

Peningkatan Kapasitas Penyisipan *Audio Data Hiding* Berbasiskan Modifikasi Metode *Least Significant Digit*

Tegar Palyus Fiqar¹, Tohari Ahmad²

Abstract— Nowadays, the needs of data security is proportional to the amount of data usage as the effect from easiness in internet access. There are many ways to protect data in order to avoid data leakage, such as cryptography, watermarking, and data hiding. The latter is conducted by hiding secret data into a carrier. A popular method in data hiding is Least significant digit (LSD). The hiding is carried out by changing the value of binary secret data into decimal in specific segment. Then, the decimal value of secret data is inserted into the carrier. The process in LSD method is relatively simple and easier to implement, but the LSD has weakness in pivoting point as reference of data changes. This results in a change into the original data carrier. Additionally, this also has limited amount of data capacity that can be inserted.

On the other hand, Global-Section Search method (GSS) is an optimization method used to find an optimum value between two signals. Newly acquired signal GSS is further used as a pivoting point and slot area to paste the data secret.

This study proposes a new method of data hiding by using Least Significant Digit and Global-section search (LSD-GSS) to increase the capacity of secret data insertion. The performance of LSD-GSS as proposed method is tested by comparing the LSD method and using 15 data sets of audio in various genres as carrier media and various large capacity of 12 secret data. SNR is used to find out the quality of hidden data audio. The experimental result shows that the LSD-GSS has increased the capacity of secret data that can be inserted without changing the original data carrier.

Intisari— Saat ini, kebutuhan keamanan data berbanding lurus dengan jumlah pengguna data sebagai efek kemudahan mengakses internet. Banyak cara memberikan perlindungan data agar tidak terjadi kebocoran data, antara lain kriptografi, watermarking, dan data hiding. Data hiding dilakukan dengan menyembunyikan data rahasia ke dalam media pembawa. Metode yang populer dalam data hiding ialah *Least Significant Digit* (LSD). Metode penyembunyian ini diawali dengan mengubah data rahasia biner menjadi data rahasia desimal dengan segmen tertentu. Selanjutnya, nilai desimal dari data rahasia disisipkan pada media pembawa. Proses metode LSD cenderung relatif sederhana dan mudah diterapkan, tetapi LSD memiliki kelemahan pada *pivoting point* sebagai acuan perubahan data. Hal ini berakibat terjadinya perubahan data asli media pembawa. Selain itu, LSD memiliki keterbatasan pada besarnya jumlah kapasitas data yang dapat disisipkan.

¹Mahasiswa, Jurusan Teknik Informatika Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Kampus ITS Sukolilo, Kota Surabaya, Jawa Timur 60111 (tlp: 031-5939214; fax: 031-5913804; e-mail: tegar.pf@gmail.com)

²Dosen, Jurusan Teknik Informatika Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Kampus ITS Sukolilo, Kota Surabaya, Jawa Timur 60111 (tlp: 031-5939214; fax: 031-5913804; e-mail: tohari@if.its.ac.id)

Di sisi lain, metode *Global-Section Search* (GSS) merupakan metode optimasi yang digunakan untuk mencari suatu nilai optimum antara dua sinyal. Sinyal baru yang diperoleh GSS selanjutnya digunakan sebagai *pivoting point* dan *slot area* untuk menyisipkan data rahasia.

Makalah ini mengusulkan metode baru data hiding menggunakan *Least Significant Digit* dan *Global-Section Search* (LSD-GSS) untuk meningkatkan kapasitas penyisipan data rahasia. Ujicoba kinerja metode usulan LSD-GSS dibandingkan dengan metode LSD menggunakan 15 data set berbagai genre sebagai media pembawa dan 12 data rahasia yang bervariasi besar kapasitasnya. SNR digunakan untuk mengetahui kualitas hasil penyembunyian data audio. Hasil menunjukkan LSD-GSS telah berhasil meningkatkan kapasitas data rahasia yang dapat disisipkan tanpa mengubah data asli media pembawa.

Kata Kunci— Keamanan Informasi, Penyisipan Data, Proteksi Data, *Least Significant Digit*, *Golden-Section Search*.

