MODEL KEAMANAN PESAN PADA VIDEO MENGGUNAKAN METODE ONE'S COMPLEMENT CRYPTOGRAPHY DAN TRACK FREE ATOM STEGANOGRAPHY 1

Pujianto

Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Raya Ciledug, Petukangan Utara, Pesanggrahan, Jakarta Selatan 12260
Telp. (021) 5853753, Fax : (021)5853752
pujianto@budiluhur.ac.id

ABSTRAK

Steganography merupakan ilmu dan seni yang mempelajari cara penyembunyian informasi rahasia ke dalam suatu media sehingga manusia tidak dapat menyadari keberadaan pesan tersebut. Penyembunyian pesan pada file video dikenal dengan istilah steganografi video. Metode track free atom merupakan metode penyisipan pesan pada hirarki elements free video mp4 yang penggunaannya mudah, cepat dan dapat menampung pesan rahasia dalam jumlah yang relatif banyak. Untuk meningkatkan keamanan pesan rahasia pada video dengan ditambahkan metode One's Complement pada Cryptography. Dengan menggunakan teknik gabungan Cryptography dan Steganography, pesan rahasia yang disisipkan pada video dapat lebih aman dan stego video yang dihasilkan tidak mengalami perubahan kualitas dibandingkan dengan cover videonya sehingga tidak menimbulkan kecurigaan bahwa ada pesan rahasia di dalam video, serta waktu yang dibutuhkan mulai dari penyisipan pesan ke dalam cover video sampai pengambilan pesan dari stego video relatif sangat cepat. Hasil analisis pengujian penyisipan pesan pada video mp4 menggunaan algoritma track free atom berhasil dilakukan dengan baik, bahkan tanpa mempengaruhi kualitas audio dan gambar yang ada didalamnya dikarenakan atom mdat yang digunakan untuk menyimpan sample audio dan gambar tidak dirubah. Ini menyebabkan nilai pengujian menggunakan PSNR adalah 100, APNSR adalah 100, MSE adalah 0, SSIM adalah 1, MSSSIM adalah 1, dan 3-Component SSIM INDEX adalah 1. Sehingga tidak ada perubahan kualitas antara video asli dengan video steganography.

Kata Kunci : *Steganography, One's Complement, Cryptography, Track Free Atom*

I. PENDAHULUAN

Di dalam kehidupan sehari-hari terkadang seseorang memerlukan sistem keamanan dalam berinteraksi dengan orang lain. Steganografi merupakan teknik keamanan data yang dapat digunakan untuk melayani kebutuhan tersebu[1]. Data yang tersembunyi didalam steganografi merupakan hal yang sulit untuk dideteksi dan bila dikombinasikan dengan algoritma yang cocok maka akan lebih sulit untuk diuraikan[2].

Penyembunyian informasi terbagi menjadi dua teknik yaitu Steganografi dan Watermaking[3]. Atas sifat dari media yang bisa digunakan, teknik steganografi dapat dipisahkan menjadi lima jenis, yaitu image, audio, video, text dan protocol[4].

Banyak penulis ataupun artikel yang membahas steganografi, tetapi kebanyakan membahas steganografi pada citra [2], [5], [6], [1], [3], [7], dan audio [8], [9], [10].

Namun pada saat ini, banyak penelitian yang mengembangkan steganografi video. Sudah banyak metode yang dilakukan untuk steganografi dan sudah banyak pula metode steganalisis yang digunakan untuk mendeteksinya. Diantara metode steganografi video adalah metode Least Significant Bit (LSB) dan metode Discrete Cosine Transform (DCT). Metode LSB menyembunyikan bit-bit pesan pada bit-bit segmen frame video. Sedangkan metode DCT menyembunyikan bit-bit pesan dengan melakukan perubahan pada koefisien DCT[11].

Keuntungan dari steganografi video adalah banyaknya data yang dapat disembunyikan didalamnya, serta fakta bahwa video merupakan streams dari beberapa image menyebabkan adanya distorsi pada salah satu frame imge tidak dapat dilihat dengan mudah oleh mata manusia.

Teknik steganografi video lain yang jarang digunakan adalah dengan menyembunyikan pesan pada format media container dari content tersebut. Misalnya, pada sebuah file format container AVI, kita tidak menyembunyikan pesan pada byte data gambar yang disimpan dalam AVI, melaikan kita menyelipkan pesan diantara data gambar dengan mengekspolitasi spesifikasi format cara file AVI menyimpan data, tanpa merusak kemampuan file untuk dibuka oleh video viewer. Keuntungan utama dari metode ini adalah bahwa karena pesan rahasia tidak pernah menyentuh data content, tidak akan ada degradasi content untuk dideteksi [12].

Format media container adalah meta-file format, dengan spesifikasi yang menjabarkan berbagai elemen data dan meta data disimpan secara bersamaan pada file ditigal[12]. Penggunaan paling umum dari format container adalah untuk membungkus data multimedia.

Berdasarkan uraian di atas, peneliti akan mengusulkan teknik steganografi yang mampu menyembukan informasi rahasia di dalam sebuah format media container video. Metode yang digunakan adalah menyisipkan pesan pada track free atom

dari sebuah video yang berformat mp4 dan ditambahkan metode One's Complement Cryptography [13] untuk mengacak pesan sebelum disisipkan. *One's Complement* pada *binary number* (*biner*) adalah *complement* dari 0 dan 1, *complement* 0 adalah 1 dan *complement* 1 adalah 0[14]. Metode *One's Complement* digunakan untuk merubah biner isi pesan dari biner 0 diubah menjadi 1 dan biner 1 menjadi 0

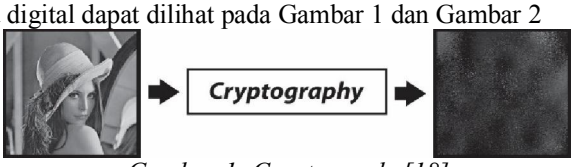
II. LANDASAN TEORI

A. Tinjauan Pustaka

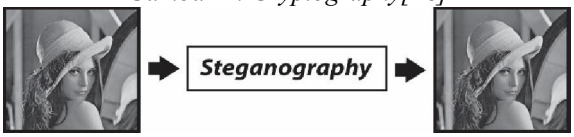
1) *Steganography*

Steganography adalah ilmu dan seni menulis pesan rahasia ke dalam sebuah media dengan menggunakan teknik yang sedemikian rupa sehingga keberadaan pesan rahasia akan sulit disadari dan dideteksi oleh orang lain selain pengirim dan penerima pesan tersebut [15] . Kata *Steganography* berasal dari bahasa Yunani yaitu "*steganos*" yang berarti tersembunyi atau terselubung dan "*graphy*" yang berarti tulisan atau gambar [16]. *Steganography* membutuhkan 2 media untuk pengimplementasiannya yaitu media penyimpan (*cover object*) dan pesan rahasia yang akan disisipkan ke dalam media penyimpanan [17].

Steganography muncul *untuk* menyempurnakan kekurangan *Cryptography* dalam menyembunyikan data penting di dalam sebuah *cover object*, sehingga hanya pihak tertentu yang dimaksudkan bisa mendapatkan pesan yang ingin disampaikan. Kelebihan *Steganography* dibandingkan dengan *Cryptography* adalah pesan-pesannya tidak menarik perhatian orang lain. Perbedaan *Cryptography* dan *Steganography* pada citra digital dapat dilihat pada Gambar 1 dan Gambar 2



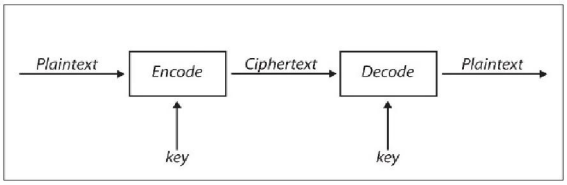
Gambar 1: Cryptography[18]



Gambar 2 : Steganography[18]

2) *Cryptography*

Crytography adalah bidang ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, keabsahan data, integritas data dan autentikasi [19] . *Cryptography* adalah ilmu dan seni yang digunakan untuk keamanan pesan rahasia yang akan dikirimkan dengan cara mengacaukan, menyamarkan atau menyandikan pesan rahasia menjadi bentuk yang tidak dibaca dan tidak dapat dimengerti dengan menggunakan teknik yang disebut *encode* (enkripsi), kemudian pesan rahasia yang telah diubah menjadi *ciphertext* tidak dapat dibaca oleh orang lain selain pengirim dan penerima pesan rahasia tersebut dan proses kebalikan dari *encode* adalah *decode* (dekripsi) [20] . Proses *encode* dan *decode* dapat dilihat pada Gambar 3.



