

# IMPLEMENTASI METODE HUFFMAN UNTUK KOMPRESI CITRA HASIL DARI STEGANOGRAFI *DISCRETE COSINE TRANSFORM* (DCT)

Iqbal Bagaskoro Aji<sup>1</sup>, Wijanarto<sup>2</sup>

Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Jl. Nakula I No. 5-11, Semarang, Jawa Tengah 50131 - (024) 3517261

E-mail : [iqbal.bagas@gmail.com](mailto:iqbal.bagas@gmail.com)<sup>1</sup>, [wijanarto.udinus@gmail.com](mailto:wijanarto.udinus@gmail.com)<sup>2</sup>

---

## *Abstrak*

Berkembangnya ilmu pengetahuan dalam dunia informatika memungkinkan pengamanan data dari ancaman yang ada. Salah satu teknik pengamanan data yaitu dengan menggunakan Steganografi. Steganografi adalah teknik menyembunyikan data rahasia di dalam wadah (media) digital sehingga keberadaan data rahasia tersebut tidak diketahui oleh orang. Disisi lain, aplikasi menginginkan representasi citra dengan kebutuhan memori yang sesedikit mungkin. Kompresi citra bertujuan meminimalkan kebutuhan memori. Pada penelitian ini akan membahas mengenai Implementasi Metode Huffman untuk Kompresi Citra Hasil Dari Steganografi *Discrete Cosine Transform* (DCT). Pada tahap akhir akan dilakukan pengukuran besarnya kesalahan steganografi dan tingkat kualitas citra menggunakan MSE dan PSNR. Hasil pengukuran nilai PSNR dan MSE pada steganografi DCT, menunjukkan bahwa nilai PSNR diatas 30 dB dan MSE mendekati nol. Itu berarti kualitas citra stego sangat baik. Pada penelitian ini digunakan nilai D (*Distance*) untuk proses penyisipan. Semakin besar nilai D, ketahanan pesan yang disisipkan semakin baik. Akan tetapi kualitas citra menjadi sangat buruk. Dengan D=1 sampai D= 6 perbedaan citra tidak dapat diketahui oleh penglihatan manusia, tetapi dengan menggunakan D=400 citra menjadi rusak dan perbedaannya dapat terlihat jelas oleh penglihatan manusia. Kompresi algoritma huffman tidak merubah ukuran citra dikarenakan hasil stego DCT sudah melakukan kuantisasi citra sampai piksel yang paling minimum. Jadi tidak perlu digunakan kompresi algoritma huffman untuk citra hasil steganografi *Discrete Cosine Transform* (DCT).

**Kata Kunci:** Citra Digital, Steganografi, DCT (*Discrete Cosine Transform*), Huffman.

## *Abstract*

*Knowledge in informatics grows significantly. It enables data security from existing threats. One of data security techniques is Steganography. Steganography is a technique to hide secret data in the digital media so that the existence of secret data is not known by people. On the other side, the applications require minimum image representation memory. This research discusses the Huffman method implementation to Image Compression from Steganography Result using Discrete Cosine Transform (DCT). Steganography error and steganography image quality level measured using MSE and PSNR. The DCT steganography simulation results show that the PSNR value is more than 30 dB and the MSE values approach zero. It means that steganography image quality is very good. In this research, the value of Distance (D) is needed for the insertion process. The larger D value, insertion message robustness is getting better. However, the image quality is lower. Using D = 1 and D = 6, the difference of image can not be known by human vision, but by using D = 400, the image is damaged and the difference of image can be seen clearly by human vision. Huffman compression algorithm does not change the image size due to the DCT stego results already doing image quantization process to the pixel image minimum. It means that the Huffman compression algorithm is not needed to be used for Discrete Cosine Transform (DCT) steganography image.*

**Keywords:** Digital Image, Steganography, DCT (*Discrete Cosine Transform*), Huffman.

## 1. PENDAHULUAN

Berkembangnya ilmu pengetahuan dalam dunia informatika memungkinkan pengamanan data dari ancaman yang ada. Sebagai contoh adalah kriptografi, yaitu ilmu yang digunakan untuk menjaga keamanan dari pihak yang tidak memiliki hak akses terhadap suatu data, baik data berupa *e-mail*, dokumen, maupun berkas pribadi. Namun disisi lain kriptografi dapat menimbulkan kecurigaan pada orang yang membaca data terenkripsi. Kecurigaan ini dapat memicu orang untuk memecahkan enkripsi tersebut walau membutuhkan waktu yang cukup lama. Teknik lain sebagai upaya pengamanan data adalah steganografi [1]. Steganografi adalah teknik menyembunyikan data rahasia di dalam wadah (media) digital sehingga keberadaan data rahasia tersebut tidak diketahui oleh orang.

