

PENYISIPAN PESAN TEKS TERKOMPRESI HUFFMAN KE DALAM CITRA RGB DENGAN MENGGUNAKAN METODE BINARY AS DECIMAL CODE

Ansar Rizal¹⁾, Rihartanto²⁾

^{1,2)}Jurusan Teknologi Informatika – Politeknik Negeri Samarinda

^{1,2)}Jl. Cipto Mangunkusumo, Kampus Gunung Lipan, Samarinda Seberang, 75131

email : anrisal@yahoo.com¹⁾, rihart.c@gmail.com²⁾

Abstrak

Steganografi adalah teknik menyembunyikan pesan rahasia ke dalam cover media sedemikian rupa sehingga keberadaannya tidak disadari oleh pihak lain. Terdapat berbagai macam cara untuk memanipulasi pesan rahasia sebelum disisipkan ke dalam *cover media*. Di dalam penelitian ini, pesan rahasia yang tersimpan di dalam *file.txt* terlebih dahulu dikompresi dengan menggunakan Algoritma Huffman. Kode bit pesan terkompresi kemudian disisipkan ke dalam citra RGB sebagai *cover media* pada komponen R dengan menggunakan metode *Binary as Decimal Code* (BDC). Kamus Huffman diubah ke dalam 3 (tiga) bit biner untuk kemudian disisipkan ke dalam komponen B dengan menggunakan metode BDC. Sedangkan panjang bit pesan dan bit Kamus Huffman disisipkan ke dalam komponen G. Pengujian metode dilakukan dengan menggunakan 3 (tiga) *file.txt* yang berbeda yang disisipkan ke dalam 3 (tiga) file citra JPG dengan ukuran yang berbeda. Hasil pengukuran performansi menunjukkan bahwa metode yang diusulkan menghasilkan *stego-image* dengan kualitas citra yang sangat baik. Hal ini dibuktikan dengan RPSNR sebesar 55.68 dB (di atas 50 dB).

Kata kunci : *cover-image*, citra RGB, BDC, Huffman, PSNR

1. PENDAHULUAN

Terdapat berbagai macam cara melindungi suatu informasi, di antaranya dengan kriptografi dan steganografi. Kriptografi mengubah pesan menjadi tidak bermakna agar pihak lain tidak mengetahui isi pesan di dalamnya. Sedangkan steganografi menyamarkan suatu pesan sedemikian rupa sehingga keberadaannya tidak disadari oleh pihak lain.

Steganografi dapat digunakan pada berbagai macam bentuk data seperti teks, citra, audio, dan video. Pada prinsipnya steganografi memerlukan dua media yaitu *cover-media* sebagai tempat menyembunyikan pesan rahasia dan *embedded media* yaitu pesan yang disembunyikan. Hasil proses penyisipan pesan rahasia adalah *stego-media*.

Terdapat berbagai macam cara untuk memanipulasi pesan sebelum disisipkan ke dalam *cover-media*. Pesan dapat dienkripsi terlebih dahulu agar pesan menjadi tidak bermakna atau pesan dikompresi terlebih dahulu agar ukurannya menjadi semakin kecil. Untuk *cover-media* dapat pula dimanipulasi sedemikian rupa sebelum disisipkan pesan rahasia. Sedangkan untuk menyisipkan pesan rahasia ke dalam *cover-media* juga terdapat berbagai macam metode.

2. RUANG LINGKUP PENELITIAN

Dalam penelitian ini permasalahan mencakup :

1. Pesan rahasia yang disimpan ke dalam *file.txt*.
2. *Cover-media* berupa citra RGB.
3. Sebelum disisipkan, pesan rahasia dikompresi menggunakan Algoritma Huffman.
4. Penyisipan pesan rahasia terkompresi dilakukan dengan metode BDC (*Binary as Decimal Code*) ke dalam komponen R dari citra RGB.