Gambar 3 : Encode dan Decode

Salah satu contohnya adalah *encode* dan *decode* yang digunakan oleh Julius Caesar. Pada proses *encode*, *plaintext* diubah menjadi *ciphertext* yaitu menggantikan masing-masing huruf dengan 3 huruf selanjutnya[21] .

3) *One's Complement*

Sistem yang dikenal dengan nama komplemen satu (*ones' complement*) juga dapat digunakan untuk *merepresentasikan* bilangan negatif. Bentuk komplemen satu untuk bilangan biner negatif diperoleh dengan cara membalik seluruh bit dari bilangan biner positifnya. Metode *One's Complement* pada *binary number* (*biner*) adalah *complement* dari 0 dan 1, di mana *complement* 0 adalah 1 dan *complement* 1 adalah 0 [13] .

Tabel 1: Contoh One's Complement

Karakter	Biner	One's Complement	Karakter
P	01010000	10101111	¯
U	01010101	10101010	ª
J	01001010	10110101	µ
I	01001001	10110110	¶
A	01000001	10111110	¾
N	01001110	10110001	±
T	01010100	10101011	«
O	01001111	10110000	°

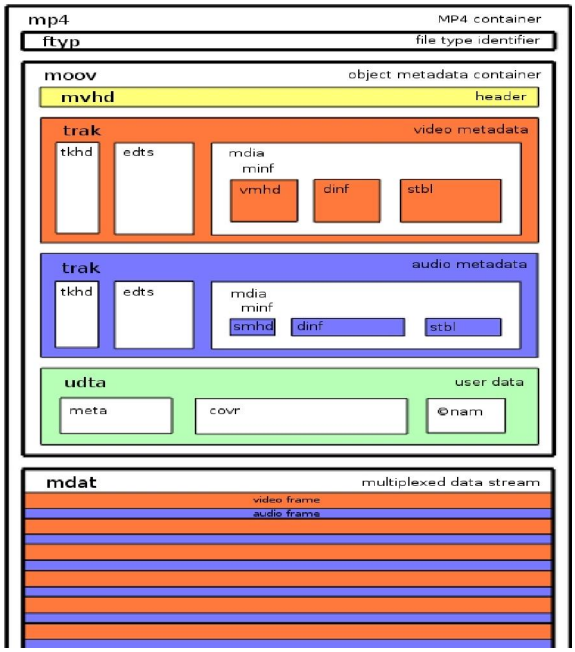
4) MPGEG-4 Part 14 (MP4)

MPEG-4 *Part* 14 atau lebih dikenal sebagai MP4 adalah salah satu format berkas pengodean suara dan gambar/video digital yang dikeluarkan oleh sebuah organisasi MPEG [12] . Ekstensi nama berkas jenis MPEG-4 ini banyak menggunakan .mp4

Dalam spesifikasi file format MP4, terdapat tiga atom utama yang wajib ada, yaitu:

- Atom file header (atom ID "*ftyp*") : atom ini hanya berukuran beberapa bytes dan tujuannya adalah menunjukkan bahwa ini adalah file MP4.
- Atom media header (atom ID "*moov*") : atom ini menyimpan infomasi header dari video yang disimpan dan lokasi data video tersebut dalam atom mdat. Informasi metadata juga disimpan disini.
- Atom media data (atom ID "*mdat*") : Atom ini menyimpan data dari media video dan audio.

Mengenai susunan atom sendiri, satu-satunya aturan adalah bahwa atom ftyp harus disimpan didepan. Untuk tipe atom lain tidak harus berurutan. Struktur berkas MP4 secara umum ditunjukan pada gambar II-8 serta secara keseluruhan ditunjukkan pada gambar 4 dibawah ini.



Gambar 4 : Struktur Atom Video MP4 secara keseluruhan

5) *Peak Signal to Noise Ratio* (PSNR)

Metode Peak *Signal to Noise Ratio (PSNR)* banyak digunakan untuk mengukur perbandingan kualitas warna *cover image* dengan *stego image* dengan satuan *decibel* (dB). Nilai *PSNR* dapat diketahui dengan menghitung terlebih dahulu nilai *Mean Square Error* (*MSE*) dari *cover image* dengan *stego image*. Rumus *MSE* untuk citra digital abu-abu (*grayscale*)[11] dapat dilihat berikut ini:

$$MSE = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \qquad (1)$$

Keterangan:

m = panjang (baris) citra dalam satuan *pixel*

n = lebar (kolom) citra dalam satuan *pixel*

(x, y) = koordinat *pixel*

I(x, y) = nilai warna *pixel cover image* pada baris ke-x, kolom ke-y

K(x, y) = nilai warna *pixel stego image* pada baris ke-x, kolom ke-y

Sedangkan untuk pengukuran kualitas citra pada citra digital berwarna, *MSE* yang digunakan adalah MSE rata-rata[22] dengan rumus sebagai berikut:

$$MSE_{AVG} = \frac{MSE_R + MSE_G + MSE_B}{3} \qquad (2)$$

Rumus yang digunakan untuk menghitung *PSNR* sebagai berikut:

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned} \qquad (3)$$

Kualitas *stego image* dikatakan baik jika *PSNR stego image* bernilai tinggi. Nilai *PSNR* memiliki tingkat proporsi berbanding terbalik dengan nilai *MSE*. Jadi, jika nilai *PSNR* tinggi maka nilai *MSE* rendah. Oleh karena itu semakin baik kualitas *stego image* maka nilai *MSE* akan semakin rendah[22].

6) *Align Peak Signal to Noise Ratio* (APSNR)

Penelitian MPSNR dikembangkan lagi seperti tertulis pada publikasi Yoanda[23]. Akurasi metode MPSNR ditingkatkan dengan metode yang disebut APSNR. Desain APSNR dibagi menjadi tiga bagian utama yang terdiri dari pencarian pasangan frame, pergeseran frame, dan penghitungan final nilai PSNR.

7) *Structural Similiarity* (SSIM)

SSIM adalah metode yang digunakan untuk menghitung kesamaan antara dua gambar. SSIM dirancang sebagai perbaikan metode *Peak Signal to Noise Ratio* (PSNR) dan *Mean Squared Error* (MSE) yang terbukti tidak konsisten dengan persepsi mata manusia[24].

8) *Multi-Scale Structural Similiarity* (MSSSIM)

Indeks Multi-scale SSIM berdasarkan metrik SSIM dari beberapa tingkatan *downscaled* gambar aslinya[25]. Mempersepsikan rincian gambar tergantung kepadatan sampling sinyal gambar, jarak dari bidang gambar untuk pengamat, dan kemampuan perseptual dari sistem visual pengamat. Dalam prakteknya, evaluasi subjektif dari citra yang diberikan bervariasi ketika faktor-faktor ini bervariasi.

9) 3-Component SSIM INDEX

3-Komponen Indeks SSIM berdasarkan pembagian wilayah sumber frame. Ada 3 jenis daerah, yaitu : tepi, tekstur dan daerah halus. Hasil metrik dihitung sebagai rata-rata tertimbang SSIM metrik untuk daerah-daerah. Bahkan mata manusia bisa melihat perbedaan pada daerah bertekstur atau tepi tepatnya dari pada daerah halus. Pembagian berdasarkan besarnya gradien di setiap pixel dari gambar[25].

B. Tinjauan Studi

Beberapa tinjauan studi di bawah ini merupakan acuan adalam penulisan penelitian ini.

Changyoung Xu, Xijian Ping dan Tao Zang [26] mengusulkan algoritma steganografi di dalam stream video MPEG yang terkompres. Algoritma tersebut menyisipkan pesan dalam sebuah I *Frame,* P *Frame* dan B *Frame* pada tiap-tiap GOP dari sebuah video MPEG. Pada pengujian sebuah video yang mempunyai 240 *frame,* setiap GOP mempunyai 12 *frames* yang terdiri dari satu I *frame*, tiga P *frame* dan delapan B *frame*. Setiap frame mempunyai ukuran 325x240, dan memberikan ukuran macro-block 16x16. Dari pengujian

PSNR, didapat nilai I frame 35,2273, P frame 34,6136 dan B frame 33, 3175. Hasil penelitian menunjukkan bahwa algoritma yang diusulkan memiliki karakteristik sedikit merendahkan efek visual, kapasitas embedding yang lebih besar dan menolak pemrosesan video seperti penambahan bingkai atau frame.