I. PENDAHULUAN

Pada era globalisasi, internet menjadi penghubung komunikasi antara satu pihak dengan pihak lain. Kemudahan yang diberikan oleh penyedia layanan internet membuat transaksi lalu lintas data semakin meningkat. Pengguna internet di dunia tumbuh secara eksponensial dalam 20 tahun terakhir. Hal ini ditunjukkan dari laporan lembaga *International Telecommunication Union* (ITU). Laporan pada tahun 2000 menunjukkan pengguna internet hanya 6,8 % dari populasi dunia. Pada tahun 2016, jumlah pengguna internet jauh meningkat menjadi sebesar 46,1% [1]. Pertukaran lalu lintas data yang dilakukan berupa berkas teks, berkas citra, berkas video, maupun berkas audio. Berkas-berkas ini ada yang bersifat publik dan bersifat rahasia. Untuk berkas yang bersifat public, setiap orang dapat mengaksesnya, sedangkan berkas yang bersifat rahasia tidak semua orang dapat mengaksesnya. Namun, terkadang ada pihak yang tidak berkepentingan mencoba untuk mengakses dan selanjutnya menyalahgunakan berkas data yang bersifat rahasia [2]. Oleh karenanya, diperlukan suatu mekanisme perlindungan atau keamanan data rahasia. Ada berbagai macam mekanisme keamanan data yang disesuaikan dengan peruntukan perlindungan datanya, antara lain *watermarking* digunakan untuk perlindungan data digital dengan cara diberi penanda [3]; *cryptography* memiliki cara kerja mengubah makna dengan bantuan *key* tertentu untuk membuka pesannya [4]; *biometric security* memberikan perlindungan terhadap data digital anggota tubuh seperti retina dan sidik jari [5]; serta *steganography* memiliki mekanisme menyisipkan pesan rahasia pada media pembawanya [6].

Watermarking bertujuan untuk melindungi data yang memiliki hak kekayaan intelektual agar terhindar dari

pembajakan ataupun penduplikatan berkas tersebut. *Watermarking* sering digunakan pada berkas teks, berkas citra, berkas audio, maupun berkas video [7]. *Watermarking* memiliki dua bagian penting, yaitu berkas asli dan berkas penanda. *Watermarking* berkerja dengan cara menandai berkas asli dengan berkas penanda. Adapun metode *watermarking* yang dikembangkan antara lain *phase coding*, yaitu mereduksi sinyal dan selanjutnya mengganti dengan penanda [8]; *echo hiding* hanya digunakan pada media pembawa audio dengan memanfaatkan *delay* sinyal yang dihasilkan [9]; *perceptual coding* memampatkan ruang sinyal untuk penyisipan [10]; *quantization index modulation* [11]; *cepstrum domain* menyisipkan pesan pada beberapa komponen *cepstral* [12]; *patchwork* berkerja dengan mengubah nilai koefisien sinyal kemudian menggantinya dengan data sisipan [13]; dan *spread spectrum* pengembang dari metode *least significant bit* (LSB) [7].

Steganography sering disebut juga sebagai *data hiding*, karena *steganography* berkerja menyembunyikan data dengan cara menyisipkan berkas rahasia pada berkas pembawa atau *carrier*. Berkas pembawa dan berkas rahasia dapat berupa teks, citra, audio, maupun video. Banyak metode *data hiding* yang berkembang. Hal ini disebabkan oleh tantangan dari *data hiding* ialah menyisipkan data rahasia dengan kapasitas yang besar tanpa merusak kualitas dari media pembawa yang telah disisipi data rahasia [14]. Pada umumnya, banyak penelitian yang berfokus pada peningkatan kuantitas saja atau kualitas saja. Namun, tidak dipungkiri ada juga penelitian yang dikembangkan untuk meningkatkan dua hal tersebut. Khususnya untuk penelitian yang terkait dengan media pembawanya, berupa berkas audio, ada berbagai macam metode yang dapat digunakan dalam proses penyisipan data pada berkas audio, antara lain LSB; *echo hiding*, dilakukan dengan menyisipkan satu bit data rahasia pada sinyal *echo* [15]; *phase coding*, yaitu penyisipan dengan mengubah sinyal ke domain frekuensi menggunakan *discrete fourier transform* (DFT) [16]; *spread spectrum*, yaitu penyisipan dengan proses modulasi yang diawali dengan proses enkripsi pada pesan rahasia [17]; *tone insertion*, yang menggunakan *tone* rendah untuk ruang penyisipan [18]; dan LSD, yaitu hasil *sampling* sinyal disisipkan berupa nilai desimal [19].