5. Kamus Huffman disisipkan ke dalam komponen B dari citra RGB dengan metode BDC.
6. Panjang bit pesan dan kamus Huffman disimpan ke dalam komponen G dari citra RGB yang memiliki nilai piksel di bawah $[255 - \text{length}(\text{pesan} + \text{kamus huffman})]$.

3. BAHAN DAN METODE

3.1. Kompresi Data

Kompresi data artinya mengecilkan atau memampatkan ukuran bit data. Kompresi data bertujuan untuk efisiensi memori penyimpanan data. Terdapat dua jenis kompresi data yaitu :

1. *lossles compression*, yaitu data terkompresi tidak akan berubah karena akan digunakan sebagaimana data sebelum dikompresi. Jenis kompresi ini biasa digunakan untuk penyimpanan *database records*, *spreadsheets*, dan *word processing file*.
2. *lossy compression*, yaitu data terkompresi akan digunakan sebagai data baru dimana sebagian data yang tidak penting akan dibuang. Jenis kompresi ini biasa digunakan untuk data citra, video, dan audio.
(Ramya, 2016).

Pada prinsipnya kompresi terdiri dari dua bagian yaitu bagaimana memodelkan data dan bagaimana mentransformasikan model data ke dalam bentuk kode-kode hasil kompresi.

3.2. Kode Huffman

Kode Huffman adalah teknik kompresi data *lossless* yang secara statistik mengurangi panjang kode rata-rata untuk mewakili simbol alfabet. Kode Huffman bekerja

berdasarkan pohon biner. Representasi pohon biner diakses dari daun terlebih dahulu mulai dari daun kiri, daun kanan kemudian akar. Pohon biner yang akar-akarnya diberi nama 0 dan 1 disebut label. Daun kiri diberi label 0 dan daun kanan diberi label 1.

Untuk memperoleh kode Huffman terlebih dahulu perlu dihitung frekuensi kemunculan tiap simbol dalam teks. Pohon Huffman dibentuk dengan cara sebagai berikut :

1. Hitung frekuensi kemunculan simbol dalam teks.
2. Hitung probabilitas kemunculan simbol.
3. Urutkan simbol secara *ascending* berdasarkan probabilitas kemunculannya.
4. Gabungkan dua simbol dengan peluang paling kecil, kemudian jumlahkan probabilitas kemunculannya.
5. Ulangi no.2) hingga tersisa satu pohon biner.
6. Beri label 0 untuk sisi kiri dan 1 untuk sisi kanan.
7. Penelusuran pohon biner dari akar ke daun adalah kode Huffman.

(Senthilkumar, 2016).

Sebagai contoh pesan teks yang disimbolkan dalam ASCII dan kode biner seperti yang ditunjukkan dalam Tabel 1.

Tabel 1. Contoh Pemodelan Simbol Pesan

Char	ASCII	biner
A	65	01000001
L	76	01001100
I	73	01001001
B	66	01000010
A	65	01000001
B	66	01000010
A	65	01000001

Hasil perhitungan probabilitas kemunculan simbol ditunjukkan dalam Tabel 2.

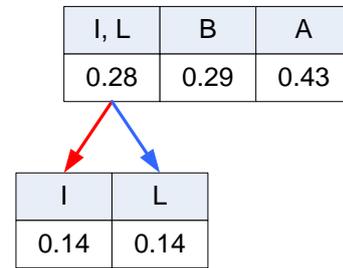
Tabel 2. Hasil perhitungan probabilitas kemunculan simbol

Char	frekuensi	probabilitas
A	3	0.43
B	2	0.29
I	1	0.14
L	1	0.14
Jumlah	7	1.00

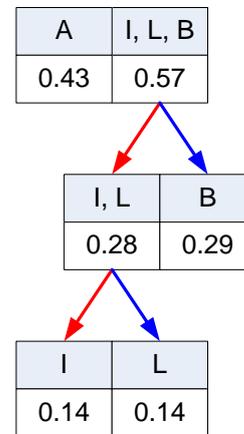
Simbol kemudian diurutkan secara ascending menjadi seperti berikut :