Pada penelitian yang dilakukan oleh Kamred Udham Singh[4] melakukan pengujian penyembunyian pesan didalam video avi menggunakan algoritma LSB Substitution. Metode LSB yang diusulkan untuk menyisipkan pesan rahasia kedalam frame video avi yang sudah ditentukan terlebih dahulu frame yang akan disisipkan pesan. Teknik LSB yang diusulkan dimodifikasi dengan cara menentukan posisi bit tertentu pada frame video. Hasil penelitian menunjukan bahwa algoritma lsb dapat digunakan untuk menyisipkan pesan pada image video avi.

K. V. Vinodkumar dan V. Lokesware Reddy [27] membahas tentang penyisipan pesan rahasia kedalam file video menggunakan teknik LSB. Sebelum pesan disembunyikan terdapat proses enkripsi terlebih dahulu menggunakan algoritma DES dan mengkompres pesan dengan algoritma Lempel-Ziv. Proses penyisipan pesan dilakukan dengan cara merubah file video kedalam byte dan dirubah menjadi bit patern. Kemudian setiap karakter dari pesan rahasia dikonfersi menjadi bit patern. Setelah itu mengganti bit LSB dari file video dengan bit pesan. Hasil pengujian pesan dapat disimpan kedalam semua jenis file video dengan teknik LSB.

Pada penelitian yang dilakukan oleh Jason Paul Cruz, Nathaniel Joseph Libatique, dan Gregory Tangonan [28] melakukan penelitian steganografi pada file video flv. Pengujian dengan banyak metode, yaitu FLV *Header is Modified, Tags are Removed in Whole, Stegofile is Injected at End Of File, Stegofile is Hidden at the Metadata, Stegofile is Hidden Before or Inside the Actual Data of Video Tag, Stegofile is Hidden at The End of a Video or Audio Tag, Stegofile is Distributted Among All of The Video tags, Tags are Removed to Compensate for the Size of the Added Stegofile*.

Wafa Hasan Alwan [29] melakukan penelitan steganografi pada video AVI yang belum dikompresi menggunakan metode DLSB. Algoritma DLSB adalah modifikasi dari algoritma LSB. DLSB didapat dengan cara mengambil 4 bit dari setiap RGB cover frame video dan mengambil 4 bit dari setiap RGB pesan frame video kemudian digabungkan.

Penelitian yang dilakukan oleh Ms. D. S. Maind, M. Tech dan Dr. B. K. Sarkar [30] melakukan penelitan steganografi pada file video avi. Pesan disisipkan pada audio yang ada pada video tersebut menggunakan metode LSB. Sebelumnya pesan dienkripsi terlebih dahulu dengan algoritma AES. Hasil dari penelitian tersebut, pesan dapat disisipkan dengan metode LSB.

Sedangkan B. Suneetha [31] melakukan penelitian steganografi pada video avi yang terkompresi menggunakan metode LSB. Hasil yang didapat dari hasil pengujian PSNR adalah nilai rata-rata lebih dari 36 dB. Pesan disisipkan menggunakan basis GOP yaitu disisipkan dalam I-Frame, P-Frame dan B-Frame yang pertama pada video avi yang terkompres menggunakan h264.

Mahmuddin Yunus dan Agus Harjoko [11] melakukan penelitian steganografi video menggunakan 3 metode, yaitu: LSB, DCT dan gabungan LSB-DCT. Hasil pengujian menunjukkan tingkat keberhasilan steganografi video dengan menggunakan metode LSB adalah 38%, metode DCT adalah 90%, dan gabungan metode LSB-DCT adalah 64%. Sedangkan hasil perhitungan MSE, nilai MSE metode DCT paling rendah dibandingkan metode LSB dan gabungan metode LSB-DCT. Sedangkan metode LSB-DCT mempunyai nilai yang lebih kecil dibandingkan metode LSB. Pada pengujian PSNR diperoleh data bahwa nilai PSNR metode DCT lebih tinggi dibandingkan metode LSB dan gabungan metode LSB-DCT. Sedangkan nilai PSNR metode gabungan LSB-DCT lebih tinggi dibandingkan metode LSB.

Sedangkan K. Parvathi Divya dan K. Mahesh [6] melakukan penelitian steganografi pada video MPEG menggunakan metode LSB. Pesan rahasia yang disisipkan berupa gambar. Pesan disisipkan menggunakan basis GOP yaitu disisipkan dalam I-Frame, P-Frame dan B-Frame yang pertama pada video mp4 yang terkompres menggunakan h264. Dari pengukuran PSNR terlihat nilai rata-rata 66.9 dB.

M. Suresh Kumar dan G. Madhavi Latha [32] melakukan penelitian steganografi pada media video avi yang belum terkompresi. Metode yang digunakan untuk menyisipkan pesan rahasia pada frame video adalah DCT. Hasil akhir menyatakan bahwa aplikasi yang dibangun dapat menyisipkan pesan dengan metode tersebut.

P. Paulpandi, Dr. T. Meyyappan, M. Sc., M. Phil, M.BA.,Ph.D[33] melakukan penelitian steganografi pada video MPEG yang terkompressi menggunakan metode Motion Vector. Pesan disisipkan menggunakan basis GOP yaitu disisipkan dalam I-Frame, P-Frame dan B-Frame yang pertama pada video mp4 yang terkompres menggunakan h264. Pesan disisipkan didalam element motion vector dan sebelumnya pesan diacak menggunakan aes.

S. Sarangeswari dan Mrs. V. Aruna [16] melakukan penelitian penyisipan pesan berupa gambar kedalam video mp4 menggunakan metode DCT. Dari penelitian yang dilakukan pesan dapat disisipkan dan diambil kembali dari stego video. Sebelumnya pesan gambar dilakukan proses enkripsi dengan metode RSA.

R. Shanthakumari dan Dr. S. Malliga [15] melakukan penelitian steganografi pada video avi dengan metode LSB Matching Revisited Algorithm. Dengan metode ini dapat melakukan penyisipan dalam jumlah banyak didalam setiap frame yang ada pada video avi. Hasil perhitungan PSNR rata-rata diatas 80 dB.

Penelitian yang dilakukan oleh S. Suma Christal Mary M.E (Ph.D) [34] menggunakan metode *Compressed Video Secure Steganography (*CVSS*)*. Dalam proses penyisipan dilakukan secara realtime pada video mpeg-2. Pesan disisipkan pada *compressed domain.*

Sedangkan penelitian yang dilakukan oleh Alston Evan Wijaya, Henni Rachmawati, dan Yusapril Eka Putra [35] melakukan penelitian steganografi pada video avi menggunakan metode LSB. Besar pesan yang dapat disimpan pada video adalah sekitar 4,1% dari ukuran file video. Video keluaran yang dihasilkan memiliki ukuran file yang sama dengan file video asli dan tidak terjadi perubahan kualitas yang signifikan dimana nilai PSNR dari video keluaran berada di atas 30 dB. Dalam mengimplementasikan steganografi pada media

video avi, sebagai tambahan untuk pengamanan informasi, file pesan akan dienkripsi terlebih dahulu menggunakan algoritma Blowfish dengan kunci yang ditentukan oleh user.

III. METODOLOGI

A. Metode Penelitian

Penelitian ini menggunakan metode penelitian eksperimen untuk melakukan enkripsi pesan teks dengan metode *One's Complement* dan penyisipan pesan teks pada *free atom* dan *atom mdat* pada video mp4 dengan menggunakan ukuran teks yang berbeda-beda. Kemudian dilanjutkan dengan eksperimen kinerja algoritma yang telah diterapkan terhadap ukuran teks yang berbeda. Adapun beberapa eksperimen yang dilakukan dalam penelitian ini meliputi:

- Eksperiment: Masukan
  Panjang karakter diukur dalam satuan byte. Setiap satu karakter berart mempunyai ukuran teks sebesar satu byte, jika ada sepuluh karakter berarti ukuran teks tersebut adalah sepuluh byte. Parameter yang digunakan dalam penelitian ini adalah besarnya ukuran teks yang dinyatakan dalam satuan byte (terdiri dari 8 bit).
- Eksperiment: Proses
  Eksperimen pada proses enkripsi dan metode penyisipan pesan merupakan bagian terpenting dalam penelitian ini, karena akan memberikan pengaruh terhadap hasil penelitian. Pada penelitian ini dilakukan dua proses, yang pertama adalah proses enkripsi teks. Pada tahap ini dilakukan proses pengacakan pesan dengan algoritma *One's Complement.* Sedangkan proses yang kedua adalah proses penyisipan pesan itu sendiri pada *free atom* dan *atom mdat container* video mp4.
- Eksperiment: Hasil Keluaran
  Hasil keluaran merupakan hasil akhir (*output*) berupa file stego video berekstensi mp4 yang terdapat pesan rahasia yang telah dienkripsi sebelumnya. Sehingga apabila pesan berhasil ditemukan oleh seseorang maka orang tersebut tidak akan berhasil membacanya.