LSB dianggap sebagai metode yang pertama kali digunakan untuk proses penyembunyian informasi data rahasia [9]. Metode ini merupakan metode yang paling sederhana dari berbagai macam metode yang dikembangkan untuk penyembunyian informasi data. Metode ini dapat diterapkan pada berbagai media pembawa. Metode ini berkerja menyisipkan data rahasia pada media pembawa dengan cara mengubah kedua media menjadi bentuk biner. Selanjutnya, bit terakhir dari sekumpulan bit media pembawa digantikan dengan bit dari biner data rahasia. Metode ini memiliki keunggulan, antara lain waktu pemrosesan metode ini sangat cepat, tidak ada proses komputasi yang kompleks, dan sederhana. Di sisi lain, metode ini memiliki kelemahan, antara lain mudah dideteksi penyisipannya serta kapasitas penyisipannya sangat rendah.

Berikutnya, metode *phase coding* bekerja dengan memilih fase dari komponen sinyal yang dihasilkan (spektrum sinyal asli), lalu komponen sinyal ini akan digantikan dengan data

rahasia [20]. Komponen fase sinyal yang menyimpan data rahasia diharapkan memiliki sinyal yang lebih kecil dibandingkan sinyal asli atau sinyal pembawa. Hal ini bertujuan agar hasil audio penyisipan tidak terlalu banyak berubah dari audio asli. Keuntungan metode ini adalah tahan terhadap sinyal distorsi dibandingkan dengan metode lainnya. Metode ini juga tahan terhadap kompresi. Di sisi lain, metode ini memiliki kelemahan, yaitu sangat rentan terhadap *low pass filter* dan kapasitas penyisipannya sangat sedikit.

Pada *spread spectrum*, salah satu metode penyisipan data yang tidak tergantung pada sinyal asli, pesan rahasia disisipkan sepanjang frekuensi *spectrum* audio sinyal [17]. Metode ini memiliki beberapa media yang dibutuhkan, antara lain media pembawa, data rahasia, dan sinyal *chip*. Sinyal *chip* digunakan sebagai penanda antara media pembawa dan data rahasia. Pada tahap pertama, media dimodulasi dengan *chip* sinyal. Hasil dari proses ini dimodulasi kembali dengan data rahasia. Setelah itu, diperoleh media baru yang telah disisipkan dengan data rahasia. Pada bagian *decoding*, media gabungan dimodulasi dengan *chip* sinyal, selanjutnya dilakukan filter dengan *bandpass filter* dan *phase detector*.

Metode *tone insertion* memanfaatkan keuntungan dari keterbatasan kemampuan pendengaran manusia, yaitu nada-nada rendah yang tidak mampu didengarkan oleh manusia akan menjadi ruang penyisipan data rahasia [18]. Secara garis besar, sinyal terbagi menjadi dua bagian. Sinyal audio rendah digunakan untuk ruang penyisipan data rahasia dan sinyal audio tinggi digunakan untuk ruang sinyal audio asli. Keuntungan dari metode ini adalah audio hasil sisipan dan audio asli terdengar sangat mirip. Metode ini memiliki kelemahan yaitu teknik ini kurang terjamin keamanannya.

LSD merupakan salah satu metode penyisipan data yang dikembangkan dari metode LSB. LSD mempunyai sistem kerja yang sama dengan LSB. Perbedaan terletak pada tipe data yang disisipkan dari media pembawa. LSB menyisipkan data dalam bentuk bit biner, sedangkan pada LSD berbentuk *integer* [21]. LSD sendiri memiliki kekurangan, yaitu pada kualitas yang dihasilkan. Ini disebabkan oleh penyisipan LSD yang dilakukan pada data aslinya, sehingga diperlukan suatu usulan baru membuat skema *pivoting point* dengan GSS untuk menghasilkan sinyal baru yang selanjutnya penyisipan dilakukan pada sinyal GSS, bukan pada sinyal asli audio. Usulan baru ini akan meningkatkan kualitas dari hasil sisipan dan secara tidak langsung kapasitas penyisipan pun akan meningkat.

