

DIGITAL SIGNATURE MENGGUNAKAN METODE SPREAD SPECTRUM SEBAGAI PERLINDUNGAN HAK CIPTA PADA CITRA DIGITAL MPEG-4

DIGITAL SIGNATURE USING SPREAD SPECTRUM METHOD AS COPYRIGHT PROTECTION IN MPEG-4 DIGITAL IMAGE

Farah Shahnaz Imami¹, Rizky Rahman J², Eka Fitrajaya Rahman³

Departemen Pendidikan Ilmu Komputer Universitas Pendidikan Indonesia
Bandung, Indonesia

¹faradorable@student.upi.edu, ²rizky_rjp@upi.edu,

³ekafitrajaya@upi.edu

Abstrak— Seiring perkembangan zaman yang semakin maju, masyarakat saat ini tidak lepas dari penggunaan gawai yang semakin canggih untuk melakukan berbagai kegiatan, salah satunya bersosialisasi menggunakan aplikasi media sosial. Berbagi berkas multimedia menjadi salah satu fitur unggulan yang ditawarkan berbagai media sosial, salah satunya adalah video. Penyebaran video tidak lepas dari perlindungan hak cipta, namun saat ini masih jarang ditemui aplikasi yang dapat melindungi video dari penyalahgunaan hak cipta. Hal tersebut dapat ditangani dengan teknologi Digital Signature menggunakan metode Steganografi Spread Spectrum, yaitu dengan menyisipkan pesan ke piksel-piksel video dan menggunakan algoritma Linear Congruential Generator agar meningkatkan kompleksitas penyebaran. Ada beberapa kriteria dalam teknik watermarking ini yaitu fidelity, recovery, dan uji ketahanan robustness. Hasil pengujian pada lima video menghasilkan rata-rata nilai PSNR sebesar 74.51714 dB yang menunjukkan bahwa video watermarked tergolong baik meski sudah disisipi pesan.

Kata Kunci: *Digital Signature, Watermarking, Steganografi, Spread Spectrum, LCG, PSNR, Video MP4.*

Abstract— As time advances, today's society cannot be separated from the use of increasingly sophisticated devices to carry out various activities, one of which is to socialize using social media applications. Sharing multimedia files is one of the excellent features offered by various social media, one of which is video. The distribution of videos cannot be separated from copyright protection, but currently there are only few applications that can protect videos from copyright abuse. This can be handled with Digital Signature technology using the Spread Spectrum Steganography method, namely by inserting messages into video pixels and using the Linear Congruential Generator algorithm to increase the complexity of the deployment. There are several criteria in this watermarking technique, namely fidelity, recovery, and robustness resistance test. The test results on five videos

produces an average PSNR value of 74.51714 dB which indicates that the watermarked video is labelled good even though the message has been inserted.

Keywords: *Digital Signature, Watermarking, Steganography, Spread Spectrum, LCG, PSNR, Video MP4.*

I. PENDAHULUAN

Digital signature atau tanda tangan digital berfungsi sebagai penanda kepemilikan dari suatu dokumen *digital*. Salah satu cara penerapannya adalah dengan menggunakan teknologi steganografi. Salah satu upaya dalam memerangi pelanggaran hak cipta dapat dilakukan dengan ilmu steganografi. Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Steganografi berasal dari Bahasa Yunani yang berarti “tulisan tersembunyi” (*covered writing*). Steganografi membutuhkan wadah penampung (suara, gambar, video, dan teks) dan juga pesan rahasia yang akan disisipi [1].

Salah satu upaya dalam memerangi pelanggaran hak cipta dapat dilakukan dengan ilmu steganografi. Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Steganografi berasal dari Bahasa Yunani yang berarti “tulisan tersembunyi” (*covered writing*). Steganografi membutuhkan wadah penampung (suara, gambar, video, dan teks) dan juga pesan rahasia yang akan disisipi [1], pesan yang disembunyikan dapat berupa teks, gambar, atau pesan lainnya.

