

# Studi Dan Implementasi Steganografi Pada Video Digital Di Mobile Phone Dengan DCT Modification

Paul Gunawan Hariyanto (13504023)

Teknik Informatika, Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Bandung  
e-mail: paul\_xp87@yahoo.com

**Abstrak** - Steganografi merupakan ilmu dan seni yang mempelajari cara penyembunyian informasi rahasia ke dalam suatu media sedemikian sehingga manusia tidak dapat menyadari keberadaan pesan tersebut. Pada makalah ini, dilakukan studi mengenai bagaimana steganografi pada media video digital. Video yang digunakan memiliki format 3GP dan codec H.263, dan teknik steganografi akan menggunakan DCT Modification, yaitu melakukan perubahan terhadap koefisien DCT (Discrete Cosine Transform) pada video sesuai dengan pesan masukan.

Terdapat juga sebuah perangkat lunak yang dihasilkan, yang memiliki fungsi utama untuk melakukan steganografi pada media video 3GP. Penggunaan kunci juga dilakukan untuk memperkuat keamanan, dimana hanya kunci yang digunakan pada saat penyisipan saja yang dapat mengekstraksi pesan tersebut. Perangkat lunak ini dibangun pada perangkat mobile phone yang mendukung aplikasi Java dengan konfigurasi CLDC 1.1 dan MIDP 2.0. Kakas pembangun yang digunakan adalah Java 2 Micro Edition, dengan IDE NetBeans, dan emulator Sun Java Wireless Toolkit.

**Kata kunci:** steganografi, DCT, video 3GP, mobile phone.

## 1. PENDAHULUAN

Steganografi merupakan salah satu bagian dari kriptografi, yaitu ilmu dan seni dalam menyembunyikan pesan rahasia sedemikian sehingga manusia tidak dapat menyadari keberadaan pesan tersebut. Pada masa kini, steganografi lebih banyak dilakukan pada data digital, dengan menggunakan bentuk media digital seperti teks, gambar, audio, atau video [03].

Proses penyisipan pesan pada steganografi membutuhkan dua buah masukan, yaitu pesan yang media penyisipan, pesan yang ingin disisipkan, serta kunci sebagai pengaman. Sistem proses ini ditunjukkan pada Gambar 1.

Penjelasan detail mengenai format 3GP dapat dilihat pada [01], sedangkan codec H.263 dapat dilihat pada [02]. Proses penyisipan diawali dengan mengklasifikasi *frame* dari video yang akan

ingin disembunyikan, dan media penyisipan. Hasil dari proses ini dinamakan dengan *stego-object*, yaitu suatu media yang mirip dengan media pada masukan, yang sudah terdapat pesan tersembunyi di dalamnya. Kebanyakan media yang merupakan *stego-object* tidak dapat dikembalikan lagi seperti semula, karena data dari media *stego-object* sudah diubah.

Teknik steganografi yang digunakan adalah DCT Modification, yang bekerja pada domain frekuensi. Oleh karena itu, cara ini hanya dapat dipakai pada format gambar yang disimpan dalam domain frekuensi, seperti JPEG yang menggunakan Discrete Cosine Transform (DCT) sebagai proses transformasi domain.

Proses steganografi biasanya terdapat masukan kunci tambahan, yang berguna untuk menambah faktor keamanan, sehingga hanya kunci yang digunakan pada saat penyisipan saja yang dapat mengekstraksi pesan.