I	L	B	A
0.14	0.14	0.29	0.43

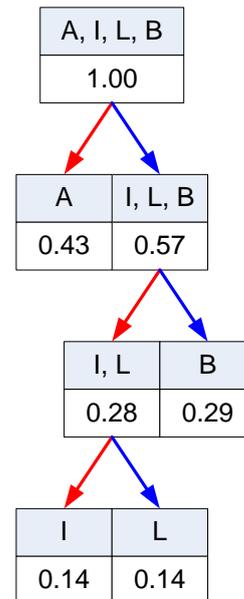
Langkah berikutnya adalah menggabungkan dua simbol dengan probabilitas kemunculan terkecil kemudian diurutkan kembali, menjadi seperti berikut :



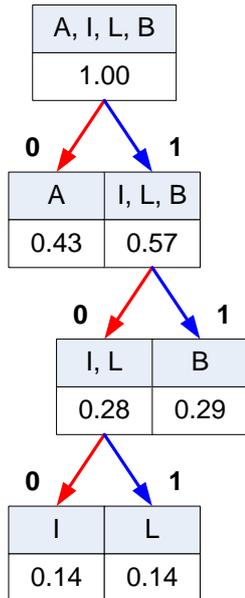
Gabungkan lagi dua simbol dengan probabilitas kemunculan terkecil kemudian urutkan kembali, menjadi seperti berikut :



Lakukan lagi hal yang sama hingga menjadi seperti berikut :



Beri label dari akar ke daun, sisi kiri = 0, sisi kanan = 1, menjadi seperti berikut :



Penelusuran dari akar ke daun (dari atas ke bawah) menghasilkan Kamus Huffman seperti yang ditunjukkan dalam Tabel 3.

Tabel 3. Kamus Huffman Hasil Penelusuran Pohon Biner

char	frekuensi	Kode Huffman	Jumlah Bit
A	3	0	1
B	2	11	2
I	1	100	3
L	1	101	3

Dengan menggunakan Kamus Huffman seperti yang ditunjukkan dalam Tabel 3 maka string “ALIBABA” dikodekan menjadi :

0 101 100 11 0 11 0

Ukuran string sebelum kompresi (dalam kode ASCII) adalah :

$$bitsize_{ASCII} = 7 \times 8 = 56 \text{ bit}$$

Ukuran string setelah kompresi (dalam kode Huffman) adalah :

$$bitsize_{Huffman} = (3 \times 1) + (2 \times 2) + (1 \times 3) + (1 \times 3) = 13 \text{ bit}$$

Rasio kompresi yang dihasilkan adalah :

$$R = \left(1 - \frac{13}{56}\right) \times 100\% \approx 76.8\%$$

Untuk proses ekstraksi pesan dari pesan terkompresi adalah dengan menggunakan Kamus Huffman seperti yang ditunjukkan dalam Tabel 3.

3.3. Steganografi

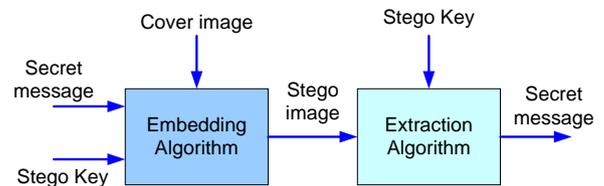
Dalam bidang keamanan data, steganografi digunakan untuk menyembunyikan pesan rahasia ke dalam *cover-media*. Pesan disamarkan sedemikian rupa sehingga tidak diketahui oleh pihak lain. Steganografi

dapat digunakan pada berbagai macam bentuk data seperti teks, citra, audio, dan video. Terdapat beberapa kriteria untuk menghasilkan *stego-media* yang baik yaitu sebagai berikut :

1. *Imperceptible*. Keberadaan pesan rahasia tidak dapat dipersepsikan oleh inderawi. Jika pesan rahasia disisipkan ke dalam suatu citra maka citra yang telah disisipi tidak dapat dibedakan oleh mata dengan citra yang belum disisipi.
2. *Fidelity*. Kualitas media penampung tidak mengalami banyak perubahan setelah disisipi.
3. *Recovery*. Pesan rahasia yang disembunyikan harus dapat diungkap kembali.
4. *Robustness*. Pesan rahasia yang disembunyikan harus tahan terhadap manipulasi pada *cover-media*.

