STEGANOGRAFI PADA GAMBAR BERPOLA WARNA RGB BERDASARKAN FUNGSI ACAK

Dwinanto Cahyo – NIM : 13505025

Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if15025@students.if.itb.ac.id

Abstrak

Makalah ini akan membahas seputar teknik baru dalam steganografi menggunakan gambar yang bernama algoritma *triple-A* beserta garis besar implementasi dan pengujian yang dilakukan dibandingkan dengan sejumlah teknik yang telah ada. Algoritma tersebut diusulkan oleh tiga orang rekan dari Departemen Teknik Komputer, Sekolah Tinggi Ilmu dan Teknik Komputer *King Fahd University of Petroleum and Minerals* (KFUPM) Saudi Arabia, yaitu Adnan Gutub, Ayed Al-Qahtani dan Abdulaziz Tabakh.

Algoritma tersebut menggunakan konsep yang sama dengan teknik LSB (*Least Significant Bits*), di mana pesan rahasia disimpan di dalam bit yang paling tidak signifikan dari titik warna pada gambar, dengan memanfaatkan lebih banyak proses pengacakan dalam pemilihan jumlah dari bit yang digunakan dan saluran warna yang dilibatkan. Dengan demikian, diharapkan dapat meningkatkan aspek keamanan dari sistem serta menambah kapasitas pesan yang dapat disisipkan dalam suatu gambar.

Teknik ini dapat diterapkan dengan menggunakan gambar bertipe warna RGB (*Red-Green-Blue*), di mana setiap titik warna direpresentasikan dengan tiga byte data yang masing-masing menyatakan intensitas warna merah, hijau dan biru dalam titik warna tersebut.

***Kata kunci**:* steganografi, pengacakan, gambar RGB, algoritma *triple-A*, keamanan komputer.

1. Pendahuluan

Dalam bahasa Yunani steganografi berarti tulisan rahasia. Dalam ilmu penyandian, steganografi adalah seni penyembunyian informasi ke dalam media penyimpan lain sedemikian rupa sehingga tak ada orang lain yang mampu mengetahui dan membaca pesan rahasia yang disembunyikan selain pihak penerima. Steganografi adalah seni yang berkembang sejak dahulu kala dan digunakan sejak masa kejayaan Romawi untuk menuliskan pesan dalam bentuk ukiran pada meja kayu yang kemudian dilapisi dengan lilin dan penggunaan manusia sebagai pengirim pesan dengan menuliskan pesannya pada kulit kepala setelah dibotaki dan menunggu hingga ditumbuhi rambut lagi baru dikirimkan ke pihak penerima pesan. Teknik lain yang digunakan adalah dengan memanfaatkan tinta transparan, rangkaian titik-titik dan pengacakan susunan karakter.

Dengan teknologi digital yang ada saat ini, banyak aplikasi dari steganografi digital dalam kehidupan sehari-hari yang berkembang hingga saat ini, diantaranya adalah *watermarking* untuk melindungi hak cipta dan menjaga kerahasiaa nilai suatu dokumen dari kemungkinan sabotase dan pencurian

Teknik steganografi menggunakan gambar membutuhkan sebuah gambar sebagai media penyembunyi data yang disebut dengan *cover image*. Steganografi pun banyak dilakukan dengan memanfaatkan media digital, salah satu media yang sering digunakan adalah gambar. Gambar tersebut disimpan dalam dokumen digital pada sistem komputer sebagai kumpulan titik, di mana setiap titik memiliki tiga komponen warna, yaitu merah, hijau dan biru (*red*, *green*, *blue*: RGB). Setiap titik direpresentasikan oleh data seukuran tiga *byte* untuk menggambarkan porsi dari masing-masing komponen warna tersebut. Beberapa teknik yang digunakan dalam steganografi gambar diantaranya adalah LSB (*Least Significant Bit*), SCC dan *Pixel Indicator*.