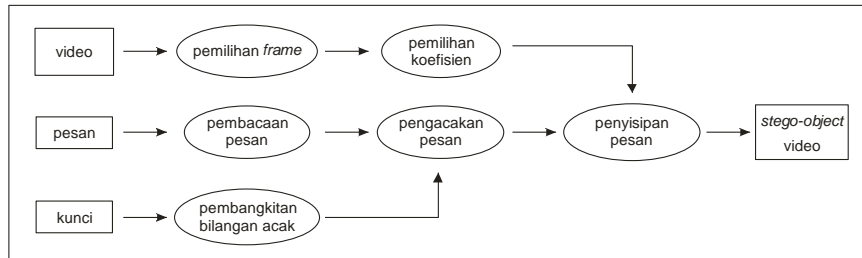
Dalam makalah ini juga akan dibahas mengenai dampak perubahan dari video yang dihasilkan setelah penyisipan, yang akan menggunakan metode pengukuran secara subjektif dan objektif. Subjektif berarti dilakukan pengamatan secara langsung, dan objektif akan menggunakan metode SSIM (*Structural SIMilarity*) yang mengukur tingkat perbedaan video tersebut.

## 2. ANALISIS

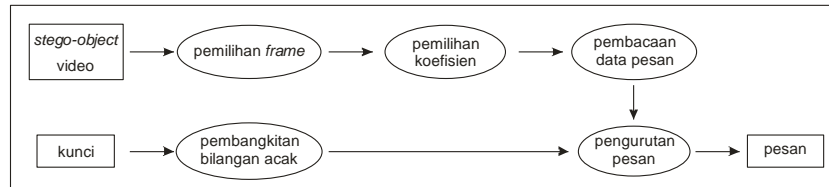
Berikut ini adalah analisis dari proses penyisipan pesan dan proses ekstraksinya. Terdapat juga analisis mengenai penggunaan kunci, pengukuran kualitas video, serta pengaruh perangkat *mobile phone* pada implementasi perangkat lunak.

### 2.1. Penyisipan Pesan

Sistem untuk menyisipkan pesan pada video, membutuhkan masukan berupa video sebagai disisipkan. Video codec H.263 yang digunakan memiliki dua jenis *frame*. Jenis *frame* pertama adalah *I-frame*, yang menggunakan kompresi *intraframe* dengan DCT. Kemudian jenis *frame* yang kedua adalah *P-frame* yang menggunakan kompresi *interframe* yaitu *motion estimation* dan *compensation*.



**Gambar 1 Sistem Penyisipan Pesan**



**Gambar 2 Sistem Ekstraksi Pesan**

Teknik penyisipan pesan yang akan digunakan adalah *DCT Modification*, yang berarti melakukan perubahan pada koefisien DCT. Oleh karena itu, penyisipan hanya dapat dilakukan pada *I-frame* yang menyimpan koefisien DCT di dalamnya. Perubahan pada *frame* ini akan mempengaruhi semua *P-frame* berikutnya, karena kompresi *interframe* menjadikan *I-frame* ini sebagai *frame* referensi, sehingga perubahan pada *I-frame* akan ikut terbawa. Setelah didapatkan *frame* yang sesuai, dicari juga koefisien yang dapat disisipkan, yaitu INTRA-DC, yang merupakan koefisien DC dari matriks frekuensi pada gambar.

Kedua masukan lainnya, yaitu pesan dan kunci, dipakai untuk menghasilkan pesan yang acak. Proses ini akan dijelaskan pada Subbab 2.3.

Objek keluaran yang dihasilkan adalah sebuah *stego-object*, yaitu video yang sudah memiliki pesan di dalamnya. Video ini akan mengalami sedikit penurunan kualitas dibandingkan video masukan, dan diharapkan penurunan ini tidak dapat dideteksi oleh manusia. Untuk menguji penurunan kualitas video, akan dilakukan pengukuran kualitas video yang akan dibahas pada Subbab 2.4.

## 2.2. Ekstraksi Pesan

Sistem untuk mengekstraksi pesan dari video memerlukan dua buah masukan, yaitu video yang mengandung pesan, serta kunci sebagai pengaman. Sistem ini ditunjukkan pada Gambar 2.