Salah satu metode steganografi yang digunakan adalah metode *Spread Spectrum*. Metode *spread spectrum* adalah sebuah teknik pentransmisian dengan menggunakan *pseudonoise code*, yang independen terhadap data informasi, sebagai modulator bentuk gelombang untuk menyebarkan energi sinyal dalam sebuah jalur komunikasi (*bandwidth*) yang lebih besar daripada sinyal jalur komunikasi informasi. Oleh penerima, sinyal dikumpulkan kembali menggunakan replika *pseudonoise code* tersinkronisasi. Berdasarkan definisi, dapat dikatakan bahwa steganografi yang menggunakan metode *spread spectrum* memperlakukan *cover-object* baik sebagai derau (*noise*) ataupun sebagai usaha untuk menambahkan derau semu (*pseudonoise*) ke dalam *cover-object*. Secara garis besar proses penyisipan pesan menggunakan metode *spread spectrum* terdiri dari tiga proses, yaitu penyebaran, modulasi, dan penyisipan pesan ke piksel-piksel video. Sedangkan Proses ekstraksi pesan menggunakan metode *spread spectrum* juga secara garis besar terdiri dari tiga proses, yaitu pengambilan pesan dari piksel-piksel video, demodulasi, dan penyusutan pesan [2].

Agar meningkatkan kompleksitas dalam penyebaran maka dalam penelitian ini akan menggunakan metode *Linear Congruential Generator* sebagai pembangkit bilangan acak. Untuk inialisasi awal akan menggunakan kata kunci dari pengguna yang selanjutnya akan dikembangkan menjadi awal penyebaran pesan. Citra yang digunakan adalah video berformat MPEG-4 (MP4) yang mempunyai *codec* H.263. Format video ini dipilih karena selain popularitas yang tinggi di masyarakat juga mempunyai frame yang dapat disisipi teks sehingga dapat menampung lebih banyak pesan.

Steganografi juga memiliki kelemahan. Tidak seperti kriptografi, steganografi memerlukan banyak ruang untuk dapat menyembunyikan beberapa bit pesan. Akan tetapi, kelemahan ini sedikit demi sedikit dapat diatasi seiring dengan perkembangan teknik-teknik dalam melakukan steganografi.

II. PENELITIAN TERKAIT

Telah banyak penelitian mengenai penerapan steganografi dan watermarking. Penelitian yang dilakukan pun berbeda-beda. Mulai dari metode, citra yang digunakan, maupun platform aplikasi

Salah satunya adalah penelitian [4]. Penelitian tersebut telah menciptakan algoritma steganografi *spread spectrum* yang dapat digunakan secara universal baik di citra digital berbentuk gambar, video, maupun audio.

Riset lain adalah tentang pembuatan aplikasi digital watermarking pada format video MPEG-4 pada penelitian [3]. Dalam Penelitian ini format video MPEG-4 yang mempunyai *codec* H.263 dipilih karena selain popularitas yang tinggi di masyarakat juga mempunyai frame yang dapat disisipi teks sehingga dapat menampung lebih banyak pesan.

Penelitian lainnya yaitu penelitian [2]. Penelitian ini telah membandingkan metode *spread spectrum* dengan metode LSB, teknik kompresi JPEG, dan patchwork yang

menghasilkan kesimpulan bahwa metode *spread spectrum* adalah metode terbaik dilihat dari kriteria invisibility, payload capacity, robustness, independent of file format, dan unsuspecting files, tetapi mempunyai kekurangan pada kurangnya kompleksitas perhitungan. Berikutnya adalah penelitian [5] tentang penerapan random number generator LCG pada metode *spread spectrum* yang ditujukan untuk pengamanan agar penyebaran pesan semakin sulit ditebak karena adanya kunci inialisasi LCG. Penelitian-penelitian yang telah disebutkan diatas menjadi acuan penyusunan dan penelitian terhadap kecocokan metode *Spread Spectrum* pada dokumen citra berformat video MPEG-4.

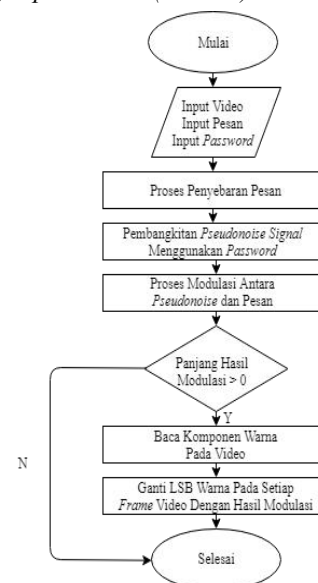
III. MODEL SISTEM VIDEO WATERMARKING SPREAD SPECTRUM

Model sistem video watermarking *spread spectrum* memiliki dua bagian penting yaitu penyisipan dan pengungkapan pesan dimana setiap bagian mempunyai beberapa tahap. Model ini merepresentasikan tahap-tahap yang ada dalam pembuatan sistem.