Dalam teknik LSB (*Least Significant Bits*), bit yang paling tidak signifikan pada sebuah warna tertentu atau untuk semua saluran warna diganti dengan sebuah bit dari pesan yang ingin disisipkan. Meskipun LSB merupakan teknik yang sederhana, namun kemungkinan untuk mendeteksi keberadaan pesan rahasia yang disembunyikan pun tergolong cukup tinggi.

Teknik SCC merupakan pengembangan lebih lanjut, dengan menggunakan saluran warna sebagai tempat penyimpan pesan rahasia. Saluran warna tersebut dirotasikan secara berkala untuk setiap bit berdasarkan suatu pola yang telah ditentukan. Contohnya, bit pertama dari suatu pesan rahasia disimpan dalam LSB dari saluran warna merah, bit kedua dari pesan disimpan dalam saluran warna hijau, bit ketiga dari pesan disimpan dalam saluran warna biru dan seterusnya. Teknik ini lebih aman jika dibandingkan dengan LSB, namun masih menyimpan potensi ancaman deteksi pola rotasi yang akan membongkar penyimpanan pesan rahasia. Selain itu, teknik ini juga memiliki kapasitas penyimpanan yang lebih sedikit dibandingkan dengan LSB.

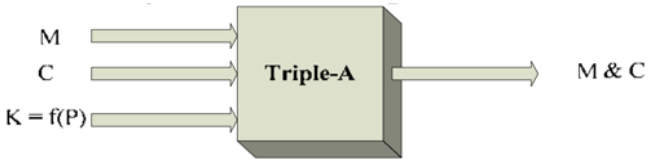
Teknik *Pixel Indicator* adalah teknik steganografi berbasis gambar yang memanfaatkan dua bit yang paling tidak signifikan dari suatu saluran warna tertentu untuk mengindikasikan keberadaan data rahasia pada dua bit yang paling tidak signifikan dari dua saluran warna lain yang ditentukan berdasarkan aturan yang baku seperti yang tercantum dalam [1]. Sebuah ide baru dari steganografi menggunakan sistem RGB diajukan dalam [2], teknik tersebut didasarkan pada intensitas warna yang berada di luar fokus bahasan makalah ini.

Meskipun teknik *Pixel Indicator* menambahkan sejumlah proses pengacakan nilai untuk mempersulit pendeteksian keberadaan data rahasia, namun kapasitas penyimpanan pesan rahasia menggunakan teknik tersebut sangat bervariasi, tergantung pada nilai dari saluran warna yang berperan sebagai indikator, sehingga kapasistas sebenanya tidak dapat diperkirakan secara pasti. Teknik yang digunakan dalam algoritma *triple-A* adalah dengan menambahkan lebih banyak proses pengacakan dalam langkah pemilihan titik warna tempat penyimpan data rahasia, yang berdampak pada jumlah bit dan saluran warna yang digunakan untuk menyimpan data rahasia tersebut.

Dalam bagian berikutnya, akan dibahas algoritma yang disebut dengan *triple-A*, beserta penjelasan singkat mengenai implementasi dan hasil percobaan pada bagian tiga. Bagian empat akan membandingkan algoritma tersebut dengan sejumlah algoritma yang telah ada saat ini. Bagian lima berisi kesimpulan dan diakhiri daftar pustaka.

2. Algoritma *Triple-A*

Mekanisme proses algoritma *triple-A* adalah dengan menggunakan masukan berupa pesan yang ingin disisipkan (*message*/M), gambar yang akan digunakan untuk menyisipkan pesan (*carrier image*/C) dan kunci yang dibangkitkan menggunakan kata kunci (*key*/K) dan menghasilkan pesan (M) yang telah tersembunyi di dalam gambar penyimpan (C). Gambar 1 menunjukkan ilustrasi mekanisme algoritma *triple-A*.