Banyak metode yang dikembangkan dalam steganografi, salah satunya adalah metode *Discrete Cosine Transform* (DCT). Untuk komponen frekuensi ruang pada citra, penglihatan manusia relatif kurang peka terhadap frekuensi tinggi dibandingkan dengan frekuensi rendah. Jadi dimungkinkan untuk mengidentifikasi dan menghilangkan komponen frekuensi tinggi yang tak dapat dideteksi mata tersebut tanpa mengurangi kualitas citra. Transformasi matriks dua dimensi dari nilai piksel menjadi matriks komponen frekuensi spasial dapat dilakukan dengan menggunakan teknik matematika yang disebut dengan *Discrete Cosine Transform* (DCT) [3]. Dengan kata lain DCT adalah sebuah transformasi yang digunakan untuk mengubah sebuah sinyal diskrit dari bidang waktu (*time domain*) ke dalam bidang frekuensi (*frequency domain*).

Disisi lain, kebanyakan aplikasi menginginkan representasi citra dengan kebutuhan memori yang sesedikit mungkin. Kompresi citra (*image compression*) bertujuan meminimalkan kebutuhan memori untuk merepresentasikan citra digital. Prinsip umum yang digunakan dalam kompresi citra adalah mengurangi duplikasi data di dalam citra sehingga memori yang dibutuhkan untuk merepresentasikan citra menjadi lebih sedikit dari pada representasi citra semula. Kompresi citra memberikan manfaat yang sangat besar dalam industri multimedia saat ini. Salah satunya adalah pada proses pengiriman data (*data transmission*) pada saluran komunikasi data. Citra yang telah dikompresi membutuhkan waktu pengiriman yang lebih singkat dibandingkan dengan citra yang tidak dikompresi [4].

Banyak metode yang dikembangkan dalam kompresi citra, salah satunya adalah metode Huffman. Algoritma kompresi Huffman atau disebut dengan *encoding* Huffman adalah algoritma yang dipakai untuk mengkompresi *file*. Teknik kompresi ini dengan menggunakan code yang lebih kecil pada karakter yang sering dipakai dan *code* yang lebih panjang untuk karakter yang tidak begitu sering dipakai [4].

Pada skripsi ini, peneliti akan membahas mengenai kompresi citra menggunakan metode Huffman pada hasil steganografi *Discrete Cosine Transform* atau yang sering disingkat DCT.

## 2. LANDASAN TEORI

### 2.1 Steganografi

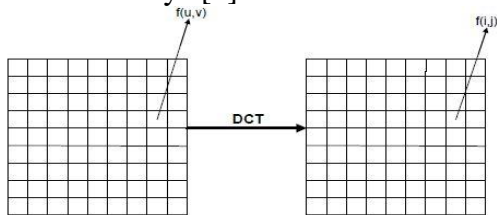
Steganografi merupakan teknik menyembunyikan pesan rahasia ke dalam pesan lainnya sebagai wadah (media) sedemikian rupa sehingga

keberadaan pesan rahasia tersebut tidak diketahui atau disadari keberadaannya oleh orang lain. Steganografi membutuhkan dua properti: pesan sebagai wadah penampung dan pesan rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital baik sebagai wadah penampung maupun data rahasia, misalnya citra, suara (audio), teks, dan video [3].