B. Metode Pengumpulan Data

Data yang digunakan sebagai sampel penelitian dalam pengujian adalah teks yang memiliki ukuran yang berbeda-beda. Dalam penelitian ini menggunakan tujuh teks yang memiliki ukuran berbeda.

Tabel 2 : Sample teks uji

No	Keterangan	Ukuran File (Bytes)
1	Teks1	1
2	Teks2	2
3	Teks3	8
4	Teks4	22
5	Teks5	214
6	Teks6	642
7	Teks7	1283

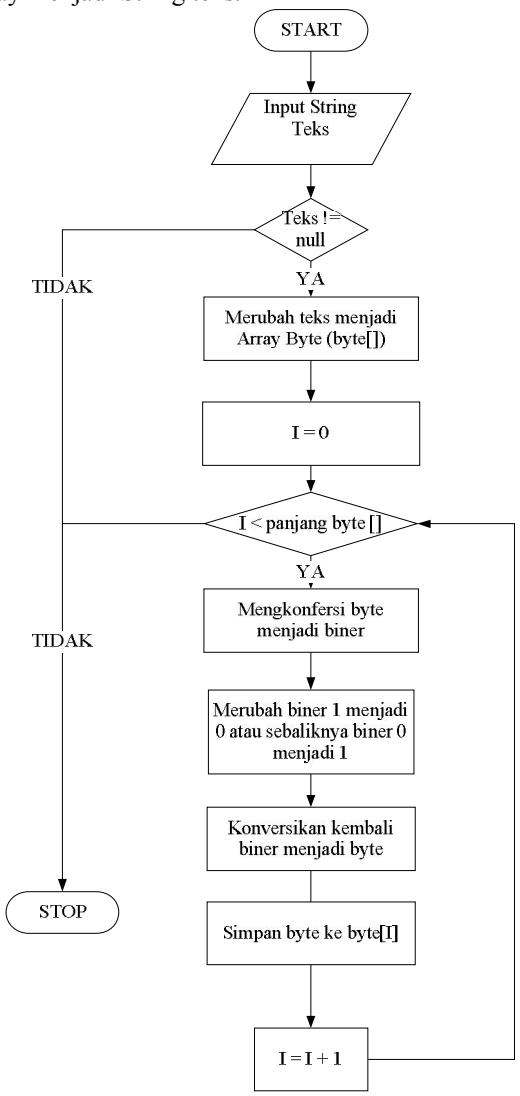
C. Teknik Analisis

Penelitian menggunakan beberapa teknik analisis untuk mencapai hasil yang diharapkan. Teknik analisis pertama adalah analisis proses/cara kerja serta analisis algoritma pada pengacakan pesan atau teknik kriptografi dengan algoritma *One's Complements* dan analisis penyisipan pesan pada *free atom* dan *atom mdat* video mp4. Dan yang terakhir adalah analisis terhadap proses pengujian yakni pengukuran waktu yang dibutuhkan untuk menyisipkan pesan dan perubahan perbedaan video asli dengan stego video.

1) Algoritma *One's Complement*

a) *One's Complement Encoding*

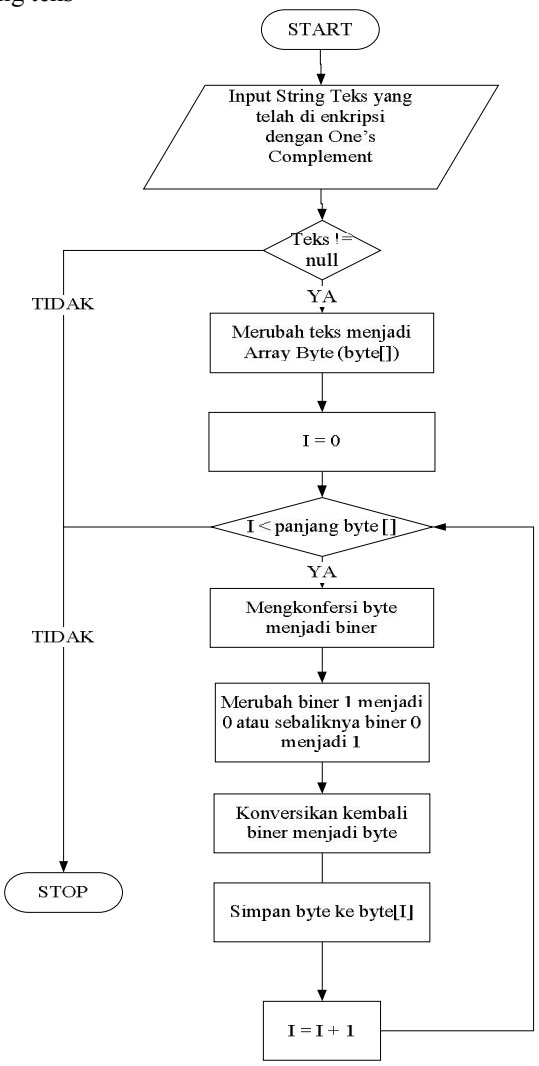
Proses encoding dari algoritma *One's Complement* ini adalah mengkonversikan teks String menjadi *byte array*. Kemudian melakukan perulangan sebanyak panjang byte array, setiap byte dirubah menjadi biner. Setelah itu mengganti biner 1 menjadi 0 atau mengganti biner 0 menjadi 1. Lalu merubah kembali biner menjadi byte. Setelah proses perulangan selesai, simpan byte kedalam byte array. Kemudian konversikan byte array menjadi String teks.



Gambar 5 : Flowchart One's Complement Encoding

b) *One's Complement Decoding*

Pembentukan kembali String teks dari teks yang telah di enkripsi menggunakan *one's complement* dapat dilakukan dengan cara yang sama seperti proses *encode* yaitu dengan mengkonversikan teks String menjadi *byte array*. Kemudian melakukan perulangan sebanyak panjang byte array, setiap byte dirubah menjadi biner. Setelah itu mengganti biner 1 menjadi 0 atau mengganti biner 0 menjadi 1. Lalu merubah kembali biner menjadi byte. Setelah proses perulangan selesai, simpan byte kedalam byte array. Kemudian konversikan byte array menjadi String teks