Selain itu, model ini juga dibuat untuk mempermudah dan meringkas keseluruhan tahap pada sistem agar mudah dimengerti jika ada yang akan mengembangkan sistem ini secara lanjut.

Gambaran model pembuatan sistem ini terdapat pada Gambar 1 untuk tahap penyisipan dan Gambar 2 untuk tahap pengungkapan. Penjelasan mengenai tahap-tahap tersebut akan dijelaskan pada sub bagian selanjutnya.

A. Proses Penyisipan Pesan (Encode)



Gambar 1. Alur model penyisipan pesan

Untuk menaruh pesan pada video tentunya harus melewati tahap penyisipan, tahap ini terdiri dari beberapa proses yaitu:

1) *Input Video, Pesan dan Password*: Pada proses ini pengguna memasukkan video berformat mp4 yang akan

disisipi pesan, selanjutnya pengguna memasukkan pesan untuk disisipi, pesan yang diinput berupa teks. Sebagai contoh teks pesan yang dimasukkan adalah 'RAHASIA', teks pesan ini kemudian diubah menjadi biner. Contoh pesan teks yang sudah diubah menjadi biner adalah seperti pada Tabel 1.

TABEL I. PERUBAHAN TEKS PESAN KE BINER

Table with 3 columns: No., Karakter, Biner. Rows: 1 R 01010010, 2 A 01000001, 3 H 01001000, 4 A 01000001, 5 S 01010011, 6 I 01001001, 7 A 01000001.

Selanjutnya biner segmen pesan yang telah didapat disebar dengan besaran skalar pengali empat sehingga akan menghasilkan segmen baru. Contohnya pada karakter 'R' yang mempunyai biner 01010010 setiap byte-nya akan dikali dengan skalar empat sehingga akan berubah menjadi 00001111000011110000000011110000 dan begitu seterusnya sampai karakter terakhir pesan. Selanjutnya pengguna harus memasukkan kata kunci yang nantinya difungsikan untuk inisialisasi pembangkit bilangan acak LCG. Contohnya pengguna memasukkan kata kunci 'SECRET', proses pemrosesan kata kunci akan dijelaskan pada poin selanjutnya.

2) Pembangkitan Pseudonoise Signal Menggunakan Passwor: Pada tahap pembangkitan pseudonoise, kata kunci 'secret' diubah terlebih dahulu ke bentuk biner lalu menggunakan perhitungan XOR pada setiap karakternya.

S = 01110011
E = 01100101
00010110 (1)

C = 01100011
01110101 (2)

R = 01110010
00000111 (3)

E = 01100101
01100010 (4)

T = 01110100
00010110 22 -> inisialisasi (5)

Nilai dari kata kunci yang telah didapatkan (22) digunakan sebagai acuan awal pembangkitan bilangan acak Linear Congruential Generator (LCG) yang mempunyai rumus perhitungan sebagai berikut:

X_{n+1} = (aX_n + C) mod (1)

a = 17, c = 7, m = 84

X_n = bilangan bulat ke- n

Perhitungannya adalah sebagai berikut :

X_1 = (17 * 22 + 7) mod 84 hasilnya X_1 = 45 (2)

X_2 = (17 * 45 + 7) mod 84 hasilnya X_2 = 16 (3)

X_3 = (17 * 16 + 7) mod 84 hasilnya X_3 = 27 (4)

Berlanjut sampai:

X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, ..., X_n (5)

Pada percobaan ini dilakukan penyebaran sebanyak enam kali sehingga menghasilkan nilai "45, 16, 27, 46, 33, 64" lalu berikutnya nilai tersebut diubah ke bentuk biner sehingga menjadi "00101101, 00010000, 00011011, 00101110, 00100001, 01000000".