Proses ekstraksi pesan dimulai dengan pemilihan *frame* dan koefisien pada video yang akan dibaca. Kemudian pesan di dalamnya dibaca, menjadi pesan dalam bentuk acak. Kunci yang dimasukkan akan menjadi penentu kebenaran pesan, dimana deretan bilangan acak yang dibangkitkan oleh kunci akan mengatur bagaimana urutan pesan tersebut. Hanya kunci yang digunakan pada saat penyisipan yang dapat menghasilkan pesan asli

kembali. Pada proses ini, sistem tidak bisa mengetahui apakah kunci yang dimasukkan benar atau salah, karena kunci masukan hanya digunakan sebagai pengatur data saja. Proses pengurutan dengan kunci akan dijelaskan pada Subbab 2.3.

Sistem ekstraksi pesan tidak dapat mengembalikan *stego-object* menjadi video 3GP yang asli, karena pesan di dalam video sudah menjadi bagian dari video tersebut.

## 2.3. Penggunaan Kunci

Pesan yang disisipkan akan melalui proses pengacakan terlebih dahulu, sehingga proses ekstraksi nantinya juga harus diurutkan kembali. Kedua proses ini menggunakan kunci, yaitu sebagai *seed* dalam pembangkitan deretan bilangan acak yang menjadi pengatur letak pesan. Pesan diubah ke dalam bentuk biner, dan pengacakan atau pengurutan dilakukan dengan mengubah letak biner tersebut.

Deretan bilangan acak ini memakai algoritma LCG (*Linear Congruential Algorithm*), yang telah terdapat pada Java. Nilai *seed* dibangkitkan melalui fungsi MD5 dari *string* kunci, yang menjadi sebuah bilangan dengan ukuran 63-bit. Karena MD5 yang dihasilkan adalah 128-bit, maka hanya digunakan separuh pertama saja.

Jumlah bilangan acak yang dihasilkan adalah sebanyak biner pesan, dan proses pengurutan akan memakai deretan yang sama untuk mengembalikannya menjadi pesan yang asli.

## 2.4. Pengukuran Kualitas Video

Metode proses pengukuran kualitas video akan dilakukan secara subjektif dan objektif. Cara subjektif yaitu dengan melakukan pengamatan langsung terhadap video hasil penyisipan dan video yang asli.

Sedangkan cara objektif akan memakai teknik *Structural Similarity* (SSIM). Pemilihan metode ini dikarenakan akan lebih cocok untuk pengukuran terhadap video yang telah mengalami kompresi, dan perbandingan yang berbasis struktur dari gambar lebih mirip terhadap persepsi manusia. Kode SSIM dapat diambil pada [04].

Langkah-langkah yang dilakukan pada pengukuran kualitas video adalah:

1. Mengekstrak semua *frame* pada video 3GP yang asli dan video yang sudah disisipkan pesan.
2. Menggunakan penghitungan SSIM untuk mengukur kualitas dari semua *frame*, pada kedua video tersebut.
3. Mencari nilai index *Mean* SSIM (MSSIM), yaitu rata-rata dari semua nilai SSIM yang diperoleh.

Untuk nilai batas perbandingan, diambil nilai antara dari kisaran MSSIM, yaitu 0,7. Sehingga apabila nilai MSSIM yang dihasilkan lebih besar atau sama dengan 0,7, maka video hasil penyisipan sudah dapat dikatakan baik. Demikian juga sebaliknya, nilai MSSIM lebih kecil dari 0,7 berarti video yang dihasilkan memiliki perbedaan yang cukup signifikan dibandingkan video aslinya.

### 2.5. Implementasi Pada Mobile Phone

Perangkat lunak akan diimplementasikan pada *mobile phone*, dimana memiliki beberapa perbedaan dengan implementasi pada komputer biasa. Salah satunya adalah terbatasnya jumlah memori yang dapat digunakan, yang mengakibatkan pembacaan video atau pesan harus dilakukan secara bagian per bagian. Dampak lainnya adalah pengacakan pesan tidak dapat dilakukan langsung terhadap pesan secara utuh, tetapi tiap sebagian saja, dimana dalam hal ini sejumlah 1024-bit, atau 128 *byte*.