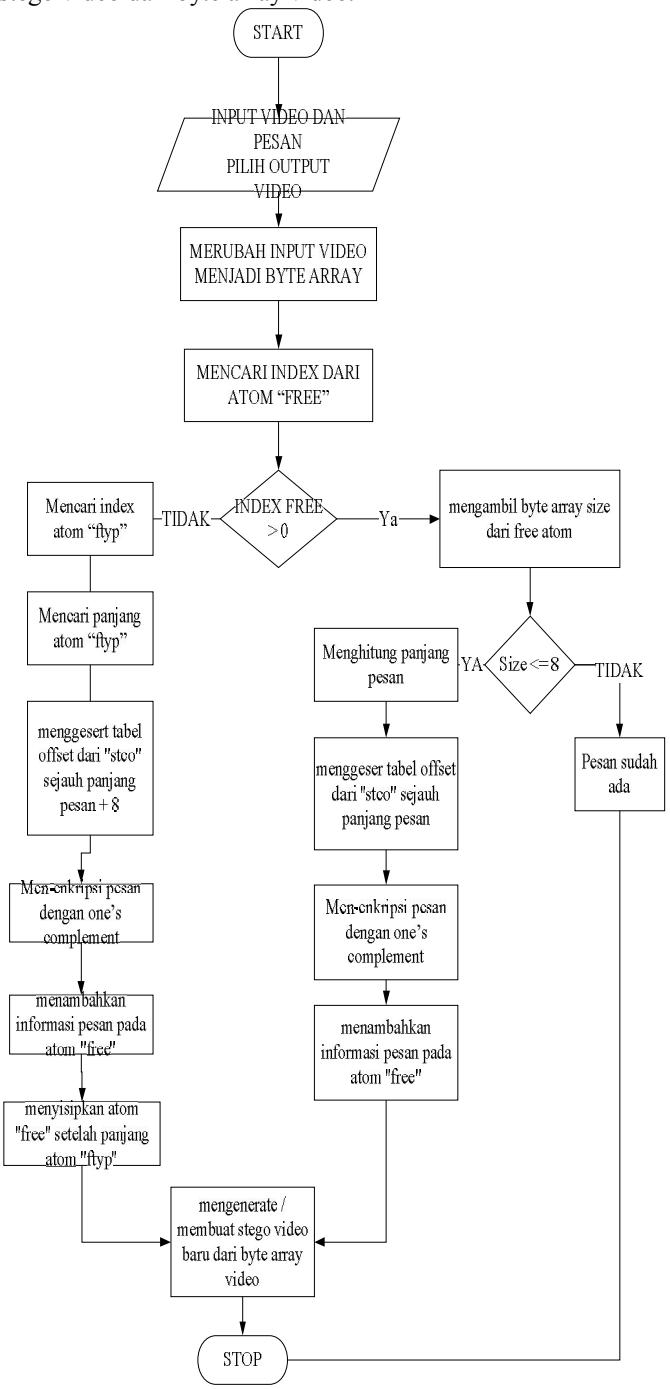


Gambar 6 : Flowchart One's Complement Decoding

2) Penyisipan Pesan pada *Free Atom* dan *Atom mdat* Video MP4

a) Penyisipan *Pesan*

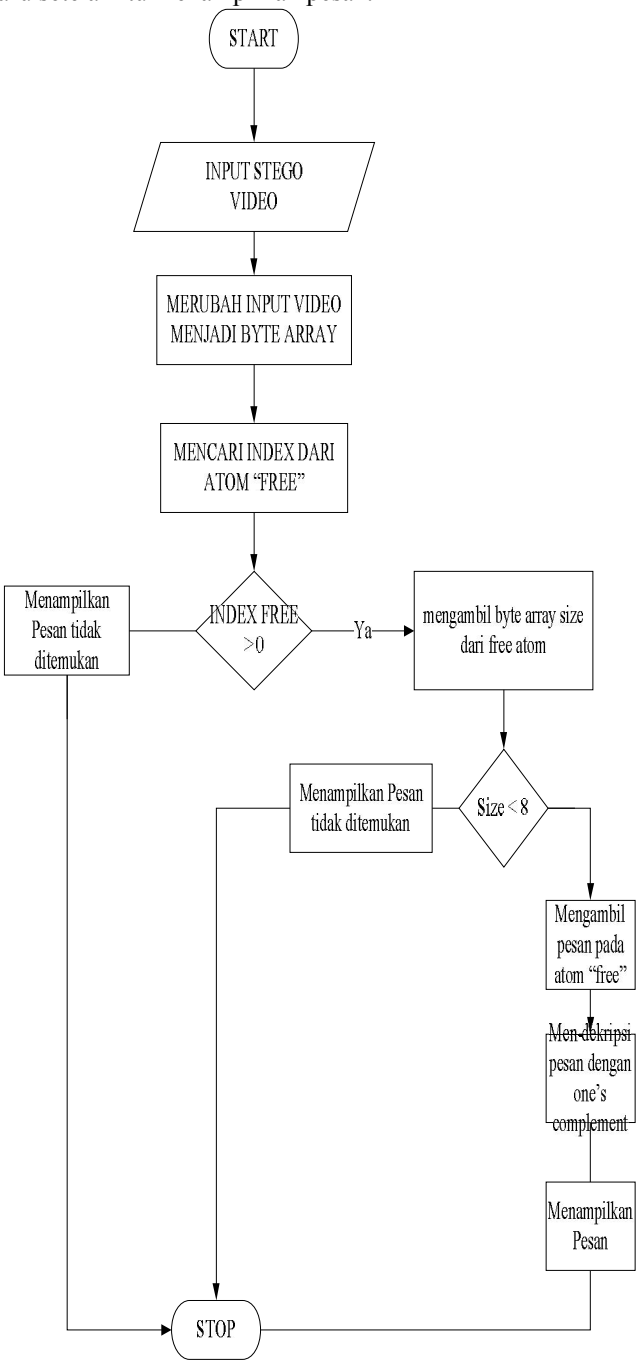
Untuk melakukan penyisipan pesan kedalam video terlebih dahulu mengenkripsi pesan menggunakan algoritma *one's complement* seperti terlihat pada gambar III-4. Jadikan file video menjadi byte array, cari index dari atom "free". Jika index atom "free" lebih dari nol maka ambil byte array "size" dari atom "free". Jika panjang "size" dari atom "free"lebih dari atau sama dengan dari delapan maka akan menampilkan pesan sudah ada jika tidak maka menghitung panjang pesan kemudian menggeser table dari atom "stco" sejauh panjang pesan dan menambahkan pesan kedalam atom "free". Lalu mengenerate file stego video. Jika index atom "free" kurang dari nol maka mencari index atom "ftyp" kemudian mencari panjang atom "ftyp", setelah itu menggeser tabel offset dari atom "stco" sejauh panjang pesan + 8. Lalu menambahkan pesan kedalam atom "free". Sisipkan atom "free" setelah panjang atom "ftyp". Terakhir mengenerate atau membuat file stego video dari byte array video.



Gambar 7 Flowchart Proses Penyisipan Pesan pada file video mp4

b) Pengambilan Pesan

Dalam proses pengambilan pesan dari file stego vidoe mp4 sangatlah mudah. Cukup melakukan pengecekan index dari atom "free". Jika panjang index dari atom "free" kurang dari nol maka pesan tidak ditemukan. Tetapi jika panjangnya lebih dari no maka cek panjang atom "free". Jika panjang atom "free" kurang dari delapan maka menampilkan pesan tidak ditemukan. Jika tidak maka mengambil pesan dari atom "free" dan mendekode pesan menggunakan algoritma one's complement baru setelah itu menampilkan pesan.



Gambar 8 : Flowchart Proses Pengambilan Pesan pada file video

IV. PEMBAHASAN HASIL PENELITIAN

A. Implementasi Pengujian

1) Pengujian *Objective*

a) Pengujian Terhadap Perbandingan Isi File Berkas Rahasia

Hasil pengujian yang dilakukan dengan membandingka isi dan ukuran dari berkas rahasia sebelum dan sesudah dilakukan proses steganografi. Hasil dari pengujian menunjukkan bahwa isi dan ukuran dari file berkas yang berhasil disisipkan sama dengan berkas rahasia aslinya, sehingga pengujian ini dikatakan berhasil. Hasil pengujian dapat dilihat dari tabel dibawah ini:

Tabel 3 : Hasil Pengujian Video Steganografi dengan file Video yang berbeda

No	Nama media Video	Ukuran Video (byte)	Ukuran Teks Pesan Rahasia Sebelum (byte)	Ukuran Teks Pesan Rahasia Sesudah (byte)
1	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	120	120
2	Gara-Gara Rasta.wmv.mp4	9.778.298	200	200
3	KOPI HITAM KUPU KUPU BY MOMONON.mp4	20.535.613	12	12
4	LAGU MOTIVASI PENGHILANG GALAU — STRESS — KANTUK.mp4	108.478.150	234	234
5	Masanies - Pil Koplo.mp4	7.542.368	23	23

b) Pengujian Steganografi Video Berdasarkan Ukuran Teks

Pengujian dengan ukuran pesan rahasia yang berbeda-beda dapat dilihat dari tabel dibawah ini:

Tabel 4 : Pengujian Steganografi Video Berdasarkan Ukuran Teks

No	Nama media Video	Ukuran Video (byte)	Ukuran Teks Pesan Rahasia Sebelum (byte)	Waktu (miliseconds)
1	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	4	105
2	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	1.824	230
3	Sejedewe Beda Dan	4.641.161	7.296	190

	Rasa Rasta YouTube.mp4			
4	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	21.888	209
5	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	218.880	757
6	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	656.640	2012
7	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	1.313.280	3653

Tabel diatas memperlihatkan bahwa dengan media yang sama yaitu Sejedewe Beda Dan Rasa Rasta YouTube.mp4 dengan ukuran 4.641.161 byte yang relative kecil dan ukuran pesan yang berbeda mulai dari 4 byte sampai dengan 1.313.280 dapat dilakukan dengan duraksi waktu penyisipan yang tidak sama. Terkadang pesan dengan ukuran yang lebih besar membutuhkan waktu yang tidak lama dibandingkan dengan pesan yang ukuranya sangat kecil seperti percobaan no dua dan no tiga, dikarenakan pada OS Windows kondisi yang didapati adalah bahwa komputer senantiasa melakukan *multiprocessing* sekalipun itu kecil. Hal tersebut mengakibatkan inkonsistensi waktu pada proses penyisipan dan pengambilan pesan.

c) Pengujian Steganografi Video Berdasarkan Ukuran Video

Pengujian ini menggunakan ukuran pesan 1.313.280 byte pada media video dengan ukuran yang berbeda menghasilkan waktu proses yang berbeda pula sebagaiman terlihat pada tabel berikut ini:

Tabel 5 : Pengujian Steganografi Video Berdasarkan Ukuran Video

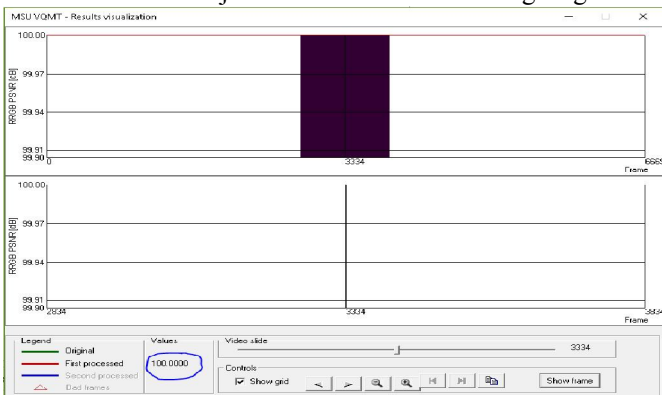
No	Nama media Video	Ukuran Video (byte)	Ukuran Teks Pesan Rahasia Sebelum (byte)	Ukuran Teks Pesan Rahasia Sesudah (byte)
1	Sejedewe Beda Dan Rasa Rasta YouTube.mp4	4.641.161	1.313.280	3772
2	Gara-Gara Rasta.wmv.mp4	9.778.298	1.313.280	3821
3	KOPI HITAM KUPU KUPU BY MOMONON.mp4	20.535.613	1.313.280	4322
4	LAGU MOTIVASI PENGHILANG GALAU__STRESS__ KANTUK.mp4	108.478.150	1.313.280	8458
5	Masanies - Pil Koplo.mp4	7.542.368	1.313.280	3928
6	roompoet hijau - dangdut jamaica (lirik).mp4	19.791.755	1.313.280	3668
7	Roompoet hijau-salah kawan.mp4	4.425.841	1.313.280	3129
8	Ceramah Mengesankan KH ANWAR ZAHID Pengajian Yang Menggemparkan Kalimantan Timur.mp4	248.063.369	1.313.280	13834

Terlihat pada tabel diatas bawha penyisipan pesan dapat dilakukan pada video Ceramah Mengesankan KH ANWAR ZAHID Pengajian Yang Menggemparkan Kalimantan Timur.mp4 dengan ukuran 248.063.369 byte dengan waktu tempuh 13834 *miliseconds* atau 13,834 seconds.

d) Pengujian Kualitas Video

- **PSNR**

Semakin besar nilai PSNR maka video hasil steganografi semakin mendekati video aslinya. Sebaliknya, semakin kecil nilai PSNR semakin jelek kualitas video hasil steganografi.

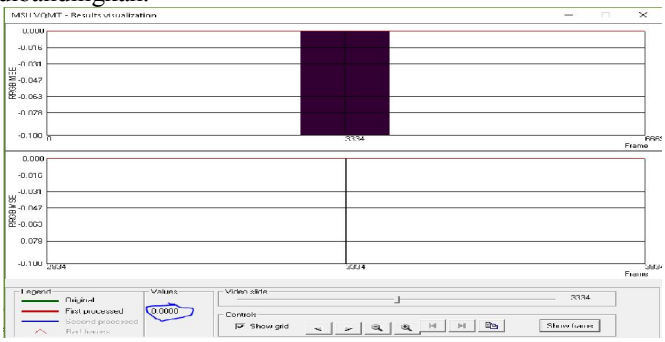


Gambar 9 : Pengujian Menggunakan PSNR

Hasil pengujian pada menggunakan perhitungan PSNR memperlihatkan bahwa mempunyai nilai 100, yang berarti bahwa kualitas video setelah disisipkan sama seperti kualitas video sebelum disisipkan.

- **MSE**

Pada perhitungan MSE semakin besar nilai MSE, maka semakin besar perbedaan antara dua buah video yang dibandingkan.

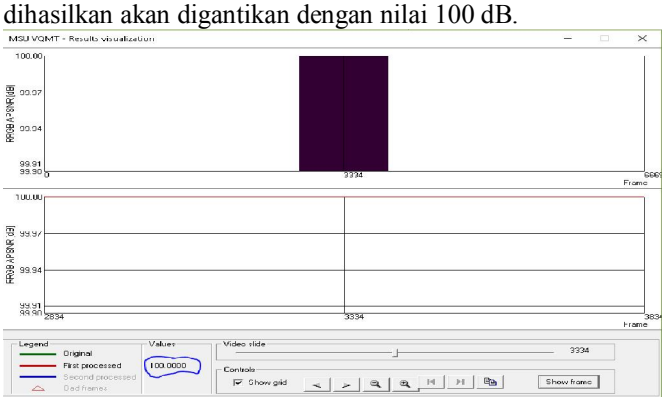


Gambar 10 Pengujian Menggunakan MSE

Perhitungan menggunakan MSE menghasilkan nilai 0, yang berarti bahwa kualitas video setelah disisipkan sama seperti kualitas video sebelum disisipkan.

- **APSNR**

Untuk menghitung kualitas video dengan APSNR, dengan cara mengambil nilai rata-rata dari PSNR setiap frame video. Jika tidak ada kerusakan pada pasangan frame video, maka sesuai penelitian oleh Chan[36], nilai tak hingga yang dihasilkan akan digantikan dengan nilai 100 dB.

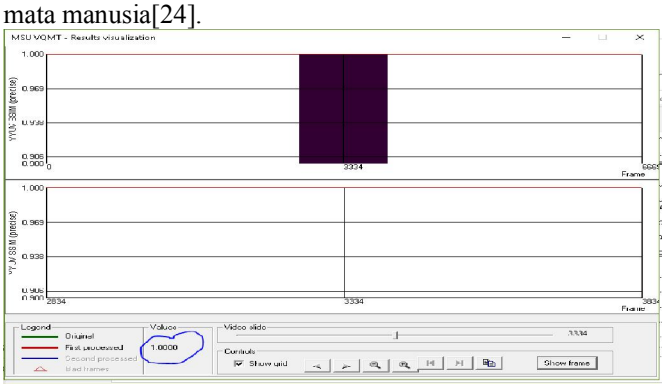


Gambar 11: Pengujian menggunakan APSNR

Pengukuran dengan menggunakan APSNR menunjukan nilai 100 dB, yang berarti tidak ada kerusakana video pada pasangan frame.

- **SSIM**

SSIM adalah metode yang digunakan untuk menghitung kesamaan antara dua gambar. Pengukuran gambar didasarkan pada gambar asli sebelum disisipi pesan atau gambar bebas distorsi sebagai referensi. SSIM dirancang sebagai perbaikan metode Peak Signal to Noise Ratio (PSNR) dan Mean Squared Error (MSE) yang terbukti tidak konsisten dengan persepsi mata manusia[24].

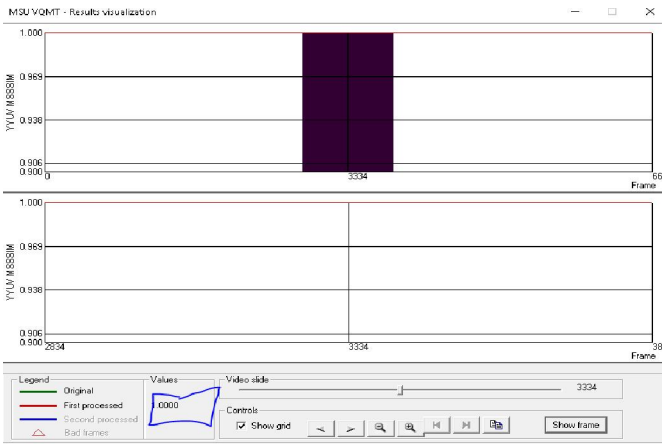


Gambar 12 : Pengujian Menggunakan SSIM

Penelitian yang dilakukan Wang et al. (2004) yang dicantumkan oleh Hariyanto (2008) menyatakan bahwa video hasil penyisipan pesan dikatakan dalam kategori baik jika nilai MSSIM yang dihasilkan lebih besar atau sama dengan 0.7 (MSSIM >= 0.7). Sebaliknya, perbedaan video hasil penyisipan akan signifikan jika hasil perhitungan MSSIM di bawah 0.7. MSSIM adalah nilai rata-rata SSIM dari pengukuran setiap frame video. Dari hasil pengujian ini menunjukkan nilai MSSIM nya adalah satu, berarti video dikategorikan baik.