3) Proses Modulasi Antara Pseudonoise dan Pesan: Setelah didapatkan segmen pesan dan pseudonoise signal selanjutnya adalah melakukan modulasi dengan menggunakan perhitungan XOR yang dijelaskan pada Tabel 2. Secara singkat, operasi XOR akan mengembalikan nilai 1 jika jumlah operand bernilai satu ganjil, jika tidak maka akan mengembalikan hasil 0.

TABEL II. PROSES MODULASI PESAN DAN PSEUDONOISE

Table with 2 columns: Segmen Pesan, Pseudonoise Signal, Hasil XOR. Rows show binary data for each category.

4) Proses Pemilihan Piksel Video: Setiap frame video mempunyai nilai piksel, pada penelitian ini hasil modulasi pesan akan disisipkan di frame-frame acak pada header atau footer video lalu sistem akan melakukan pembacaan pada piksel-piksel yang akan disisipi pesan. Sebagai contoh, dari beberapa frame acak diambil sampel tiga puluh enam bit dan sistem akan melakukan pembacaan piksel sebagai berikut:

TABEL III. PEMILIHAN PIKSEL YANG AKAN DISISIPI PESAN

Table containing a grid of binary values representing selected pixels for message insertion.

Tabel diatas adalah representasi bit piksel pada frame video yang diambil secara acak oleh sistem untuk selanjutnya diolah untuk disisipi pesan.

5) Penyisipan Hasil Modulasi Pesan: Tahap terakhir yaitu menyisipkan hasil modulasi ke piksel-piksel yang telah dipilih oleh sistem sehingga akan menghasilkan segmen baru sebagai Tabel 4 berikut ini:

TABEL IV
PENYISIPAN PESAN KE PIKSEL YANG TELAH TERPILIH

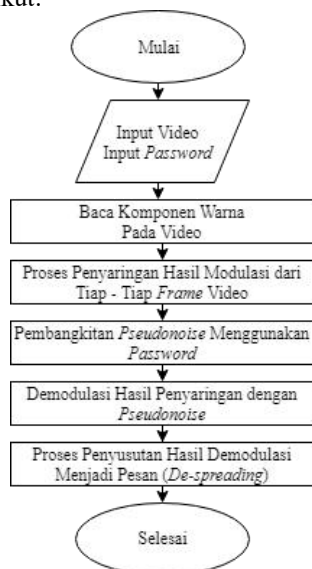
10111000	10111010	10111011	10111010	10111010
10111000	10110111	10110110	10111010	10111000
10111000	10111001	10101001	10101011	10101011
10101011	10101000	10101000	10101010	10101101
10101011	10101100	10101111	10101011	10000001
10000011	01111100	01111011	01111001	01110101
01111001	01110010	01110010	01110100	01110011
01101110

Dan seterusnya sampai semua *byte* pesan telah disisipi

Tabel diatas menunjukkan bahwa pada beberapa *byte-byte* terakhir bit pesan mengalami perubahan yang diakibatkan oleh penyisipan pesan. *Byte-byte* yang telah berubah ini akan disaring kembali pada saat proses pengungkapan pesan

B. Proses Pengungkapan Pesan (Decode)

Setiap citra yang telah disisipi pesan harus dapat diungkap kembali agar memenuhi syarat *recovery*, proses pengungkapan pesan (Gambar 2) akan dijelaskan pada poin-poin berikut:



Gambar 2. Alur model pengungkapan pesan

1) *Input Video dan Password*: Pada proses ekstraksi, langkah-langkah yang dilakukan merupakan kebalikan dari proses penyisipan pesan. Pertama, user harus memilih video yang akan diekstrasi. Sama seperti proses encode, video yang dipilih pada proses decode harus berformat mp4. Selanjutnya user harus memasukkan kembali kata kunci yang sebelumnya telah dibuat pada proses encode. Pada tahap encode sebelumnya, kata kunci yang dimasukkan adalah ‘SECRET’ yang kemudian harus diubah menjadi bentuk biner. Selanjutnya kata kunci yang telah diubah menjadi biner kembali diproses menjadi pseudonoise signal dengan mengoperasikan fungsi XOR pada tiap karakter sehingga menghasilkan angka inialisasi untuk proses pembangkitan bilangan acak LCG.