(Emam, 2016).

Jika pesan rahasia berupa teks dan *cover-media* berupa citra maka model steganografi seperti yang ditunjukkan dalam Gambar 1.



Gambar 1. Model Steganografi

Performansi steganografi dengan *cover-media* berupa citra adalah pengukuran terhadap kualitas citra dengan menggunakan MSE (*Mean Square Error*) dan PSNR (*Peak Signal to Noise Ratio*) yang dinyatakan dengan :

$$MSE = \frac{1}{M * N} \sum_{i=1}^M \sum_{j=1}^N (X_{ij} - X^*_{ij})^2 \quad (1)$$

$$PSNR = 10 \log_{10} \frac{I^2}{MSE} \quad (2)$$

dimana X_{ij} adalah intensitas piksel baris ke i dan kolom ke i dari *cover-image*, X^*_{ij} adalah intensitas piksel baris ke i dan kolom ke i dari *stego-image*, M dan N adalah ukuran baris dan kolom *cover-image*, dan I adalah intensitas piksel maksimum. Untuk citra 8 bit maka $I = 255$. Semakin besar PSNR (semakin kecil MSE) maka kualitas *stego image* akan semakin baik. Di dalam penelitian ini direncanakan PSNR di atas 50 dB.

3.4. Binary as Decimal Code (BDC)

BDC adalah metode penyisipan suatu data dimana data yang sudah dikodekan ke dalam format bit biner disisipkan sebagai kode desimal. Pada contoh sebelumnya, string “ALIBABA” telah dikodekan menjadi 0101100110110. Semua kode bit biner ini kemudian ditambahkan ke setiap intensitas piksel dari *cover-media* yang memiliki nilai intensitas piksel < 255. Sebagai contoh suatu *cover-image* berukuran 2x3 seperti berikut :

208 32 160 231 200
 55 32 233 160 155
 67 75 87 200 205

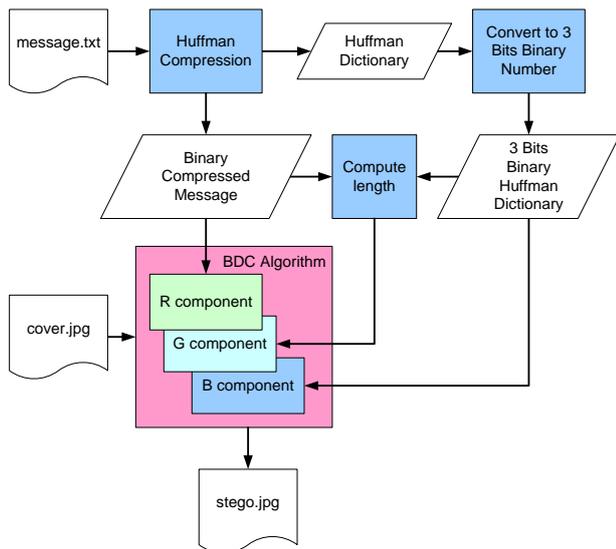
Pesan rahasia terkompresi yang sudah dalam bentuk bit biner kemudian ditambahkan ke *cover-image* seperti berikut :

208 33 160 232 201
 55 32 234 161 155
 68 76 87 200 205

Nilai piksel yang tebal adalah piksel yang sudah ditambahkan dengan kode biner pesan rahasia terkompresi.