Steganografi memiliki dua proses, yaitu *encoding* dan *decoding*. *Encoding* merupakan proses penyisipan pesan kedalam media penampung (*covertext*) dalam hal ini adalah gambar/citra digital, sedangkan *decoding* adalah proses ekstraksi pesan dari citra stego (*stego image*). Kedua proses tersebut mungkin memerlukan kunci rahasia (*stegokey*) untuk proses penyisipan pesan dan ekstraksi pesan, agar hanya pihak yang berhak saja yang dapat melakukan penyisipan dan ekstraksi pesan [2].

## 2.2 Discrete Cosine Transform (DCT)

Metode DCT yaitu suatu teknik yang digunakan untuk melakukan konversi sinyal kedalam komponen frekuensi pembentuknya dengan memperhitungkan nilai real dari hasil transformasinya [8].



Gambar 2.1 Transformasi DCT

Proses penyisipan pesan dengan metode DCT dilakukan dengan cara melakukan transformasi DCT dari citra digital [9]. Fungsi transformasi DCT adalah:

$$f(u,v) = \left(\frac{2}{N}\right)^{\frac{1}{2}} \left(\frac{2}{M}\right)^{\frac{1}{2}} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} f(i,j) \cos\left(\frac{\pi \cdot u}{2N} 2i+1\right) \cos\left(\frac{\pi \cdot v}{2M} 2j+1\right)$$

Ukuran image adalah  $M \times N$ . Sedangkan  $f(u,v)$  adalah nilai koefisien DCT pada matrik  $8 \times 8$  kolom ke- $u$  baris ke- $v$ . Adapun  $f(i,j)$  merupakan nilai data yang hendak ditransformasikan pada matrik  $8 \times 8$  kolom ke- $i$  baris ke- $j$ .

Tahap berikutnya adalah menghitung nilai koefisien DCT dan menggantikan bit pada citra tersebut dengan bit pesan yang akan dimasukkan agar tidak terlihat. Kemudian dilakukan transformasi invers DCT sebelum menyimpannya sebagai citra yang disisipi pesan. Fungsi *invers* DCT sebagai berikut:

$$f(i,j) = \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} \left(\frac{2}{N}\right)^{\frac{1}{2}} \left(\frac{2}{M}\right)^{\frac{1}{2}} f(u,v) \cos\left(\frac{\pi(2i+1)u}{2N}\right) \cos\left(\frac{\pi(2j+1)v}{2M}\right)$$

## 2.3 Huffman

Algoritma kompresi Huffman atau disebut dengan *encoding* Huffman adalah algoritma yang dipakai untuk mengkompresi *file*. Teknik kompresi ini dengan menggunakan code yang lebih kecil pada karakter yang sering dipakai dan *code* yang lebih panjang untuk karakter yang tidak begitu sering dipakai [4].

Algoritma Huffman adalah algoritma pemampatan citra yang menggunakan pendekatan statistik. Urutan langkah proses *encode* algoritma ini adalah sebagai berikut [7].

1. Urutkan nilai-nilai *grayscale* berdasarkan frekuensi kemunculannya.
2. Gabung dua buah pohon yang mempunyai frekuensi kemunculan terkecil dan urutkan kembali.
3. Ulangi langkah (2) sampai tersisa satu biner.

4. Beri label pohon biner tersebut dengan cara sisi kiri pohon diberi label 0 dan sisi kanan pohon diberi label 1.
5. Telusuri pohon biner dari akar ke daun. Barisan label-label sisi dari akar ke daun adalah kode Huffman.

### 3. ANALISIS DAN PEMBAHASAN

#### 3.1 Pesan yang disisipkan

Pesan yang disisipkan merupakan berkas dengan ekstensi .txt dengan nama *pesan*. Jumlah blok dalam berkas citra merupakan kelompok piksel berukuran 8x8 piksel dalam berkas citra. Pada proses penyisipan, tiap bit pesan disimpan dalam 1 blok. Pesan teks yang disisipkan diubah ke dalam bentuk biner sehingga 1 karakter pesan disimpan dalam 8 bit. Dengan demikian jumlah karakter pesan maksimum adalah seperti yang dirumuskan pada persamaan berikut ini:

Jumlah karakter pesan =

$$\frac{\text{Jumlah blok dalam berkas citra}}{8}$$

Pada penelitian ini, berkas pesan rahasia yang akan digunakan adalah *rahasia.txt* yang berisi “Steganografi DCT”. Dengan matlab, pembacaan pesan ini diubah ke dalam bentuk biner 8 bit seperti terlihat pada Tabel 1.