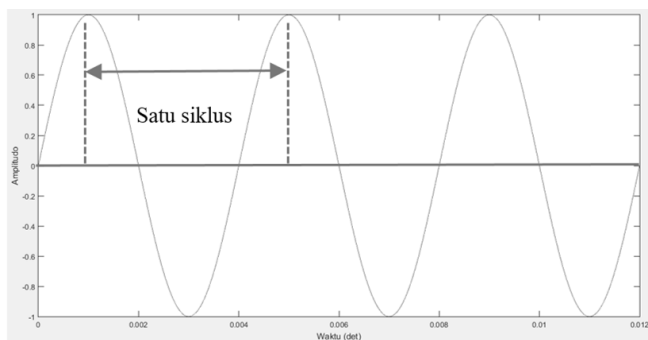
II. METODE PENYEMBUNYIAN DATA

Pada bagian ini dijelaskan beberapa teori terkait penelitian yang dilakukan. Beberapa topik bahasan antara lain konsep dasar berkas audio, konsep dasar mengenai *data hiding* dan konsep teori metode LSD, serta yang terakhir GSS, yaitu metode yang digunakan sebagai penunjang usulan penelitian.

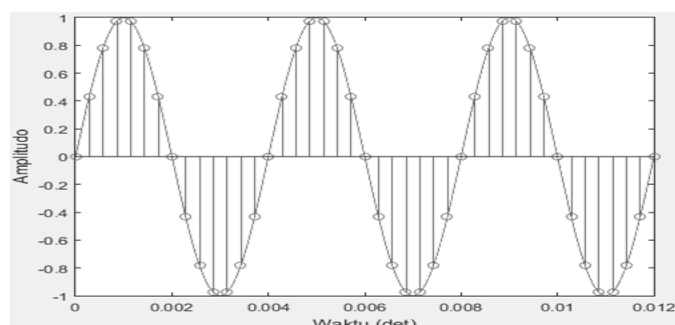
A. Berkas Audio

Audio merupakan representasi dari gelombang analog suara. Gelombang suara membawa informasi yang berasal dari sumbernya. Pola sinus merupakan bentuk dari gelombang

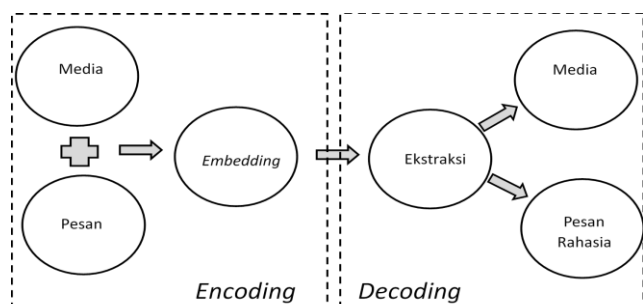
suara. Gelombang ini terdiri atas beberapa bukit dan beberapa lembah yang beresilasi terus menerus, seperti ditunjukkan pada Gbr. 1. Jarak antara puncak bukit dan lembah disebut sebagai amplitude. Gelombang yang dibaca setiap satuan detik disebut sebagai frekuensi dengan satuan Hertz(Hz).



Gbr. 1 Representasi gelombang suara.



Gbr. 2 Representasi proses *sampling*.



Gbr. 3 Konsep proses *data hiding*.

Komputer memerlukan proses untuk mengolah gelombang suara. Proses ini bertujuan agar data dapat diinterpretasikan sehingga maksud dari pesan yang dibawa dapat dipahami. Proses yang diperlukan untuk mengolah gelombang ini dikenal dengan *analog to digital*. Komputer hanya mengenali gelombang dalam bentuk digital, oleh karena ini dibutuhkan *pulse code modulation* (PCM) untuk proses mengubah sinyal analog menjadi digital. Pada proses PCM, sinyal analog dikuantisasi nilainya. *Sampling* merupakan proses pencuplikan sinyal analog menjadi digital, seperti tampak pada Gbr. 2. Pada proses pembacaan berkas audio, dikenal juga istilah *sampling rate*, yaitu proses pembacaan banyaknya *sample* berkas audio dalam kurun waktu satu detik. Berkas audio memiliki *sampling rate* sebesar 44.100 Hz. Besarnya

nilai masing-masing *sample* berkas audio yaitu 16 bit atau 2 byte.