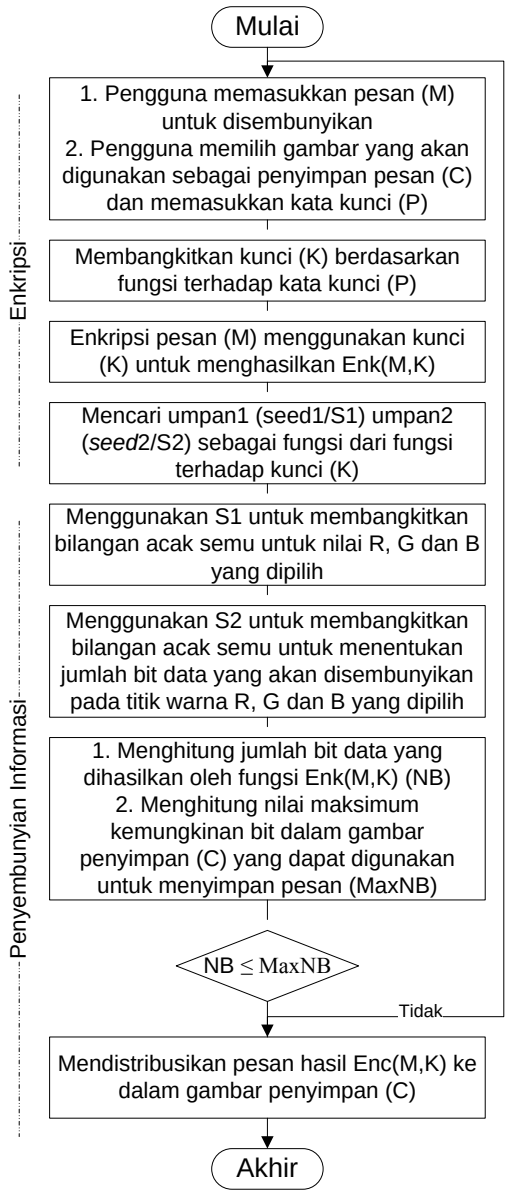
Gambar 1 Masukan dan Keluaran Algoritma *Triple-A*

Secara garis besar, algoritma *triple-A* terdiri atas dua bagian, yaitu proses enkripsi dan penyembunyian informasi, seperti yang ditunjukkan dalam Gambar 2.

Fokus pada bagian pertama adalah proses enkripsi pesan (M) dengan menggunakan algoritma AES (*Advanced Encryption Standard*) yang akan menghasilkan Enk(M,K). Pada implementasi yang dilakukan, kunci K dapat dibangkitkan dari kumpulan kata kunci pengguna yang masing-masing memiliki kunci tertentu dengan menggunakan operasi XOR sederhana. Dengan demikian, aspek keamanan akan dapat ditingkatkan, khususnya pada kondisi di mana diinginkan agar pesan rahasia yang disembunyikan hanya jika semua pengguna memasukkan kata kunci mereka masing-masing.

Dalam bagian kedua, digunakan gambar berbasis warna RGB sebagai penyimpan informasi, dengan mengambil keuntungan dari aspek gambar bertipe BMP (*bitmap*), di mana setiap titik warna bersifat tidak saling tergantung dari

pengaturan ulang gambar BMP secara keseluruhan.



Gambar 2 *Flow Chart* Algoritma *Triple-A*

Hasil dari Enk(M,K) tersembunyi, seperti dalam langkah pada Gambar 2, dengan memanfaatkan pembangkit bilangan acak semu (*pseudo-random number generator*/PRNG). PRNG digunakan untuk menghasilkan dua bilangan acak baru dalam setiap pengulangan. Umpan yang digunakan dalam setiap proses PRNG, disebut dengan umpan1 (*seed*1/S1) dan umpan2 (*seed*2/S2), dihasilkan sebagai sebuah fungsi dari kunci K. S1 dibatasi untuk membangkitkan bilangan pada rentang [0, 6], sedangkan S2 dibatasi pada rentang [1, 3]. Bilangan acak S1 digunakan untuk menentukan komponen warna dari gambar berbasis RGB yang akan digunakan untuk menyembunyikan data yang telah dienkripsi menggunakan Enc(M,K).