Tabel 1 Berkas *rahasia.txt*

Pesan dalam bentuk teks : “Steganografi DCT”	
Pesan dalam bentuk biner 8 Bit :	
01010011→ S	01110010→ r
01110100→ t	01100001→ a
01100101→ e	01100110→ f
01100111→ g	01101001→ i
01100001→ a	00100000→
01101110→ n	01000100→ D
01101111→ o	01000011→ C
01100111→ g	01010100→ T

#### 3.2 Proses Penyisipan

Penjelasan langkah-langkah proses penyisipan pesan teks ke dalam citra sampel berdasarkan metode yang diusulkan pada bab sebelumnya adalah sebagai berikut :

##### 1. Citra Sampul

Citra sampul yang dipilih dalam proses penyisipan adalah citra *grayscale* dengan format (\*.jpg), (\*.bmp), (\*.png), dan (\*.tif), yang memiliki ukuran 256x256 piksel.



Gambar 1 Citra Sampul

##### 2. Transformasi Citra Sampul dengan DCT

Pada proses steganografi citra yang menggunakan *domain transform*, citra sampel yang merupakan domain spasial ditransformasikan terlebih dahulu menjadi domain frekuensi menggunakan *Discrete Cosinus Transform (DCT)*. Adapun Rumus transformasi dengan DCT dalam bentuk matriks adalah sebagai berikut :

$$[S] = T.M.T^t$$

S = Matriks citra hasil transformasi

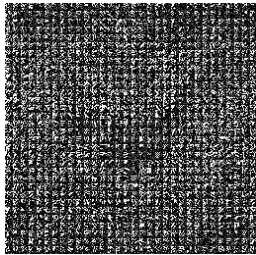
T = Matriks DCT

M = Matriks Original yang sudah dikurangi 128

T<sup>t</sup> = Matriks T transpose

Citra sampel pada gambar 1 yang digunakan sebagai media penampung di transformasikan terlebih dahulu dengan menggunakan DCT sehingga

diperoleh citra dalam domain frekuensi seperti pada gambar 2.



Gambar 2 Citra sampul setelah proses transformasi

### 3. Proses Penyisipan Pesan

Pesan teks yang akan disisipkan disimpan dalam berkas *rahasia1.txt* dan *rahasia2.txt*. Pesan tersebut dibaca dan diubah ke dalam bentuk biner. Pesan yang telah diubah menjadi bentuk biner akan berbentuk bit yang bernilai “0” atau “1”. Sehingga diperoleh jumlah bit pesan adalah :

Jumlah bit pesan = jumlah karakter pesan x 8 bit.

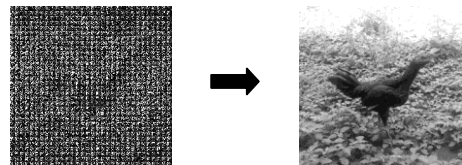
Proses penyisipan pesan teks ke dalam citra sampul dilakukan dengan menggunakan perbedaan nilai piksel (4,1) dan (2,3) pada tiap blok citra. Pertama, Matlab membaca satu blok citra dan mengalihragamkannya dengan DCT. Kemudian, berdasarkan bit pesan digunakan algoritma sebagai berikut:

1. Untuk bit pesan yang disisipkan bernilai “0”. Besar nilai pada lokasi piksel (2,3) < (4,1). Jika nilai pada lokasi piksel (2,3) > (4,1), maka kedua piksel ditukar.
2. Untuk bit pesan yang disisipkan bernilai “1”. Besar nilai pada lokasi piksel (2,3) > (4,1). Jika pada blok citra piksel pada lokasi (4,1) > (2,3), maka kedua piksel ditukar.
3. Dihitung nilai mutlak dari perbedaan besar nilai piksel (2,3) dan nilai piksel (4,1). Perbedaan ini disebut *Distance (D)*.

$$|(2,3) - (4,1)| > D$$

Apabila perbedaan piksel (2,3) dan piksel (4,1) kurang dari nilai D, maka kedua piksel ditambah dengan sembarang angka sehingga terpenuhi persyaratan pada persamaan (4.3). Semakin besar nilai D maka ketahanan pesan terhadap serangan semakin tinggi, akan tetapi kualitas citra semakin menurun.