2) *Proses Pembacaan Piksel Video & Penyaringan Pesan*: Video yang telah dipilih oleh pengguna akan diperiksa terlebih dahulu oleh sistem apakah pada body dan footer video telah disisipi pesan atau belum. Dengan menggunakan acuan algoritma penyebaran hasil modulasi pada proses penyisipan maka sistem akan mengetahui dimana piksel-piksel yang telah disisipi segmen pesan untuk selanjutnya dilakukan penyaringan. Pada contoh sebelumnya sebuah video yang telah disisipi pesan akan diungkap kembali, setelah pengguna memasukkan password yang benar maka sistem membaca video tersebut seperti Tabel 5 berikut ini:

TABEL V

PEMBACAAN PESAN PIKSEL VIDEO SESUAI PASSWORD

10111000	10111010	10111011	10111010	10111010	10111000
10110111	10110110	10111010	10111000	10111000	10111001
10101001	10101011	10101011	10101011	10101000	10101000
10101010	10101101	10101011	10101100	10101111	10101011
10000001	10000011	01111100	01111011	01111001	01110101
01111001	01110010	01110010	01110100	01110011	01101110
.

Dan seterusnya

Selanjutnya pada piksel-piksel yang telah dibaca akan dilakukan penyaringan sehingga didapatkan *byte-byte* sebagai berikut:

```

00100010000111110001101111011110001011100100000
0000000000001111000011110000000011110000000000
00000111100000000000000000000111100001111000011
11000000001111111100001111000000001111000000001
111000011110000000000000000000001111
    
```

3) *Proses Demodulasi Hasil Penyaringan Pesan*: Setelah didapatkan bit-bit hasil penyaringan langkah selanjutnya adalah proses demodulasi, yaitu mengembalikan hasil modulasi menjadi bentuk asalnya. Untuk prosesnya, bit-bit hasil penyaringan didemodulasi dengan pseudonoise signal dari kata kunci yang telah diubah ke bentuk biner menggunakan perhitungan XOR sebagai berikut:

TABEL VI

PROSES DEMODULASI HASIL PENYARINGAN DAN PSEUDONOISE SIGNAL

Hasil Penyaringan	001000100001111100011011110111100 0101110010000000000000001111000 011110000000111100000000000000 11110000000000000000000111100001 1110000111100000001111111000011 1100000001111000000011110000111 100000000000000000001111
Pseudonoise Signal	00101101000100000011011001011100 010000101000000
Hasil XOR	00001111000011110000000111100000 0001111000000000000000001111000 011110000000111100000000000000 11110000000000000000000111100001 1110000111100000001111111000011 1100000001111000000011110000111 1000000000000000000001111

4) *Proses Penyusutan Pesan Hasil Demodulasi*: Sebelumnya pada saat proses penyebaran (pesan, hasil dari konversi teks pesan diubah ke bentuk biner lalu disebar dengan besaran skalar pengali empat. Maka pada proses penyusutan pesan, hasil demodulasi segmen pesan disusutkan dengan besaran skalar pembagi empat sehingga menghasilkan segmen sebagai berikut:

01000001 01001000 01000001 01010011 01001001
01000001

Hasil penyusutan diatas merupakan pesan yang sama ketika user memasukkan pesan untuk disembunyikan pada proses encode sebelumnya. Langkah terakhir yaitu mengonversikan biner tersebut menjadi bentuk teks sehingga terbentuk kata-kata 'RAHASIA' dan menampilkannya pada program. Bila user salah memasukkan kata kunci maka demodulasi tidak dapat dilakukan karena ketidakcocokan pada saat proses penyaringan bit-bit hasil modulasi, jika hal itu terjadi maka program hanya menampilkan kata acak sehingga tidak terjadi kebocoran pesan

IV. SKENARIO EKSPERIMEN

Penelitian ini menggunakan empat skenario eksperimen. Skenario-skenario tersebut dilakukan dengan menggunakan lima video berformat mp4 yang mempunyai ukuran file, durasi, dan jumlah *frame* yang berbeda-beda. Terdapat skenario yang membutuhkan bantuan manusia untuk memberikan penilaian secara kasat mata. Skenario yang dibuat yaitu:

A. Skenario Faktor Fidelity

Skenario pertama yang dibuat adalah skenario faktor *fidelity* yang bertujuan untuk membuktikan kualitas video sebelum dan sesudah disisipi pesan. Pada skenario ini akan dilakukan perhitungan MSE (*Mean Squared Error*), PSNR (*Peak Signal to Noise Ratio*), dan pengujian terhadap 15 responden untuk memberikan pendapat apakah ada perbedaan secara kasat mata atau tidak.