Tabel 1 menunjukkan bagaimana bilangan acak S1 menentukan komponen RGB yang akan digunakan. Di sisi lain, bilangan acak S2 menentukan jumlah komponen bit paling tidak signifikan yang akan digunakan untuk menyembunyikan data rahasia. Dengan cara serupa, Tabel 2 menunjukkan bagaimana bilangan acak S2 menentukan jumlah komponen bit. Gambar BMP dalam sistem komputer direpresentasikan oleh larik (*array*) dua dimensi dengan posisi X dan Y. Posisi X dan Y dari titik warna yang mengandung data rahasia didistribusikan di dalam gambar berdasarkan hasil penghitungan ukuran data rahasia Enk(M,K) dan gambar penyimpan (C).

Tabel 1 Penggunaan Bilangan Acak S1

	Bilangan Acak	Makna dalam algoritma
PRNG pertama	0	Menggunakan R
	1	Menggunakan G
	2	Menggunakan B
	3	Menggunakan RG
	4	Menggunakan RB
	5	Menggunakan GB
	6	Menggunakan RGB

Tabel 1 menunjukkan bahwa jumlah bit maksimal yang dapat digunakan dari sebuah komponen adalah 3 bit. Sedangkan tabel 2 menunjukkan bahwa jumlah bit maksimal yang digunakan untuk menentukan komponen bit yang digunakan adalah sebanyak 2 bit. Kedua hal tersebut menunjukkan bahwa algoritma yang digunakan dapat menambahkan hingga ±7 nilai ke dalam komponen warna pada titik warna yang bersangkutan.

Tabel 2 Penggunaan Bilangan Acak S2

	Bilangan Acak	Makna dalam algoritma
PRNG kedua	1	Menggunakan 1 bit dari komponen
	2	Menggunakan 2 bit dari komponen
	3	Menggunakan 3 bit dari komponen

Selain itu, dengan mengkombinasikan data dari kedua tabel tersebut, kita dapat melihat bahwa jumlah bit minimal yang dapat digunakan dalam

setiap titik warna adalah 1 jika kita hanya menggunakan satu bit dari sebuah komponen dari warna gambar berbasis RGB yang dipilih. Jumlah maksimal bit yang dapat digunakan adalah 9 jika kita menggunakan ketiga komponen dengan tiga bit.

Tabel 3 Contoh Steganografi Gambar dengan Algoritma *Triple-A*

BA1	BA2	CMP sebelum penyisipan			CMP setelah penyisipan			CMP sebelum penyisipan			CMP setelah penyisipan		
5	3	32	93	160	32	95	161	0010_0000	0101_1101	1010_0000	0010_0000	0101_1111	1010_0001
1	1	31	93	154	31	92	154	0011_1111	0101_1101	1001_1010	0011_1111	0101_1100	1001_1010
1	1	33	94	161	33	94	161	0010_0001	0101_1110	1010_0001	0010_0001	0101_1110	1010_0001
0	1	34	93	154	35	93	154	0010_0010	0101_1101	1001_1010	0010_0011	0101_1101	1001_1010
2	2	37	95	154	37	95	155	0010_0101	0101_1111	1001_1010	0010_0101	0101_1111	1001_1011
6	3	39	97	161	35	102	166	0010_0111	0110_0001	1010_0001	0010_0011	0110_0110	1010_0110
3	2	40	97	164	40	99	164	0010_1000	0110_0001	1010_0100	0010_1000	0110_0011	1010_0100
1	2	37	96	162	37	99	162	0010_0101	0110_0000	1010_0010	0010_0101	0110_0011	1010_0010
2	1	39	99	162	39	99	162	0010_0111	0110_0011	1010_0010	0010_0111	0110_0011	1010_0010
3	2	41	102	167	43	102	167	0010_1001	0110_0110	1010_0111	0010_1011	0110_0110	1010_0111
5	3	45	104	170	45	104	169	0010_1101	0110_1000	1010_1010	0010_1101	0110_1000	1010_1001
1	1	51	110	178	51	111	178	0011_0011	0011_1110	1011_0010	0011_0011	0110_1111	1011_0010