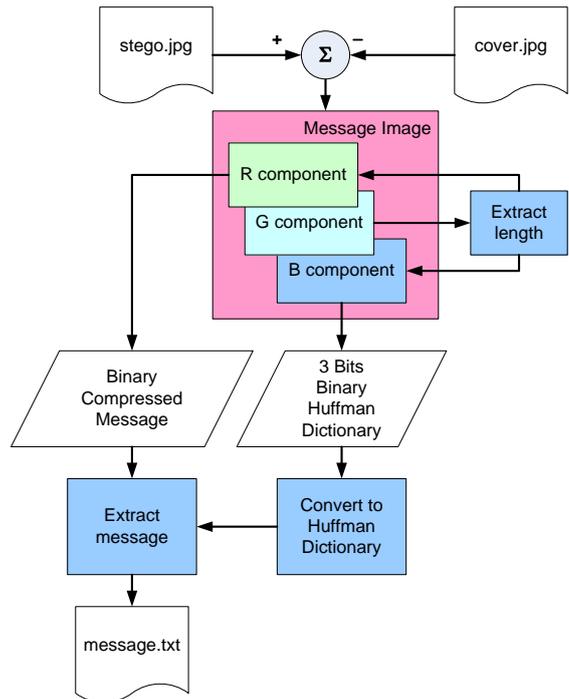
4. RANCANGAN APLIKASI

Di dalam penelitian ini, pesan berupa teks yang tersimpan dalam *message.txt*. Sedangkan *cover-media* berupa citra RGB dengan format JPG. Aplikasi dibangun dengan menggunakan M-File Matlab ver.7.6.0.324 (R2008a). Algoritma program ditunjukkan dalam Gambar 2.



Gambar 2. Proses Embedding

Algoritma ekstraksi pesan dari *stego-image* ditunjukkan dalam Gambar 3.



Gambar 3. Proses Ekstraksi

5. IMPLEMENTASI

Sebagai pengujian digunakan 3 (tiga) file pesan rahasia yang tersimpan dalam *file.txt* seperti berikut :

pesan1.txt :

Dosen adalah pendidik profesional dan ilmuwan berdasarkan Permenristekdikti no.20 tahun 2017

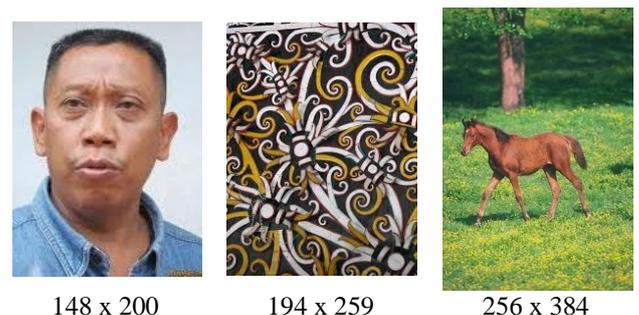
pesan2.txt :

Untuk meningkatkan kualitas dan kuantitas publikasi ilmiah di Indonesia maka dosen harus menghasilkan paling sedikit tiga karya ilmiah yang diterbitkan dalam jurnal nasional

pesan3.txt :

Standar Kompetensi Lulusan merupakan kriteria minimal tentang kualifikasi kemampuan lulusan yang mencakup sikap, pengetahuan, dan ketrampilan yang dinyatakan dalam Rumusan Capaian Pembelajaran Lulusan

Cover image yang digunakan adalah 3 (tiga) file citra RGB dengan format JPG seperti dalam Gambar 4.



Gambar 4. Cover image

Hasil pengukuran performansi dari pengujian aplikasi ditunjukkan dalam Tabel 4.

