubahan maksimal 30 (Yaimini *et.al.*, 2016).

3. HASIL DAN PEMBAHASAN

Penelitian yang dilakukan dalam tiga langkah, langkah pertama dilakukan proses enkripsi pesan menjadi *ciphertext* kemudian dilanjutkan dengan menyisipkan *ciphertext* pada gambar menghasilkan gambar baru. Gambar baru hasil penyembunyian *ciphertext* disembunyikan ke dalam *audio* menghasilkan

audio baru. Langkah kedua mengekstraksi *audio* baru hasil penyembunyian gambar baru dilanjutkan dengan ekstraksi gambar baru guna mendapatkan *ciphertext*. Dekripsi *ciphertext* untuk mendapatkan pesan awal. Langkah ketiga, dilakukan analisis keamanan guna mengetahui perubahan yang terjadi pada media *cover* setelah penyembunyian pesan.

Penelitian dilakukan menggunakan pesan 'Math2013' yang dienkripsi menggunakan algoritma GOST dengan kunci '#2n17j_2a1emaTiASa*U0e@M0t3/AtLk'. Gambar yang digunakan sebagai *cover* adalah gambar A dan B dengan ukuran masing-masing 256×256 dan 590×416 dengan format BMP dan PNG, ditunjukkan pada Gambar 8. *Cover audio* menggunakan musik *Beast and The Harlot* disingkat BTH dan *Faded* disingkat FDD dengan format WAV.



(A) (B)

Gambar 8. (A) Citra GRY dan (B) Citra RGB

3.1 Langkah Enkripsi dan Penyembunyian Ciphertext

a. Enkripsi pesan dengan algoritma GOST

Pesan awal dienkripsi menggunakan algoritma GOST sehingga menghasilkan *ciphertext* yang berbentuk tidak beraturan. Aturan yang digunakan untuk enkripsi sesuai dengan algoritma GOST.

b. Penyembunyian ciphertext pada gambar

Penyembunyian *ciphertext* hasil enkripsi dilakukan dengan menyembunyikan biner ciphertext pada gambar dengan metode LSB. Hasil dari penyembunyian ini adalah gambar baru.

c. Penyembunyian gambar baru pada *audio*

Gambar baru hasil penyembunyian *ciphertext* disembunyikan ke dalam *audio* menggunakan metode LSB dan menghaslkan *audio* baru.

3.2 Langkah Ekstraksi dan Dekripsi Ciphertext

a. Ekstraksi *audio*

Setelah mendapatkan *audio* baru hasil penyembunyian *ciphertext*, dilakukan ekstraksi agar mendapatkan gambar yang mengandung biner *ciphertext* menggunakan metode LSB.

b. Ekstraksi gambar

Gambar baru hasil ekstraksi *audio* yang mengandung biner *ciphertext* dilakukan ekstraksi menggunakan metode LSB supaya mendapatkan biner *ciphertext.*

c. Dekripsi *ciphertext*

Ciphertext yang didapatkan dari proses ekstraksi *audio* baru dilakukan untuk mendapatkan pesan aawal sebelum enkripsi.

3.3 Langkah Analisis Keamanan

a. Analisis NPCR dan UACI Pada Gambar

Dilakukan analisis NPCR dan UACI pada gambar yang telah disisipi *ciphertext* guna menentukan seberapa banyak perubahan yang terjadi pada gambar.

Tabel 2. Nilai NPCR dan UACI Pada Gambar

Gambar	Kanal	Format	Analisis (%)	
			NPCR	UACI
A	GRY	BMP	99,9771	5,8365
		PNG	99,9771	5,8365
	RGB	BMP	99,9736	4,7665
		PNG	99,9736	4,7665
B	GRY	BMP	99,9931	1,7662
		PNG	99,9931	1,7662
	RGB	BMP	99,9942	1,4892
		PNG	99,9942	1,4892

Dari Tabel 2 diketahui bahwa, nilai NPCR dan UACI bergantung pada ukuran dan jumlah kanal pada gambar. Semakin besar dan semakin banyak jumlah kanal

maka nilai NPCR semakin kecil dan nilai UACI semakin besar. Format gambar tidak memberikan pengaruh pada saat penyembunyian *ciphertext* pada gambar dikarenakan biner *ciphertext* disembunyikan dalam biner gambar yang didapat dari ukuran dan jumlah kanal gambar.

b. Analisis NPCR dan UACI Pada *Audio*

Analisis NPCR dan UACI juga dilakuakan pada *audio* yang telah disisipi gambar hasil penyembunyian *ciphertext.*