Gambar 3 Gambar Sebelum Penyisipan

Gambar 4 Gambar Setelah Penyisipan

Tabel 3 menunjukkan contoh penyembunyian lima *byte* berurutan dari hasil Enk(M,K) ke dalam sebuah gambar penyimpan (Gambar 3) dan menghasilkan Gambar 4. *Byte* yang diproses adalah 0x15, 0xF9, 0xCD, 0x5B, 0x09 = 0000_1111, 1001_1111, 1100_1101, 0101_1011, 0000_1001.

3. Garis Besar Implementasi

Algoritma *triple-A* mampu meningkatkan perbandingan kapasitas dan tingkat keamanan proses penyembunyian informasi yang dilakukan. Secara teori, jumlah bit rata-rata yang digunakan untuk setiap titik warna adalah 3.428, dengan jumlah maksimal bit yang digunakan dalam SCC adalah 3 dan untuk LSB adalah 1. Hal tersebut menunjukkan bahwa kapasitas dari algoritma *triple-A* lebih tinggi dibanding teknik sebelumnya.

Dengan menggunakan algoritma *triple-A*, perbandingan antara jumlah bit yang digunakan di dalam sebuah titik warna untuk menyembunyikan suatu bagian dari pesan rahasia dengan jumlah bit yang dikandung titik warna tersebut, atau yang disebut dengan faktor kapasitas, berada dalam rentang 1/24 hingga 9/24. Hal tersebut diperoleh ketika digunakan maksimal 3 bit seperti yang tercantum dalam tabel, namun jika kita menggunakan 4 atau bahkan 5 bit, faktor rasio yang diperoleh dapat ditingkatkan hingga mencapai 15/24 yang berarti lebih dari setengah jumlah bit pada titik warna yang bersangkutan. Akan tetapi, sisi buruk yang muncul adalah adanya penambahan sisi gambar yang tampak kabur dibanding gambar awal seiring penambahan jumlah bit yang digunakan.

Penjelasan yang telah dikemukakan hingga saat ini adalah berdasarkan sudut pandang titik warna. Jika kita melihat keseluruhan gambar, karena kita menggunakan PRNG, kita harus mencari nilai rata-rata dari jumlah bit yang digunakan untuk menyembunyikan data rahasia dalam gambar. Nilai perbandingan kapasitas dapat diperoleh dari jumlah bit yang mungkin digunakan dalam setiap kemungkinan dibagi dengan hasil kali antara total jumlah kemungkinan dengan dua puluh empat.

Karena terdapat sejumlah 21 kasus yang dideskripsikan sebagai berikut:

- Kasus yang menggunakan satu komponen; di mana terdapat tiga cara untuk

menentukan bit yang akan digunakan dan 3 cara untuk menentukan komponen R, G atau B; sehingga total terdapat 9 kasus.

- Kasus yang menggunakan dua komponen; di mana terdapat 3 cara untuk menentukan bit yang akan digunakan dan 3 cara untuk menentukan komponen RG, RB atau GB; sehingga total terdapat 9 kasus.
- Kasus yang menggunakan tiga komponen; di mana terdapat 3 cara untuk menentukan bit yang akan digunakan dan hanya ada satu pilihan komponen, yaitu RGB; sehingga total terdapat 2 kasus.