- **MSSSIM**

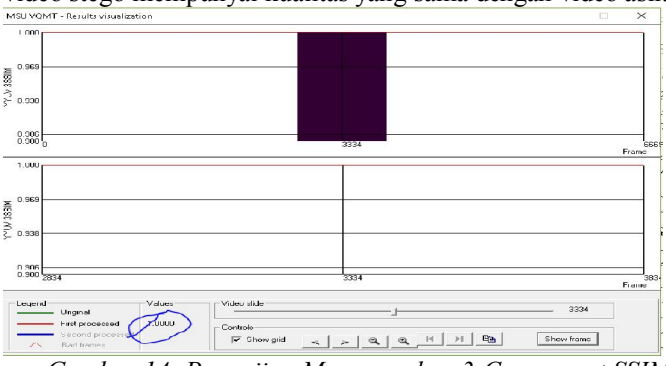
Indeks Multi-scale SSIM berdasarkan metrik SSIM dari beberapa tingkatan *downscaled* gambar aslinya[25]. Mempersepsikan rincian gambar tergantung kepadatan sampling sinyal gambar, jarak dari bidang gambar untuk pengamat, dan kemampuan perseptual dari sistem visual pengamat.



Gambar 13 : Pengujian Menggunakan MSSSIM

- **3-Component SSIM INDEX**

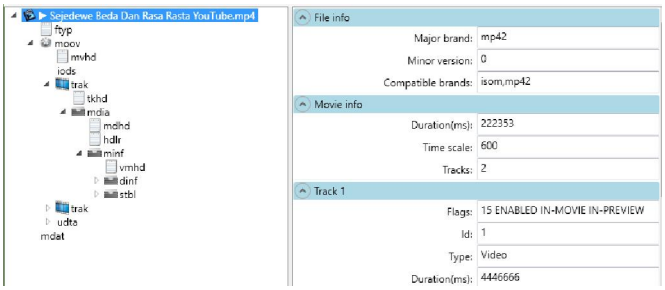
3-Komponen Indeks SSIM berdasarkan pembagian wilayah sumber frame. Ada 3 jenis daerah, yaitu : tepi, tekstur dan daerah halus. Hasil metrik dihitung sebagai rata-rata tertimbang SSIM metrik untuk daerah-daerah. Bahkan mata manusia bisa melihat perbedaan pada daerah bertekstur atau tepi tepatnya dari pada daerah halus. Pembagian berdasarkan besarnya gradien di setiap pixel dari. Hasil dari pengujian dengan 3-Component SSIM INDEX menunjukkan nilai 1, yang berarti video stego mempunyai kualitas yang sama dengan video asli.



Gambar 14: Pengujian Menggunakan 3-Component SSIM INDEX

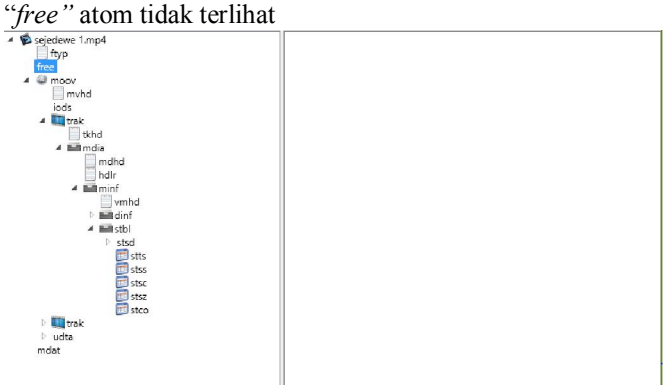
e) Spesifikasi Struktur Video

Untuk membuktikan bahwa pesan disimpan dalam "*free*" atom dapat dilihat pada gambar 15. Terlihat bahwa media video belum "*free*" atom yang berarti bahwa video tersebut belum disisipkan pesan.



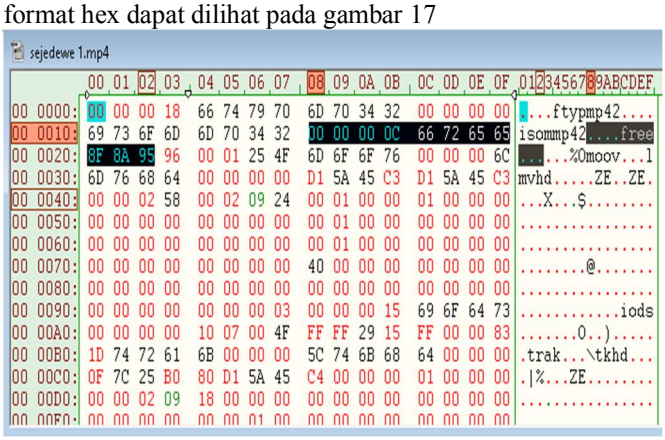
Gambar 15: Struktur file video sebelum penyisipan pesan

Sedangkan gambar 16 memperlihatkan adanya "*free"* atom yang berfungsi sebagai penampung pesan rahasia. Apabila kita melihat menggunakan program *mp4explorer,* isi pesan dari "*free"* atom tidak terlihat



Gambar 16 Struktur file video sebelum penyisipan pesan

Hasil dari penyisipan pesan rahasia jika dilihat dalam format hex dapat dilihat pada gambar 17



Gambar 17 : Format Hexadecimal Sesudah Penyisipan

2) Pengujian *Subjective*

Pengujian subyektif ditentukan berdasarkan hasil pengamatan mata manusia. Penilaian didasarkan atas karakteristik pengamatan manusia. Untuk mengetahui pengaruh terhadap wadah penampung atau stego video setelah dilakukan penyisipan pesan dapat digunakan sebuah aplikasi pemutar media digital untuk melakukan analisis sebelum dan sesudah penambahan pesan.

Gambar 18 Sebelum

Gambar 19 Sesudah

Dapat dilihat dari gambar 18 dan gambar 19 hasil video antara wadah penampung sebelum dilakukan penyisipan pesan dengan wadah penampung setelah dilakukan penyisipan pesan tidak terdapat perbedaan, kualitas gambar serta durasi pemutaran sama yaitu 26 detik

B. Analisis Hasil Pengujian

Hasil pengujian dari implementasi algoritma steganografi dengan menggunakan PNSR, APNSR, MSE, SSIM, MSSSIM, dan 3-Component SSIM INDEX, memperlihatkan bahwa kualitas gambar atau frame dari video hasil penyisipan pesan tidak mengalami penurunan kualitas. Semua itu dikarenakan pada atom "mdat" yang menyimpan samples dari audio dan video tidak mengalami perubahan sama sekali. Hasil dari pengujian obyektif dapat dilihat pada tabel di bawah ini:

Tabel 6 : Hasil dari pengujian obyektif

No	Metode	Nilai	Keterangan
1	PSNR	100	Kualitas Video Baik
2	APSNR	100	Kualitas Video Baik
3	MSE	0	Kualitas Video Baik
4	SSIM	1	Kualitas Video Baik
5	MSSSIM	1	Kualitas Video Baik
6	3-Component SSIM INDEX	1	Kualitas Video Baik

Sedangkan dengan pengujian subyektif, terlihat gambar video steganografi sama dengan gambar video asli yang dilihat dengan mata normal.

V. KESIMPULAN DAN SARAN

Hasil analisis pengujian penyisipan pesan pada video mp4 menggunaan algoritma *track free atom* berhasil dilakukan dengan baik, bahkan tanpa mempengaruhi kualitas audio dan gambar yang ada didalamnya dikarenakan atom mdat yang digunakan untuk menyimpan sample audio dan gambar tidak dirubah. Ini menyebabkan nilai pengujian menggunakan PSNR adalah 100, APNSR adalah 100, MSE adalah 0, SSIM adalah 1, MSSSIM adalah 1, dan 3-Component SSIM INDEX adalah 1. Sehingga tidak ada perubahan kualitas antara video asli dengan video steganography. Dilakukan proses enkripsi terlebih dahulu menggunakan *One's Complement* sebelum disisipkan untuk menambah kerahasiaan pesan.

Beberapa saran untuk penelitian lebih lanjut dan penyempurnaan penelitian tentang penelitian ini adalah sebagai berikut:

- Pesan sebelum disisipkan dapat dikompresi terlebih dahulu, supaya ukuran pesan yang disisipkan menjadi lebih kecil. Sehingga tidak menimbulkan kecurigaan ketika ada penambahan ukuran file video yang tidak terlalu besar.
- Inkonsistensi waktu dalam penelitian ini disebabkan karena pada OS Windows senantiasa melakukan *multiprocessing* sekalipun itu kecil dan dibutuhkannya interpreter antara aplikasi dengan sistem operasi sehingga menambah lama pada saat proses penyisipan pesan. Pada penelitian selanjutnya diharapkan dapat diimplementasikan menggunakan bahasa pemgrograman yang dapat langsung dieksekusi oleh *Operating System*
- Implementasi penyisipan pesan pada struktur komponen file pada penelitian selanjutnya dapat dikembangkan pada jenis file selain video mp4.