Proses penyisipan bit berikutnya dilanjutkan ke blok selanjutnya dengan arah kiri ke kanan dan atas ke bawah. Proses ini dilakukan sampai semua bit pesan teks disisipkan dalam citra sampul. Setelah dilakukan proses penyisipan pesan teks dengan menggunakan algoritma diatas, blok tersebut dikembalikan ke bentuk semula dengan menggunakan *invers DCT*. Proses ini dilakukan untuk mengembalikan citra sampul yang berada pada domain frekuensi ke domain spatial sehingga diperoleh Citra Stego seperti pada gambar 3.



Gambar 3 Citra Stego

### 3.2 Proses Ekstraksi

Pada proses ekstraksi pesan, tiap blok dialihragamkan dengan DCT, kemudian diperiksa nilai mana yang lebih besar antara nilai piksel (2,3) dan (4,1). Hal ini menentukan apakah bit “0” atau “1” yang diwakili oleh blok-blok pada berkas citra sampul. Jika nilai piksel (2,3) < (4,1) maka bit pesan yang diwakili adalah bit “0”. Jika nilai piksel (2,3) > (4,1) maka bit pesan yang diwakili adalah bit “1”. Kemudian bit-bit pesan yang dihasilkan digabung dalam variabel *binerpesan* dan diubah

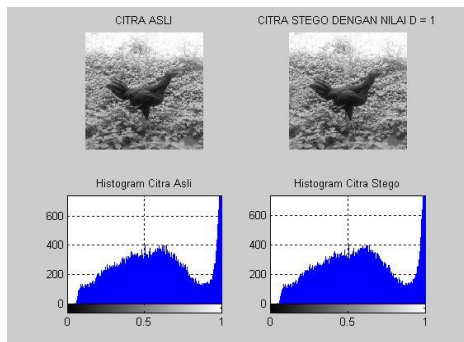
menjadi pesan semula seperti yang tersimpan dalam berkas *rahasia.txt*.

### 3.3 Hasil Simulasi dan Analisis

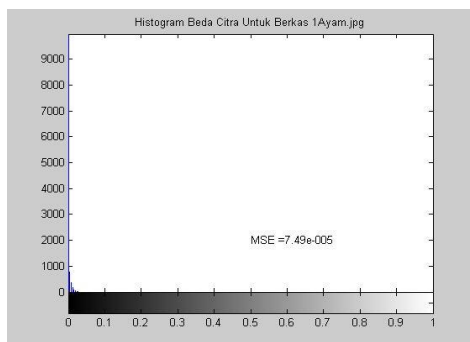
#### 1. Penyisipan pesan

Simulasi untuk penyisipan pesan dilakukan dengan menggunakan citra sampel Ayam.jpg, Balap.jpg, Persiapan.jpg, dan Rafting.jpg. Pesan yang disisipkan menggunakan berkas *rahasia.txt*.

Pesan disisipkan ke dalam citra sampel dengan menggunakan D bernilai dari satu sampai dengan delapan. Hasil dari penyisipan pesan ke dalam citra sampel tersebut disimpan sebagai citra stego. *File* citra stego diberi nama sesuai dengan nama file citra sampel dan nilai D. Misalnya *file* Ayam.jpg dengan nilai D=1, citra hasil stegonya disimpan dengan nama Ayamstego1.jpg. Berikut ini merupakan perbandingan citra sampel dan citra stego dengan D=1.