Rata-rata perbandingan kapasitas yang diperoleh berkisar antara 1/7 atau 14% dari ukuran keseluruhan gambar penyimpan yang digunakan. Hal ini masih lebih baik jika dibandingkan dengan algoritma SCC yang menghasilkan perbandingan kapasitas sejumlah 1/24 atau 4%.

Tingkat keamanan dapat dibagi menjadi dua lapisan, yaitu bagian penyimpanan dan bagian enkripsi data. Lapisan pertama merupakan proses penyimpanan data seperti yang ditunjukkan pada Gambar 2, dalam proses tersebut data rahasia disebar ke bagian-bagian di seluruh gambar, sehingga mencari informasi yang tersimpan pada gambar tanpa sama sekali mengetahui tentang umpan yang digunakan adalah hampir mustahil untuk dilakukan.

Ditambah lagi, lapisan kedua menggunakan algoritma AES untuk melakukan enkripsi terhadap data. Sehingga, bahkan jika pihak penyerang mengetahui bagaimana langkah untuk mencari informasi yang disimpan, masih dibutuhkan proses dekripsi untuk mengetahui maknanya.

Algoritma *triple-A* memiliki ukuran pesan yang sama-sama tidak dapat diprediksi, layaknya dalam teknik *Pixel Indicator*m akan tetapi nilai perbandingan kapasitas yang dimiliki masih lebih baik. Selain itu, hal yang tidak dapat diprediksi dalam teknik *Pixel Indicator* adalah fungsi terhadap gambar penyimpan (C) yang biasanya berukuran besar, sedangkan dalam algoritma *triple-A*, hal tersebut hanya merupakan fungsi atas kunci (K) yang tentunya berukuran jauh lebih kecil.

4. Pengujian dan Perbandingan

Algoritma *triple-A* telah diterapkan menggunakan paket-paket perangkat lunak yang dikembangkan dengan C#. Alat untuk proses enkripsi data sebelum disembunyikan adalah skema enkripsi AES. Gambar penyimpan informasi yang dihasilkan kemudian diujicoba dan dibandinkan dengan gambar asli menggunakan histogram yang dibangkitkan melalui MATLAB untuk memeriksa tingkat kekaburan atau distorsi yang disebabkan oleh algoritma *triple-A*.

Hasil yang diperoleh selanjutnya dibandingkan dengan gambar penyimpan informasi lain yang dihasilkan menggunakan algoritma SCC, dengan tingkat distorsi dan masalah kapasitas penyimpanan sebagai fokus utama.

Gambar 3 menunjukkan gambar penyimpan sebelum disisipkan informasi menggunakan algoritma SCC dan *triple-A*. Perbedaan antara gambar sebelum dan sesudah disisipkan informasi tidak dapat dilihat secara kasat mata, akan tetapi histogram dari gambar tersebut yang ditunjukkan oleh Gambar 5, 6 dan 7 menunjukkan perbedaan kecil dalam nilai komponen R, G dan B yang dikandung kedua gambar tersebut.

Contoh pada tabel 3 diambil dari sejumlah titik warna pertama dari Gambar 3 dan 4. Tabel tersebut menunjukkan bahwa nilai perbandingan kapasitas berkisar 13,2% dengan perbedaan 5% jika dibandingkan dengan rata-rata yang dihitung.

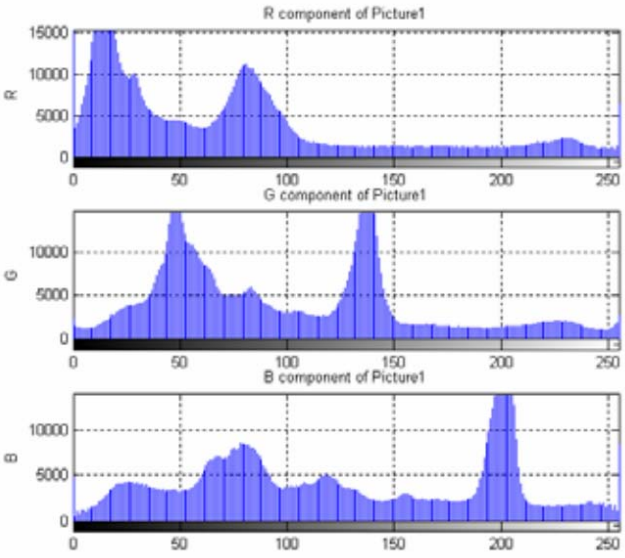
Tabel 4 Perbandingan *Triple-A* dengan SCC