Tabel 3. Nilai NPCR dan UACI Pada *Audio*

Gambar	Kanal	Format	Audio	Analisis (%)	
				NPCR	UACI
A	GRY	BMP	BTH	94,7873	3,9221
			FDD	91,6363	8,4070
		PNG	BTH	94,7873	3,9221
			FDD	91,6363	8,4070
	RGB	BMP	BTH	94,7865	3,8838
			FDD	90,8459	6,2368
		PNG	BTH	94,7865	3,8838
			FDD	90,8459	6,2368
B	GRY	BMP	BTH	80,4563	14,9642
			FDD	68,6516	24,0127
		PNG	BTH	80,4563	14,9642
			FDD	68,6516	24,0127
	RGB	BMP	BTH	80,4531	15,3075
			FDD	68,6462	24,5638
		PNG	BTH	80,4531	15,3075
			FDD	68,6462	24,5638

Berdasarkan Tabel 3, nilai NPCR dan UACI dipengaruhi oleh ukuran dan jumlah kanal gambar serta panjang *audio*. Semakin besar ukuran gambar dan semakin pendek ukuran *audio* maka nilai NPCR semakin kecil, apabila ukuran gambar yang disembunyikan lebih kecil dan ukuran *audio* lebih lama maka nilai NPCR semakin besar dan nilai UACI semakin kecil. Format citra tidak berpengaruh pada NPCR dan UACI *audio* karena biner gambar yang disembunyikan pada biner *audio* berasal dari ukuran *pixel* gambar dan jumlah kanal.

4. KESIMPULAN

Hasil analisis NPCR dan UACI menunjukkan bahwa persentase tingkat keamanan ditentukan dari ukuran pesan yang disembunyikan dan ukuran media *cover*. Semakin besar ukuran media *cover* dan semakin kecil ukuran pesan yang disembunyikan, maka semakin besar pula persentase keamanan. Apabila ukuran pesan yang disembunyikan lebih besar dari ukuran media *cover*, maka semakin kecil persentase keamanan.

UCAPAN TERIMAKASIH

Riset ini dibiayai oleh Hibah Kelompok Riset (KeRis) 2018 dari Lembaga Penelitian dan Pengabdian Kepada Masyarakat Universitas Jember.

DAFTAR PUSTAKA

Dulimarta, H.S., *Diktat Kuliah Pengolahan Citra*, Jurusan Teknik Informatika Institut Teknologi Bandung, Bandung, 1997

Heryawan, I, P., *Aplikasi Keamanan Data Menggunakan Metoda Kriptografi GOST*, Jurnal TSI, **1**(2) (2010), 138-149

Iqbal, M., *The Understanding of GOST Cryptography Technique*, International Journal of Engineering Trends and Technology, **39**(3) (2016), 168-172.

Maryam, *Audio Steganografi Dengan Metode Least Significant Bit (LSB) Terhadap Pesan Terenkripsi Dengan Algoritma Towfish*, Jurusan Informatika FMIPA, Universitas Sebelas Maret, Surakarta, 2013.

Munir, R., *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Departemen Teknik Informatika, Institut Teknologi Bandung, Bandung, 2004.

Munir, R., *Diktat Kuliah IF5054 Kriptografi*. Departemen Teknik Informatika, Institut Teknologi Bandung, Bandung, 2006.

Putra, D., *Pengolahan Citra Digital*, Edisi I, ANDI, Yogyakarta, 2010.

Sah, A., *Aplikasi Pengamanan Data Menggunakan Metode Government Standard (GOST)*, Jurusan Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer AMIKOM, Yogyakarta, 2012.

Santoso, S., *Steganografi Audio (WAV) Menggunakan Metode LSB (Least*

Significant Bit), Jurnal Informatika, **9**(2) (2014), 214-224.

Wahyuningsih, Y., *Penyembunyian Pesan Terenkripsi Hill Chiper pada Audio File dengan Metode Least Significant Bit*, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, Jember, 2016.

Wirawan, S., *Implementasi Steganografi pada Berkas Audio WAV untuk Penyisipan Pesan Gambar Menggunakan Metode Low Bit Coding*, Fakultas Komputer Sains dan Teknologi Informasi Universitas Guna Dharma, Depok, 2011.

Yaimini, J. R, Bansal, G., Sharma, B., dan Kumar, *Image Encryption Schemes: A Complete Survey*, International Journal of Signal Processing, Image Processing and Pattern Recognition, **9**(7) (2016), 157-192.