Perbedaan lainnya adalah pengembangan pada perangkat *mobile phone* biasanya lebih lambat, jika dibandingkan dengan perangkat lunak pada komputer biasa. Selain itu, tidak seragamnya spesifikasi antar *mobile phone*, sehingga implementasi pada *mobile phone* yang berbeda akan membutuhkan pengembangan perangkat lunak yang berbeda juga. Sebagai contoh, apabila ingin dibuat perangkat lunak yang dapat mendukung MIDP 1.0, maka perangkat lunak tersebut harus dikembangkan ulang yang khusus mendukung MIDP 1.0, karena akan terdapat beberapa fungsi yang tidak kompatibel dengan perangkat lunak pada MIDP 2.0.

## 3. HASIL DAN PEMBAHASAN

Berdasarkan hasil analisis, telah berhasil dikembangkan perangkat lunak yang memiliki fungsi untuk menyisipkan dan mengekstraksi pesan. Kemudian dilakukan pengujian untuk memeriksa kebenaran dari perangkat lunak tersebut, beserta kinerja perangkat lunak tersebut.

Untuk menguji kebenaran perangkat lunak, dilakukan pengujian dengan melakukan penyisipan dan ekstraksi suatu pesan pada video. Terdapat dua jenis kunci dipakai, yaitu kunci yang benar dan salah. Kunci yang benar diharapkan akan menghasilkan pesan yang sama, dan kunci yang salah akan menghasilkan pesan yang berbeda.

Sedangkan untuk menguji kinerja perangkat lunak, dilakukan pengamatan berdasarkan subjektif dan objektif terhadap video hasil penyisipan dan video yang asli.

### 3.1. Pengujian Kebenaran Perangkat Lunak Dengan Kunci Yang Sama

Pengujian ini dilakukan dengan cara menyisipkan pesan ke dalam video, kemudian mengekstraksinya kembali. Video yang menjadi media penyisipan adalah sebuah video 3GP dengan screenshot pada Gambar 3. *File* yang menjadi pesan adalah sebuah *file* teks dan *file* gambar, yang isinya ditunjukkan pada Tabel 1. String kunci yang digunakan adalah *string* 123.



Gambar 3 Screenshot Video Penyisipan

Tabel 1 Isi File Penyisipan

Jenis <i>file</i>	Isi <i>file</i>
<i>File</i> teks	Ini adalah sebuah <i>file</i> untuk pengujian.
<i>File</i> gambar	

Setelah proses penyisipan selesai, dilakukan proses ekstraksi dari masing-masing video. Kunci sama pada proses penyisipan, yaitu string '123', sehingga diharapkan isi pesan yang dihasilkan juga sama. Isi dari *file* hasil ekstraksi, dapat dilihat pada Tabel 2.

**Tabel 2 Isi File Ekstraksi**

Jenis <i>file</i>	Isi <i>file</i>
<i>File</i> teks	Ini adalah sebuah <i>file</i> untuk pengujian.
<i>File</i> gambar	

Dari hasil pengujian, terbukti bahwa perangkat lunak yang dibuat sudah berhasil menjalankan proses penyisipan dan ekstraksi dengan benar. Semua pesan yang menjadi masukan telah berhasil disisipkan, dan kemudian dapat diekstraksi dengan baik. Pesan yang diekstraksi sama dengan pesan yang asli, dan kunci yang digunakan pada proses penyisipan dan ekstraksi juga sama.

### 3.2. Pengujian Kebenaran Perangkat Lunak Dengan Kunci Yang Berbeda

Pengujian ini dilakukan dengan langkah-langkah yang sama pada pengujian sebelumnya, hanya saja menggunakan kunci yang berbeda pada saat ekstraksi, yaitu *string* 456. Sehingga diharapkan isi *file* hasil ekstraksi berbeda dari *file* yang asli. Isi *file* ekstraksi ditunjukkan pada Tabel 3.