B. Skenario Faktor Recovery

Skenario ke dua yang dibuat adalah skenario faktor *recovery*. Skenario ini merupakan skenario untuk mengetahui apakah video yang telah disisipi pesan dapat diungkap kembali atau tidak. Pada skenario ini akan dilakukan dua pengujian yaitu mengungkap pesan dengan kata kunci yang benar dan kata kunci yang salah.

C. Skenario Faktor Robustness

Skenario ke tiga yang dibuat adalah skenario faktor *robustness* untuk mengetahui apakah video yang telah disisipi pesan akan tahan dengan beragam manipulasi citra dan serangan seperti *cropping*, rotasi, peningkatan kontras dan lain sebagainya.

D. Skenario Waktu Proses Penyisipan dan Ekstraksi

Skenario terakhir atau skenario ke empat yang dibuat yaitu pengujian terhadap waktu proses penyisipan dan

ekstraksi. Pada skenario ini terdapat dua indikator yang akan dianalisis yaitu:

- Pengujian waktu proses penyisipan dan ekstraksi berdasarkan jumlah panjang pesan
- Pengujian Waktu Proses Penyisipan dan Ekstraksi Berdasarkan Jumlah *Frame* Pada Video

V. HASIL DAN PEMBAHASAN

A. Analisis Faktor Fidelity

Fidelitas (*fidelity*) adalah tingkat keakuratan dalam memproduksi gambar atau suara. Dalam penelitian ini, faktor *fidelity* merupakan penentu apakah citra video yang telah disisipi pesan menjadi berubah kualitasnya atau terlihat ada perbedaan secara kasat mata setelah disisipi pesan.

Untuk mendukung pemenuhan faktor *fidelity* yang pertama dilakukan adalah mengadakan percobaan pada 15 orang untuk melihat secara kasat mata apakah ada perbedaan secara visual pada saat menonton video sebelum dan sesudah disisipi pesan. Hasilnya 15 dari 15 orang tidak merasakan perbedaan pada kedua video yang telah diperlihatkan.

Untuk menguji faktor *fidelity* lainnya dilakukan dua perhitungan yaitu menghitung MSE (*Mean Square Error*) yaitu nilai error kuadrat rata-rata antara citra asli dengan citra manipulasi (dalam hal ini citra yang telah disisipi pesan) semakin kecil nilai MSE maka semakin sedikit error yang dihasilkan dari kedua citra, maka semakin baik juga kualitasnya., dan menghitung PSNR (*Peak Signal Noise Ratio*) yaitu perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau (*noise*) yang berpengaruh pada sinyal tersebut [6]. Derau yang dimaksud adalah kerusakan pada piksel tertentu pada sebuah citra sehingga mempengaruhi kualitas dari piksel suatu citra. PSNR menggunakan satuan decibel (dB) sementara MSE tidak menggunakan satuan apapun. Rumus perhitungan MSE dan PSNR adalah sebagai berikut:

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N (S_{xy} - C_{xy})^2 \quad (1)$$

$$PSNR = 10 \log_{10} \left(\frac{C_{max}^2}{MSE} \right) \quad (2)$$

Keterangan:

x, y : koordinat citra

M, N : dimensi citra

S_{xy} : citra yang telah disisipi pesan

C_{xy} : citra asli











C_{max}² : nilai maksimum piksel diantara semua citra

Setelah dilakukan penghitungan, semakin kecil nilai MSE pada citra video yang telah diberi watermark maka akan semakin baik faktor *fidelity*-nya, sedangkan faktor *fidelity* juga akan semakin baik apabila PSNR yang dihasilkan pada citra video yang telah diberi watermark mempunyai nilai yang semakin besar. Sebuah citra

tergolong baik apabila PSNR yang dihasilkan diatas 30 dB [7].