Gambar 4 Perbandingan hasil penyisipan citra asli dan citra stego Ayam.jpg menggunakan D=1



Gambar 5 Histogram perbedaan citra untuk Ayam.jpg

Tabel 2 Hasil perhitungan PSNR dan MSE untuk citra (\*.jpg)

Citra Sampul	D	Citra Stego	PSNR (dB)	MSE
Ayam.jpg, (77 Kb)	1	Ayamstego1.jpg, (19 Kb)	50,92	7,49e-05
Ayam.jpg, (77 Kb)	2	Ayamstego2.jpg, (19 Kb)	50,90	7,63e-05
Ayam.jpg, (77 Kb)	3	Ayamstego3.jpg, (19 Kb)	50,83	7,92e-05
Ayam.jpg, (77 Kb)	4	Ayamstego4.jpg, (19 Kb)	50,77	8,19e-05
Ayam.jpg, (77 Kb)	5	Ayamstego5.jpg, (19 Kb)	50,71	8,42e-05
Ayam.jpg, (77 Kb)	6	Ayamstego6.jpg, (19 Kb)	50,50	9,44e-05

Tabel 3 Hasil perhitungan PSNR dan MSE untuk citra (\*.bmp)

Citra Sampul	D	Citra Stego	PSNR (dB)	MSE
Ayam.bmp, (66 Kb)	1	Ayamstego1.bmp, (66 Kb)	50,92	7,44e-05
Ayam.bmp, (66 Kb)	2	Ayamstego2.bmp, (66 Kb)	50,90	7,59e-05
Ayam.bmp, (66 Kb)	3	Ayamstego3.bmp, (66 Kb)	50,83	7,85e-05

Tabel 4 Hasil perhitungan PSNR dan MSE untuk citra (\*.png)

Citra Sampul	D	Citra Stego	PSNR (dB)	MSE
Ayam.png, (50 Kb)	1	Ayamstego1.png, (49 Kb)	50,92	7,44e-05
Ayam.png, (50 Kb)	2	Ayamstego2.png, (49 Kb)	50,90	7,59e-05
Ayam.png, (50 Kb)	3	Ayamstego3.png, (49 Kb)	50,84	7,85e-05

Tabel 5 Hasil perhitungan PSNR dan MSE untuk citra (\*.tif)

Citra Sampul	D	Citra Stego	PSNR (dB)	MSE
Ayam.tif, (97 Kb)	1	Ayamstego1.tif, (65 Kb)	50,92	7,44e-05
Ayam.tif, (97 Kb)	2	Ayamstego2.tif, (65 Kb)	50,90	7,59e-05
Ayam.tif, (97 Kb)	3	Ayamstego3.tif, (65 Kb)	50,84	7,85e-05

Dari Tabel 2 sampai dengan Tabel 5, dapat dilihat bahwa ukuran citra hasil stego format citra (\*.jpg) lebih rendah dibandingkan dengan format citra lain. Hal ini dikarenakan format citra (\*.jpg) mengambil ukuran yang besar ketika mengalami proses kompresi, dimana terdapat proses kompresi pada steganografi menggunakan DCT.

Tabel tersebut juga memperlihatkan nilai PSNR dan MSE untuk format citra (\*.jpg), (\*.bmp), (\*.png), dan (\*.tif). PSNR dan MSE menunjukkan perbedaan citra sampul dan citra hasil stego. Tingkat kesamaan citra yang tinggi antara citra sampul dan citra stego adalah jika nilai PSNR tinggi dan nilai MSE kecil. Standar nilai PSNR adalah diatas 30 dB. Dari Tabel 2, 3, 4 dan 5 dapat dilihat bahwa semakin besar nilai D, nilai PSNR semakin kecil. Jadi, semakin besar nilai D, tingkat kesamaan antara citra sampul dan citra hasil stego semakin menurun. Hal ini dikarenakan semakin besar nilai D, perbedaan piksel yang disisipkan pesan teks antara citra sampul dan citra hasil stego bernilai besar, sehingga perubahan piksel ini menyebabkan perbedaan citra yang cukup signifikan. Jika perbedaan piksel ini sangat besar, perbedaan citra sampul dan citra hasil stego dapat dilihat dengan kasat mata yang menyebabkan citra hasil steganografi mudah untuk dideteksi.

Tabel 2, 3, 4 dan 5 juga memperlihatkan nilai MSE. Untuk format citra (\*.jpg), (\*.bmp), (\*.png), dan (\*.tif), semakin besar nilai D, semakin besar nilai MSE. Hal ini menunjukkan bahwa tingkat kesamaan citra semakin menurun. Tingkat kesamaan citra yang tinggi ditunjukkan dengan nilai MSE yang kecil, yaitu mendekati nol.