**Tabel 3 Isi File Ekstraksi Dengan Kunci Berbeda**

Jenis <i>file</i>	Isi <i>file</i>
<i>File</i> teks	05 "äÉÑ%" . 2 3 □ó < ó □ □ v ° Çø r □ ë ^ □ □ " Â □
<i>File</i> gambar	Gambar tidak bisa dimunculkan



Dari hasil pengujian, terbukti bahwa perangkat lunak telah dapat melakukan aspek penggunaan kunci dengan baik. Proses ekstraksi dengan kunci yang salah dapat ditangani, yaitu dengan menghasilkan pesan yang berbeda dengan pesan yang asli.

### 3.3. Pengujian Kinerja Perangkat Lunak

Pengujian ini dilakukan dengan cara membandingkan video hasil penyisipan pada Subbab 3.1, baik dengan cara subjektif maupun objektif. *Frame* dari kedua video yang ingin dibandingkan tersebut diekstraksi terlebih dahulu,

menjadi gambar berformat JPG. Untuk cara objektif, dicari nilai SSIM dari masing-masing gambar, lalu diambil nilai rata-ratanya atau nilai MSSIM-nya. Hasil pengujian dapat dilihat pada Tabel 4.

**Tabel 4 Hasil Pengujian Kinerja Perangkat Lunak**

Jenis <i>file</i> pesan	Screenshot video	Nilai SSIM
<i>File</i> teks		0.99987
<i>File</i> gambar		0.99627

Pada cara subjektif, video hasil penyisipan dianggap mirip dengan video yang asli. Sedangkan pada cara objektif, semua perbandingan memperoleh nilai SSIM di atas 0,7. Hal ini membuktikan bahwa proses penyisipan dengan metode *DCT Modification* ini tidak mengubah kualitas struktur video secara signifikan. Dan tingginya nilai ini dikarenakan resolusi video 3GP yang cukup kecil, dan SSIM akan lebih akurat dengan resolusi *frame* yang lebih tinggi, sehingga struktur *frame* lebih terlihat. Walaupun demikian, hal tersebut masih konsisten dengan persepsi manusia yang tidak dapat melihat perbedaan dari resolusi video atau *frame* yang kecil.

## 4. KESIMPULAN

Beberapa kesimpulan yang dapat diambil adalah sebagai berikut:

1. Telah berhasil dikembangkan perangkat lunak yang dapat melakukan steganografi pada video 3GP. Kebutuhan fungsional dari perangkat lunak, seperti proses penyisipan dan ekstraksi pesan, serta penggunaan kunci sudah dapat dilakukan dengan benar.
2. Teknik *DCT Modification* sebagai teknik penyisipan pesan sudah dapat dilakukan dengan benar, yaitu menyisipkan pesan ke dalam koefisien INTRA-DC. Keberhasilan

ini terletak pada proses pembacaan atau *parsing* video 3GP dan *codec* H.263, sehingga perubahan yang dilakukan dapat terjadi hanya pada bit video yang sudah ditentukan.

3. Perangkat keras *mobile phone* yang menjadi lingkungan pengembangan perangkat lunak memiliki berbagai keterbatasan, seperti jumlah memori, atau jenis fungsi yang ada, sehingga perlu dicari cara-cara yang lebih lanjut untuk menangani keterbatasan tersebut.

#### DAFTAR REFERENSI

- [01] 3rd Generation Partnership Project. 2007. Transparent end-to-end packet switched streaming service (PSS) 3GPP file format (3GP). France.
- [02] International Telecommunication Union. 2005. ITU-T Recommendation H.263: "Video coding for low bit rate communication".
- [03] Munir, Rinaldi. 2006. *Diktat Kuliah IF5054 Kriptografi*. Indonesia: Institut Teknologi Bandung.
- [04] Wang, Zhou. 2008. <<http://www.ece.uwaterloo.ca/~z70wang/research/ssim/>>  
Tanggal akses 5 Juni 2008