Hasil pengujian MSE dan PSNR pada lima video ditunjukkan pada tabel dibawah ini:

TABEL VII
HASIL PERHITUNGAN MSE DAN PSNR

No.	Original Image	Watermarked Image
1.	 rock.mp4	 rock_encoded.mp4
MSE: 0.0018504 PSNR: 75.4581 dB		
2.	 hotwheels.mp4	 hotwheels_encoded.mp4
MSE: 0.0017101 PSNR: 75.8007 dB		
3.	 nature.mp4	 nature_encoded.mp4
MSE: 0.005638 PSNR: 70.6195 dB		
4.	 car.mp4	 car_encoded.mp4
MSE: 0.0018273 PSNR: 75.5128 dB		
5.	 bunny.mp4	 bunny_encoded.mp4
MSE:0.0019661 PSNR: 75.1946 dB		

Dalam perhitungannya, MSE dan PSNR yang dihasilkan mempunyai nilai yang baik, dari percobaan pada tabel diatas rata-rata PSNR yang dihasilkan adalah 74.51714 dB. Berdasarkan hasil rata-rata tersebut, citra

yang dihasilkan menyerupai aslinya sehingga faktor *fidelity* menjadi terpenuhi.

B. Analisis Faktor Recovery

Sebuah citra yang telah disisipkan pesan harus dapat diungkap kembali (*recovery*) agar dapat mengetahui pesan di dalamnya. Pada pengujian ini dilakukan percobaan pada lima video untuk mengetahui apakah video yang telah disisipkan pesan dapat diungkap kembali dengan kata kunci yang benar, apakah video dapat diungkap kembali dengan kata kunci yang salah. Tabel 8 membuktikan bahwa kelima video dapat diungkap menggunakan kata kunci yang benar dan tidak dapat diungkap jika menggunakan kata kunci yang salah

TABEL VIII
HASIL UJI RECOVERY

Nama Berkas	Kata Kunci 'SECRET'	Kata Kunci 'WRONG'
rock_encoded.mp4	√	X
hotwheels_encoded.mp4	√	X
nature_encoded.mp4	√	X
car_encoded.mp4	√	X
bunny_encoded.mp4	√	X

C. Analisis Faktor Robustness

Kriteria steganografi yang baik juga harus tahan terhadap serangan dan manipulasi terhadap citra (*robustness*) seperti manipulasi rotasi, *cropping*, *mirroring*, dan penajaman kontras. Setelah dilakukan manipulasi terhadap citra yang telah disisipi pesan hasilnya adalah sebagai Tabel 9 berikut ini:

TABEL IX
HASIL UJI ROBUSTNESS

Nama Berkas	R	C	M	K
rock_encoded.mp4	X	X	X	X
hotwheels_encoded.mp4	X	X	X	X
nature_encoded.mp4	X	X	X	X
car_encoded.mp4	X	X	X	X
bunny_encoded.mp4	X	X	X	X

Keterangan:

R = manipulasi rotasi

C = *cropping*

M = *mirroring*

K = penajaman kontras

Berdasarkan pengujian *robustness* pada tabel diatas terungkap bahwa hasil citra yang telah disisipi pesan belum tahan terhadap serangan dan manipulasi, ini disebabkan karena citra yang dilakukan manipulasi akan secara signifikan mengubah nilai pikselnya sehingga pesan tidak dapat diungkap kembali.

D. Analisis Waktu Proses Penyisipan dan Ekstraksi

1) Waktu Proses Penyisipan Dan Ekstraksi Berdasarkan Jumlah Panjang Pesan: Hasil pengujian

waktu proses penyisipan dan ekstraksi berdasarkan jumlah panjang pesan dapat dilihat pada Tabel 10 berikut.

TABEL X
HASIL PENGUJIAN WAKTU PROSES PENYISIPAN DAN EKSTRAKSI
BERDASARKAN PANJANG PESAN

No.	rock_encoded.mp4		
	Panjang Pesan	Embedding (second)	Extracting (second)
1	25	0.28351	0.21162
2	50	0.35154	0.23095
3	100	0.42472	0.23981
4	200	0.58498	0.24578

Berdasarkan pengujian diatas terlihat bahwa semakin panjang karakter pesan yang akan disisipkan maka semakin lama pula waktu yang dibutuhkan untuk menyisipkan dan mengekstraksi pesannya.