Ukuran Pesan (*byte*)	Penyembunyian dengan SCC		Penyembunyian dengan *triple-A*	
	Titik warna yang digunakan	kapasitas	Titik warna yang digunakan	kapasitas
28 KB	27984	4,16%	7169	16,27%

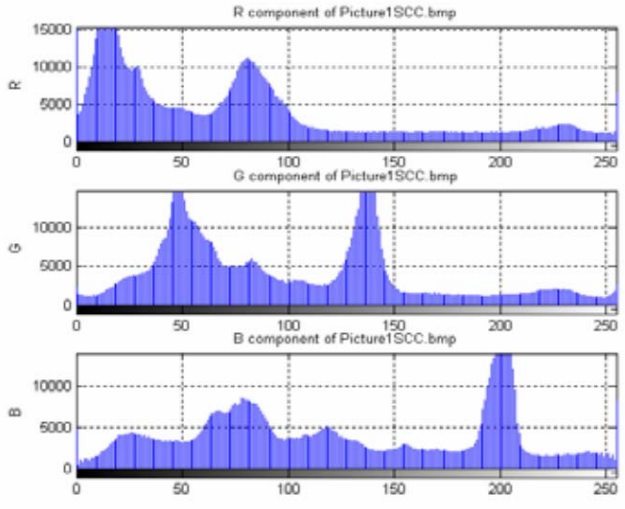
Tabel 4 menunjukkan perbandingan antara algoritma *triple-A* dan SCC. Hasil tersebut diperoleh dengan menggunakan gambar penyimpan yang berbeda dan penghitungan rata-rata jumlah titik warna yang digunakan dalam proses penyisipan informasi. Terlihat bahwa algoritma *triple-A* meningkatkan perbandingan kapasitas dengan faktor sekitar 4. Tabel tersebut menunjukkan bahwa perbandingan kapasitas algoritma *triple-A* berkisar pada angka 16,27% hingga 20% dibandingkan dengan rata-rata hitung.

Algoritma SCC memiliki perbandingan kapasita kecil yang bernilai relatif tetap, yaitu sekita 1/24, sedangkan algoritma *triple-A* menghasilka peningkatan yang cukup besar dalar perbandingan kapasitas tanpa mempengaruh gambar dengan aspek distorsi.

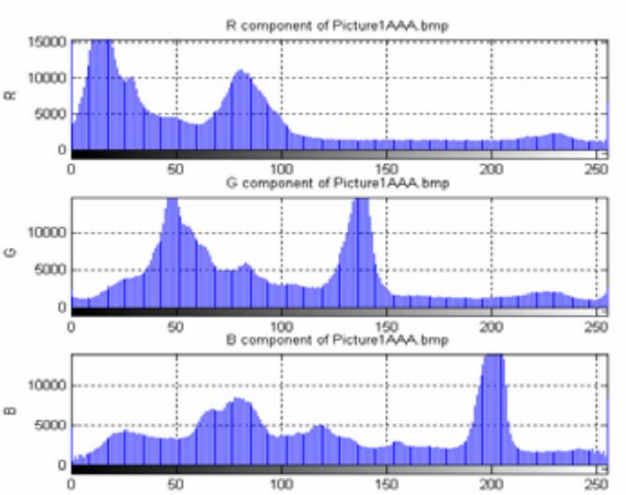
Sebagai salah satu prosedur stego-analisis, dat yang disembunyikan dengan SCC dapa diperoleh kembali dengan mudah setela mengetahui bahwa terdapat kemungkinan bahw terdapat data tersembunyi dalam sebuah gambar Akan tetapi, dalam kasus *triple-A*, hal tersebu akan jauh lebih sulit untuk dilakukan.