DAFTAR PUSTAKA

[1] S. Nidya Neyman, Lindayati, and S. Guritman, "Teknik Penyembunyian Data Rahasia pada Berkas Gambar Digital Menggunakan Steganografi Least Significant Bit Variable-Size," *Ilmu Komput. Agri-informatika*, vol. 1, no. 1, pp. 2089–6062, 2012.

[2] Andri, A. A. Lubis, A. Angkasa, and H. Angkasa, "Perancangan Aplikasi Steganografi Berbasis Matrix Pattern dengan Metode Random Blocks," *JSM STMIK Mikroskil*, vol. 16, no. 1, Apr. 2015.

[3] E. Hari Rahmawanto and C. Atika Sari, "GABUNGAN SLT-DCT UNTUK STEGANOGRAFI PENGAMANAN DATA GAMBAR PENYAKIT," *Techno.COM*, vol. 13, no. 1, pp. 38–44, Feb. 2104.

[4] K. U. Singh, "Video Steganography : Text Hiding In Video By LSB Substitution," *Kamred Udham Singh Int. J. Eng. Res. Appl.*, vol. 4, no. 5, pp. 105–108, May 2014.

[5] I. N. Piarsa, "Steganografi Pada Citra JPEG Dengan Metode Sequential Dan Spreading," *LONTAR Komput.*, vol. 2, no. 1, pp. 52–63, Jun. 2011.

[6] K. P. Divya and K. Mahesh, "Random Image Embedded in Videos using LSB Insertion Algorithm," *Int. J. Eng. Trends Technol.*, vol. 13, no. 8, pp. 381–385, Jul. 2014.

[7] P. Lucky Tirma Irawan, D. J. D. H. Santjojo, and M. Sarosa, "Implementasi Kripto-Steganografi Salsa20 dan BPCS untuk Pengamanan Data Citra Digital," *EECCIS*, vol. 8, no. 2, pp. 175–180, Dec. 2014.

[8] A. R. Lubis, M. S. Lidya, M. A. Budiman, and U. S. Utara, "Perancangan Perangkat Lunak Steganografi Audio MP3 Menggunakan Metode Least Significant Bit (LSB) Dengan Visual Basic 6 . 0," *J. DUNIA Teknol. Inf.*, vol. 1, no. 1, pp. 63–68, 2012.

[9] H. Februariyanti, "Steganografi File Audio Mp3 menggunakan Mp3Stego," *J. Teknol. Inf. Din.*, vol. XV, no. 1, pp. 57–65, Jan. 2010.

[10] M. Situmorang, D. Arisandi, and U. S. Utara, "Implementasi Steganografi Pesan Text Ke Dalam File Sound (. Wav) Dengan Modifikasi Jarak Byte Pada Algoritma Least Significant Bit (Lsb)," *J. DUNIA Teknol. Inf.*, vol. 1, no. 1, pp. 50–55, 2012.

[11] M. Yunus and A. Harjoko, "Penyembunyian Data pada File Video Menggunakan Metode LSB dan DCT," *Ijccs*, vol. 8, no. 1, pp. 81–90, Jan. 2014.

[12] A. Pratama, "Ekplorasi Penerapan Steganografi Dengan Eksploitasi Spesifikasi Format Media Container Populer," Institute Teknologi Bandung, IF3058, 2011.

[13] S. Komala, "Model Keamanan Pesan Rahasia Pada Citra Menggunakan Metode One's Complement Cryptography Dan Least Significant Bit (LSB) Steganography Di Perangkat Berbasis Android," Universitas Budi Luhur, 2013.

[14] I. Flores, *The Logic Of Computer Arithmetic*, 1st ed. Amerika: PRENTICE-HALL, INC., 1963.

[15] R. Shanthakumari and D. S. Malliga, "Video Steganography Using LSB Matching Revisited Algorithm," *IOSR J. Comput. Eng.*, vol. 16, no. 6, pp. 1–6, Dec. 2014.

[16] S. Sangareswari and V. Aruna, "Application of image hiding in mp4-video using steganography technique," *Int. J. Electr. Comput. Eng. Commun.*, vol. 1, no. 2, Apr. 2015.

[17] K. Kadam, A. Kosthi, and D. Priya, "Steganography Using Least Significant Bit Algorithm," *Dep. Comput. Eng.*, vol. 2, no. 3, 2012.

[18] Lenna, "A Complete Story of Lenna," 1997. [Online]. Available: http://www.ee.cityu.edu.hk/~lmpo/lenna/Lenna97.html. [Accessed: 20-Oct-2015].

[19] M. Alfred, *Handbook of Applied Cryptography*. Massachussets: Massachussets Institute of Techology (MIT), 1996.

[20] A.-A. G. Adnan, "Pixel Indicator Technique for RGB Image Steganography," *J. Emerg. Technol. Web Intell.*, vol. 2, 2010.

[21] Cipher, "Codes and Ciphers," 2013. [Online]. Available: http://www.braingle.com/brainteasers/codes/caesar.php. [Accessed: 20-Oct-2015].

[22] Y.-H. Yu, C.-C. Chang, and I.-C. Lin, *A New Steganography Method for Color and Grayscale Image Hiding*. Taiwan: Southern Taiwan University of Technology, 2006.

[23] Y. A. Syahbana, W. I. Yudhystira, and S. Yulina, "Algoritma Penyisipan Frame untuk Peningkatan Akurasi Metode Aligned Peak Signal-to-Noise Ratio dalam Pengukuran Kualitas Video," *J. Politek. Caltex Riau*, vol. 1, no. 2, pp. 45–56, May 2015.

[24] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[25] D. Vatolin and M. Smirnov, "MSU Quality Measurement Tool: Metrics information," *MSU Graphics & Media Lab (Video Group)*, 2015. [Online].

Available: http://www.compression.ru/video/quality_measure/info_en.html. [Accessed: 28-Oct-2015].

[26] C. Xu, X. Ping, and T. Zhang, "Steganography in compressed video stream," *Proc. First Int. Conf. Innov. Comput. Inf. Control 2006, ICICIC'06*, vol. 1, no. 11, pp. 269–272, 2006.

[27] K. V Vinodkumar and V. L. Reddy, "A Novel Data Embedding Technique for Hiding Text in Video File using Steganography," *Int. J. Comput. Appl.*, vol. 77, no. 17, pp. 13–18, Sep. 2013.

[28] J. P. Cruz, N. J. Libatique, and G. Tangonan, "Steganography and data hiding in flash video (FLV)," *TENCON 2012 IEEE Reg. 10 Conf.*, pp. 1–6, 2012.

[29] W. hasan Alwan, "Dynamic least significant bit technique for video steganography," *J. Kerbala Univ.*, vol. 11, no. 4, pp. 7–16, 2013.

[30] Maind Ms.D.S. and Sarkar Dr.B.K., "Video Steganography : an Impact of Hiding Cryptographic Data by Replacing LSB bit with Data Bit of AVI Video File," *Int. J. Adv. Res. Comput. Sci. Electron. Eng.*, vol. 1, no. 7, pp. 120–124, Sep. 2012.

[31] B. Suneetha, "Secured Data Transmission Using Video Steganographic Scheme," *M. Suresh Kumar Int. J. Eng. Res. Appl.*, vol. 4, no. 7, pp. 243–246, Jul. 2014.

[32] M. S. Kumar and G. M. Latha, "OPEN ACCESS DCT Based Secret Image Hiding In Video Sequence," *M. Suresh Kumar Int. J. Eng. Res. Appl.*, vol. 4, no. 8, pp. 5–9, Aug. 2014.

[33] P. Paulpandi, T. Meyyappan, M. Phil, M. Ba, and D. Ph, "Hiding Messages Using Motion Vector Technique In Video Steganography," *Int. J. Eng. Trends Technol.*, vol. 3, pp. 361–365, 2012.

[34] S. S. C. M. . (Ph. D. Mary, "Improved Protection in Video Steganography Used Compressed Video Bitstreams," *Improv. Prot. Video Steganography Used Compress. Video Bitstreams*, vol. 02, no. 03, pp. 764–766, 2010.

[35] A. E. Wijaya, H. Rachmawati, E. Putra, J. Teknik, I. Politeknik, and C. Riau, "Implementasi Steganografi untuk Penyembunyian Pesan pada Video dengan Metode LSB," *J. Tek. Inform.*, vol. 1, Sep. 2012.

[36] A. Chan, K. Zeng, P. Mohapatra, S.-J. Lee, and S. Banerjee, "Metrics for Evaluating Video Streaming Quality in Lossy IEEE 802.11 Wireless Networks," *Proc. IEEE INFOCOM*, pp. 1–9, Mar. 2010.