Tabel 4. Hasil pengukuran performansi

Cover Image	Ukuran Pesan (bit)	Rasio Kompresi (%)	MSE	PSNR (dB)
148 x 200	736	47.69	0.4000	52.12
	1384	51.08	0.4253	51.84
	1600	49.75	0.4295	51.80
194 x 259	736	47.69	0.1525	56.30
	1384	51.08	0.1625	56.02
	1600	49.75	0.1641	55.98
256 x 384	736	47.69	0.0779	59.21
	1384	51.08	0.0831	58.94
	1600	49.75	0.0839	58.89
Rata-rata	1240	49.51	0.2199	55.68

Tabel 4 menunjukkan bahwa kualitas *stego-image* sangat baik. Untuk rata-rata ukuran bit pesan 1240 diperoleh rasio kompresi sebesar 49.51% dengan RMSE sebesar 0.22 dan RPSNR sebesar 55.68 dB (di atas 50 dB).

6. KESIMPULAN

Dari penelitian yang telah dilakukan dapat disimpulkan sebagai berikut :

1. Pesan yang terkompresi dengan menggunakan Algoritma Huffman menghasilkan bit pesan yang jauh lebih kecil dibandingkan dengan pesan tanpa kompresi. Hal ini dapat meningkatkan ruang kemungkinan yang semakin besar bila disisipkan ke dalam citra RGB.
2. Penyisipan pesan terkompresi ke dalam citra RGB dengan menggunakan BDC pada tiap komponen R, G, dan B dapat menjamin kualitas *stego-image* oleh karena setiap intensitas piksel citra terpilih hanya ditambah dengan 0 atau 1.
3. Kombinasi pesan terkompresi huffman dengan penyisipan menggunakan BDC dapat meningkatkan keamanan pesan yang disembunyikan.
4. Dengan menyisipkan Kamus Huffman ke dalam citra RGB maka proses ekstraksi pesan tidak memerlukan *security key*. Dalam hal ini, ekstraksi pesan dari *stego-image* hanya memerlukan *cover image*.

7. SARAN

Metode yang telah diusulkan di dalam penelitian ini dapat ditingkatkan dengan target sebagai berikut :

1. Memperbesar ruang kemungkinan penyisipan bit pesan terkompresi dengan metode rata-rata menyebar ke seluruh komponen R, G, dan B dari *cover-image*.
2. Ruang kemungkinan penyisipan bit pesan terkompresi juga dapat dilakukan dengan

mengkompresi Kamus Huffman dengan cara yang sama kemudian hasil kompresi Kamus Huffman disisipkan dengan cara yang sama.

8. DAFTAR PUSTAKA

- Emam, Aly, Omara, 2016, *An Improved Image Steganography Method Based on LSB Technique with Random Pixel Selection*, International Journal of Advanced Computer Science and Applications, Vol.7, No.3, 2016.
- Ramya, Pushpa, 2016, *A Survey on LossLess and Lossy Data Compression Methods*, International Journal of Computer Science and Engineering Communications, Vol.4, Issue. 1, Page.1277-1280, (2016).
- Senthilkumar, Mathiavanan, 2016, *Analysis of Data Compression Techniques using Huffman Coding and Arithmetic Coding*, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 6, Issue 5, (2016).

Ansar Rizal. Lahir di Makassar, 9 Agustus 1970. Menyelesaikan studi sarjana (S1) Jurusan Teknik Elektro di Universitas Muslim Indonesia di Makasar tahun 1995. Menyelesaikan studi pascasarjana (S2) Ilmu Komputer FMIPA di Universitas Gadjahmada tahun 2008. Mulai tahun 1999 bekerja sebagai pengajar di Jurusan Teknik Elektro Politeknik Negeri Samarinda, dan sebagai pengajar di Jurusan Teknologi Informasi Politeknik Negeri Samarinda mulai tahun 2013 sampai dengan sekarang.

Rihartanto. Lahir di Sawahlunto, 5 Desember 2971. Menyelesaikan studi sarjana (S1) Jurusan Manajemen Informatika dan Teknik Komputer di Institut Sains dan Teknologi "Akprind" Yogyakarta tahun 1996. Mulai tahun 2003 bekerja sebagai pengajar di Jurusan Teknologi Informasi Politeknik Negeri Samarinda sampai dengan sekarang.