Pada penelitian ini, citra hasil stego dengan format citra yang berbeda mempunyai nilai PSNR diatas standar yaitu diatas 30 dB dan nilai MSE mendekati nol, sehingga dengan menggunakan DCT, citra hasil stego hampir sama dengan citra sampulnya, sehingga citra hasil steganografi sulit untuk dideteksi. Teknik steganografi menggunakan DCT baik digunakan untuk berbagai

macam format citra baik format (\*.jpg), (\*.bmp), (\*.png), dan (\*.tif). Akan tetapi, dengan metode ini, kapasitas pesan yang dapat disisipkan kurang besar karena satu blok citra hanya dapat menyimpan satu bit pesan teks.

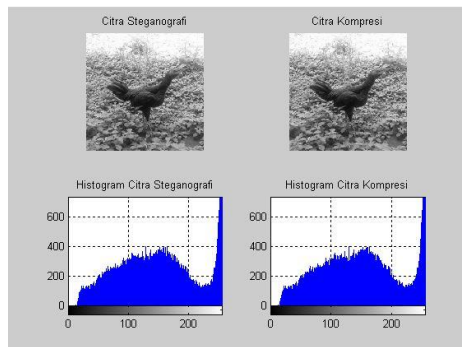
## 2. Kompresi Citra Stego

Kompresi dilakukan setelah proses steganografi, yaitu pada citra sampul yang telah disisipkan pesan teks atau citra stego. Hal ini dilakukan untuk mengetahui pengaruh kompresi terhadap citra stego. Dalam pengiriman data, ukuran data yang kecil dapat mempersingkat waktu pengiriman data. Untuk itu, penelitian ini melakukan proses kompresi pada citra stego agar ukuran citra stego dapat diminimalkan. Proses kompresi ini juga dilakukan untuk melihat apakah pesan teks yang disembunyikan pada proses steganografi dengan DCT dapat diekstrak dengan baik jika mengalami proses kompresi.

Pada penelitian ini, proses kompresi dilakukan dengan menggunakan algoritma Huffman. Dipilih algoritma Huffman karena algoritma ini merupakan algoritma yang baik dan handal untuk kompresi data.

Setelah dilakukan kompresi dengan menggunakan algoritma huffman, citra hasil kompresi disimpan sebagai citra kompresi. Penamaan file citra kompresi menyesuaikan nama file citra stego, misal Ayamstego2.jpg berarti disimpan dengan nama Ayamkompres2.jpg dan seterusnya. Berikut ini merupakan perbandingan hasil citra stego Ayamstego2.jpg dan citra stego kompresi Ayamkompresi2.jpg.





**Gambar 6 Perbandingan citra stego sebelum dan sesudah dikompres**

Dari Gambar 6 dapat dilihat bahwa citra stego dan citra kompresi tidak mengalami perubahan. Ukuran citra stego dan citra kompresi juga tetap, nilai PSNR =  $\infty$  dan MSE = 0. Pada hasil proses ekstraksi citra kompresi sama dengan hasil proses ekstraksi citra stego. Hal ini dikarenakan citra stego hasil steganografi dengan DCT merupakan citra dengan piksel yang sudah minimum, sehingga ketika dilakukan proses kompresi, tidak terdapat perubahan citra. Jadi, pada penelitian ini, citra stego hasil steganografi dengan DCT tidak perlu diberi proses kompresi dengan algoritma huffman.

## 4. KESIMPULAN DAN SARAN

### 4.1 Kesimpulan

Dari analisis dan pembahasan hasil simulasi seperti yang dikemukakan pada bab sebelumnya, maka penulis dapat mengambil kesimpulan sebagai berikut :

- Metode steganografi digital dengan *Discrete Cosine Transform* menghasilkan nilai PSNR yang bagus yaitu diatas 30 dB.
- Semakin besar nilai D, maka ketahanan pesan yang disisipkan semakin baik. Namun kualitas citra yang disisipi akan sangat menurun.
- Kompresi algoritma huffman tidak merubah ukuran citra dikarenakan hasil stego DCT sudah melakukan kuantisasi citra sampai piksel yang

paling minimum. Jadi tidak perlu digunakan kompresi algoritma huffman untuk citra hasil stego DCT.