B. Data Hiding

Steganography (*data hiding*) ialah salah satu ilmu bidang keamanan data informasi yang memiliki prinsip kerja dengan menyembunyikan data rahasia (*payload*) yang disisipkan pada media pembawa. Tujuan dari *data hiding* adalah agar orang tidak menyadari ada suatu pesan rahasia dalam suatu media. Pada *steganography* ada dua bagian penting, yaitu data pembawa atau sering disebut dengan *host* atau media *carrier* dan pesan rahasia itu sendiri. Media *carrier* yang digunakan dapat berupa berkas audio, berkas citra, maupun berkas video.

Proses penyisipan pada *steganography* dikenal sebagai proses *encoding*, sedangkan proses ekstraksi pesan dikenal dengan proses *decoding*. Pada *steganography*, selalu diupayakan agar ukuran berkas setelah penyisipan tidak jauh berbeda dengan ukuran berkas sebelumnya. Hal ini bertujuan agar orang lain tidak menyadari bahwa ada perubahan pada suatu berkas. Tantangan dalam *data hiding* yaitu kualitas hasil sisipan selalu berbanding terbalik dengan jumlah kapasitas data yang dapat disisipkan. Konsep *data hiding* ditunjukkan pada Gbr. 3.

C. Least Significant Digit (LSD)

Sebuah penelitian telah dilakukan, dengan fungsi utama dari algoritme yang dikembangkan adalah pengkodean data piksel citra dengan menggunakan algoritme LSD [21]. Algoritme LSD digunakan dengan memilih piksel tertentu yang akan digunakan sebagai ruang dari data rahasia (*payload*).

Pada penelitian tersebut, masing-masing piksel memiliki nilai integer dari *layer* (RGB). Nilai integer inilah yang akan menjadi pembawa. Proses penyisipan nilai integer ini dilakukan tanpa pengolahan data biner. Nilai integer yang diperoleh dari piksel berada pada *range* 0 hingga 255 di setiap *layer*. Data rahasia akan diproses dari kode ASCII yang diperoleh. Nilai dari ASCII ini berbentuk biner, yang selanjutnya diubah ke dalam nilai integer, sehingga antara media pembawa dan *payload* data rahasia sudah berada dalam domain yang sama.

Pada tahapan selanjutnya, data rahasia yang berbentuk integer ditambahkan dengan nilai piksel dari media pembawa. Sebagai contoh, citra memiliki RGB=[251 034 093], sedangkan data rahasia yang awalnya bernilai 111 dalam biner diubah menjadi 7. Selanjutnya dilakukan pemilihan. Dari *layer* tersebut, dipilih nilai yang paling kecil, yaitu 034 pada *layer* G. Maka akan diperoleh piksel baru bernilai 041, sehingga citra yang baru akan memiliki nilai RGB=[251 041 093].

Dalam makalah ini, metode LSD digunakan dalam proses penyisipan media pembawa audio. Proses LSD sendiri mengalami perubahan dan penyesuaian untuk diterapkan pada berkas audio. Hal ini disebabkan karena karakteristik berkas citra dan berkas audio berbeda. Perubahan ini dijelaskan pada bab metode usulan.

D. Golden-Section Search (GSS)

Dalam kasus penyelesaian akar-akar dari sebuah persamaan *nonlinear*, tujuan yang hendak dicapai adalah menemukan nilai dari suatu variabel x dari sebuah fungsi $f(x)$. *Single-variable optimization* memiliki tujuan untuk menemukan nilai x tersebut dari *extremum*. *Extremum* ini memiliki salah satu nilai saja, nilai maksimum atau nilai minimum dari sebuah fungsi.