2) Waktu Proses Penyisipan dan Ekstraksi Berdasarkan Jumlah Frame Pada Video: Hasil pengujian waktu proses penyisipan dan ekstraksi berdasarkan jumlah frame pada video dapat dilihat pada Tabel 11 berikut:

TABEL XI
HASIL PENGUJIAN WAKTU PROSES PENYISIPAN DAN EKSTRAKSI
BERDASARKAN JUMLAH FRAME

Watermarked Video	Jumlah Frame	Embedding (second)	Extracting (second)
rock_encoded.mp4	3257	0.28351	0.21162
hotwheels_encoded.mp4	3945	0.31805	0.24934
nature_encoded.mp4	375	0.072972	0.051606
car_encoded.mp4	3339	0.20892	0.17696
bunny_encoded.mp4	1440	0.6346	0.13922

Pada hasil percobaan diatas terlihat bahwa jumlah frame mempengaruhi namun tidak signifikan. Contohnya, watermarked video berjudul 'rock_encoded.mp4' yang mempunyai jumlah frame 3257 mempunyai waktu embedding lebih lambat daripada watermarked video berjudul 'car_encoded.mp4' yang mempunyai jumlah frame lebih banyak yaitu 3339. Hal ini dipengaruhi juga oleh kondisi performa perangkat lunak yang dipakai untuk menjalankan aplikasi.

VI. KESIMPULAN

Berdasarkan hasil dan pembahasan pada eksperimen yang telah dilakukan, sistem digital signature menggunakan steganografi watermarking metode spread

spectrum berhasil diimplementasikan dengan baik pada video berformat mp4. Pengguna harus memasukkan kata kunci saat menyisipkan pesan yang bertujuan untuk meningkatkan kompleksitas perhitungan pembangkit bilangan acak, lalu saat akan mengungkap pesan pengguna harus memasukkan kata kunci yang benar agar pesan dapat terungkap dan terhindar dari kebocoran pesan. Kualitas hasil video yang dihasilkan terbilang cukup baik karena menghasilkan PSNR diatas 30 dB yaitu dengan rata-rata 74.51714 dB Waktu proses penyisipan dan pengungkapan bergantung pada jumlah panjang pesan, semakin sedikit karakter pesan maka semakin cepat prosesnya. Jumlah frame pada video tidak terlalu berpengaruh pada waktu proses penyisipan dan pengungkapan. Namun sayangnya hasil watermarked video yang terkena manipulasi atau serangan tidak dapat diungkap kembali karena perubahan nilai piksel yang signifikan

Untuk penelitian berikutnya, perlu dikembangkan sistem steganografi yang tahan akan serangan dan manipulasi sehingga pesan dalam video tetap dapat dijaga dalam keadaan apapun. Kedepannya sistem steganografi ini diharapkan mampu mencakup lebih banyak format video yang populer seperti MKV, WEBM, dan lain sebagainya. Selanjutnya diharapkan sistem steganografi ini dapat terintegrasi dengan kamera gawai yang dapat memroses hasil video secara real-time sehingga mengurangi celah pelanggaran hak cipta.

REFERENSI

- [1] Mumir, R., Steganografi dan Watermarking, Bandung: Institut Teknologi Bandung, 2006.
- [2] Vembrina, Y.G., Spread Spectrum Steganography, Bandung: Institut Teknologi Bandung, 2006.
- [3] Herlinawati, Steganografi Video H263 Dengan Metode Discrete Cosine Transform. Jurnal Rekayasa dan Teknologi Elektro Vol.11, No.3, 2017.
- [4] Cox, I. J., Kilian, J., Leighton, T., & Shamon, T. "Secure Spread Spectrum Watermarking for Multimedia". NEC Research Institute, Technical Report, p 95 – 10, 1997.
- [5] Aprianto, H, Penerapan Lcg Pada Spread Spectrum Menggunakan Ruang Warna Hsv Pada Citra Digital, Biltek Vol 13 Teknik Informatika Sekolah Tinggi Teknik Harapan Medan., 2017.
- [6] Male, G. M., Wirawan, & Setijadi, E., Analisa Kualitas Citra Pada Steganografi untuk Aplikasi e-Government Prosiding Seminar Nasional Manajemen Teknologi XV, p 4. 2017.
- [7] Irawan, P. L., Santjojo, D. D., & Sarosa, M., Implementasi Kripto-Steganografi Salsa20 dan BPCS untuk Pengamanan Data Citra Digital., Jurnal EECCIS Vol. 8, No. 2, p 175-180, 2014..