### 4.2 Saran

Adapun saran-saran yang dapat diberikan dalam penelitian ini untuk pengembangan lebih lanjut agar meningkatkan kualitas dan fungsionalitas dari metode steganografi ini, adalah sebagai berikut :

- Mengembangkan metode steganografi DCT agar kualitas citra stego bisa lebih baik lagi.
- Mencoba metode kompresi lain untuk citra hasil stego *Discrete Cosine Transform* (DCT).

## DAFTAR PUSTAKA

- [1] M. Edisuryana, R. R. Isnanto dan M. Somantri, "Aplikasi Steganografi Citra Berformat Bitmap dengan menggunakan metode *End Of File*", *Transient*, vol. 2, no. 3, 2013.
- [2] B. Rakhmat dan M. Fairuzabadi, "Steganografi Menggunakan Metode Least Significant Bit dengan Kombinasi Algoritma Kriptografi Vigenere Dan RC4", *Jurnal Dinamika Informatika*, vol. 5, no. 2, 2010.
- [3] A. Prabowo, A. Hidayanto dan Y. Christiyono, "Penyembunyian Data Rahasia pada Citra Digital Berbasis *Chaos* dan *Discrete Cosine Transform*", *Transmisi*, vol. 13, no. 2, 2011.
- [4] Faradisa dan B. F. Budiono, "Implementasi Metode Huffman Sebagai Teknik Kompresi Citra", *Jurnal Elektro Eltek*, vol.2, no. 2, 2011.
- [5] C-C. Chang, P-Y. Lin, J-C. Chuang, "A Grayscale Image Steganography Based upon Discrete Cosine Transformation",



- Journal of Digital Information Management*, vol.8, no.2, 2010.
- [6] M. Yunus dan A. Harjoko, "Penyembunyian Data pada File Video Menggunakan Metode LSB dan DCT", *IJCCS*, vol.8, no. 1, 2014.
- [7] T. Suyono, S.Si., M.Kom., E. Mulyanto, S.Si., M.Kom., Dr. V. Suhartono, O. D. Nurhayati, M.T dan Wijanarto, M.Kom., "Teori Pengolahan Citra Digital", *Andi*, 2009.
- [8] H. Patel dan P. Dave, "Steganography Technique Based on DCT Coefficients", *International Journal Of Engineering Research and Applications*, vol. 2, no. 1, 2012.
- [9] P. V. Bodhak dan B. L. Gunjal, "Improved Protection In Video Steganography Using DCT & LSB", *International Journal Of Engineering and Innovative Technology*, vol. 1, no. 4, 2012.
- [10] C. Iswahyudi, "Prototipe Aplikasi Untuk Mengukur Kematangan Buah Apel Berdasarkan Kemiripan Warna", *Journal Teknologi*, vol. 3, no. 2, pp. 107-112, 2010.
- [11] M. Wahid, "Steganografi Citra Digital Dengan Discrete Wavelet Transform (DWT) dan Discrete Cosine Transform (DCT)", 2014.
- [12] M. K. Mathur, S. Loonker dan Dr. D. Saxena, "Lossless Huffman Coding Technique For Image Compression And Reconstruction Using Binary Trees", *IJCTA*, vol. 3, no. 1, 2012.
- [13] R. Wissarto, "Implementasi Slantet Transform (SLT) dan Huffman Coding Pada Steganografi Citra *Grayscale*", 2014.
- [14] R. A. Saragih dan H. Gunawan, "Simulasi Penyembuhan Error pada Citra Menggunakan Metode Multi Directional Interpolation (MDI)", *Electrial Engineering Journal*, vol. 2, no. 1, pp. 13-27, 2011.
- [15] N. Jain, S. Meshram and S. Dubey, "Image Steganography Using LSB and Edge Detection Technique", *International Journal of Soft Computing and Engineering (IJSCE)*, Vol.2, no. 3, 2012.
- [16] D. Putra, "Pengolahan Citra Digital", *Andi*, 2010