Gambar 5 Gambar Penyimpan Asli



Gambar 6 Gambar Setelah Disisipkan Informasi Menggunakan Algoritma SCC



Gambar 7 Gambar Setelah Disisipkan Informasi Menggunakan Algoritma *Triple-A*

Untuk memastikan perbedaan kecil pada gambar 3 dan 4 dengan gambar 5, 6 dan 7, digunakan gambar penyimpan lain yang berwarna merah untuk menyembunyikan data berukuran kecil. Gambar merah tersebut tidak menujukkan perbedaan antara sebelum dengan sesudah penyisipan informasi. Akan tetapi, histogram algoritma SCC menunjukkan perbedaan yang lebih jelas dibandingkan perbandingan yang ditunjukkan pada kedua kelompok gambar sebelumnya. Hal yang perlu diperhatikan adalah bahwa gambar merah tersebut tidak memiliki komponen G dan B, sehingga histogram komponen kedua warna tersebut adalah 0, sedangkan warna merah bernilai 255. Teknik SCC menggunakan hanya bit yang paling tidak signifikan dan menyebabkan cekungan yang tampak tidak lazim di histogram pada nilai 254.

Di sisi lain, algoritma *triple-A* menggunakan 3 bit yang menjadikan cekungan yang ditimbulkan tidak tampak mencolok dengan perbedaan ±7.

5. Kesimpulan

Algoritma *triple-A* diperkenalkan sebagai metode baru untuk menyembunyikan data digital di dalam medium berbasis gambar. Algoritma tersebut menambahkan proses pengacakan dengan menggunakan dua umpan berbeda yang dibangkitkan dari sebuah kunci yang dimasukkan oleh pengguna untuk memilih komponen yang digunakan untuk menyembunyikan bit data dan jumlah bit yang digunakan di dalam komponen warna RGB pada

gambar. Proses pengacakan tersebut meningkatkan aspek keamanan, khususnya jika sebuah metode enkripsi aktif digunakan, seperti AES. Nilai perbandingan kapasitas meningkat hingga melebihi metode SCC dan *Pixel Indicator*. *Triple-A* memiliki perbandingan kapasitas sebesar 14% dan dapat ditingkatkan seiring dengan jumlah bit yang digunakan dalam komponen yang bersangkutan.

Sebagai catatan terakhir, dapat dinyatakan bahwa algoritma SCC adalah kasus spesial dari algoritma *triple-A* jika jumlah bit yang digunakan adalah tetap, yaitu 1, dan nilai umpan2 (*seed*2/S2) dibatasi pada rentang [0, 2] dengan dampak sirkular.

REFERENSI

[1] Adnan Gutub, Mahmoud Ankeer, Muhammad Abu-Ghalioun, Abdulrahman Shaheen, and Aleem Alvi, "*Pixel Indicator high capacity Technique for RGB image Based Steganography*", WoSPA 2008 – 5th IEEE International Workshop on Signal Processing and its Applications, University of Sharjah, Sharjah, U.A.E. 18-20 Maret 2008.

[2] Mohammad Tanvir Parvez and Adnan Gutub, "*RGB Intensity Based Variable-Bits Image Steganography*", APSCC 2008 – Proceedings of 3rd IEEE Asia-Pacific Services Computing Conference, Yilan, Taiwan, 9-12 Desember 2008.

[3] *Gutub, Adnan and Al-Qahtani, Ayed and Tabakh, Abdulaziz (2009).* Triple-A: *Secure RGB Image Steganography Based on Randomization*. The 7th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-2009), 10-13 Mei 2009, Rabat, Morocco.