GSS merupakan sebuah metode yang sederhana untuk memperoleh suatu nilai variabel x dari sebuah fungsi $f(x)$. Nilai x diperoleh dari batasan dua nilai, yaitu nilai *lower* (x_l) dan nilai *upper* (x_u). Selanjutnya, nilai akar dari variabel x yang dicari disebut x_r . Nilai x_r berada pada nilai x_l dan x_u . Dari dua variabel x_l dan x_u , maka fungsi yang digunakan ialah $f(x_l)$ dan $f(x_u)$. Selanjutnya, nilai akar persamaan dapat diestimasi dengan nilai yang berada di tengah (*midpoint*) dari *interval* kedua fungsi tersebut. Persamaan yang digunakan untuk memperoleh nilai tersebut adalah (1).

$$x_r = \frac{x_l + x_u}{2} \quad (1)$$

Pada tahapan akhir dilakukan iterasi dari nilai *interval* yang telah ditentukan.

Makalah ini mengadopsi (1) untuk memperoleh sinyal baru yang selanjutnya disebut sebagai sinyal GSS. Sinyal GSS ini menjadi ruang untuk penyisipan data *payload*. Nilai x_l didefinisikan sebagai sinyal *sampling* audio x_n dan x_u didefinisikan sebagai sinyal x_{n+1} , dengan nilai n adalah indeks dari *sampling* sinyal audio.

III. METODE USULAN

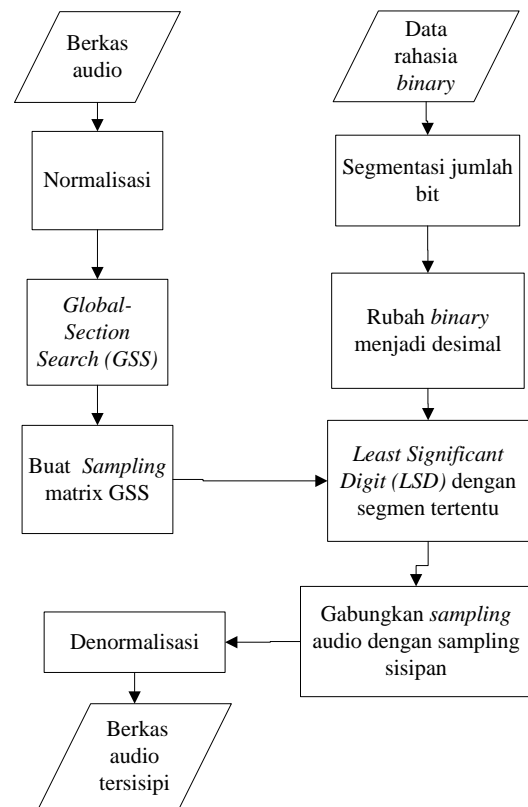
Terdapat dua bagian utama yang terpenting dalam proses *data hiding* audio yang diusulkan, yaitu proses *encoding* dan *decoding*. Proses *encoding* dikenal juga dengan proses penyisipan data *payload* pada media pembawa, dalam hal ini berkas audio. Sebelum proses *encoding*, ada tahapan *preprocessing*, yang merupakan bagian awal pada tahapan usulan penelitian. Tahapan ini merupakan tahapan persiapan pengolahan data *payload*. Proses *decoding* juga dikenal dengan proses ekstraksi data *payload* dari media pembawa. Salah satu bagian proses *decoding* yang merupakan bagian akhir pada tahapan usulan penelitian yaitu *post processing*. Tahap ini mengolah data biner yang telah diekstrak menjadi teks.

A. Preprocessing Payload

Pada tahap pertama dijelaskan tentang data rahasia yang digunakan. Adapun tahapannya adalah sebagai berikut.

1. Masukan data *payload* berbentuk karakter teks, dan selanjutnya disimpan dalam format *.txt*.
2. Menghitung jumlah karakter yang digunakan sebagai data rahasia. Pada contoh, karakter yang digunakan sebanyak satu karakter, yaitu huruf 'i'.
3. Setelah perhitungan karakter, selanjutnya kakater tersebut diubah menjadi biner ASCII, sehingga diperoleh '01101001'.

Setelah bagian akhir, data *payload* yang berbentuk biner ini menjadi masukan data pada bagian *encoding*.



Gbr. 4 Skema usulan *data hiding*.

B. Penyisipan (Encoding)

Tahap kedua merupakan inti proses penyisipan data rahasia, menjelaskan tahapan mengenai skema penyisipan yang diusulkan. Peningkatan kapasitas dilakukan dengan cara meningkatkan ruang penyisipan pada sinyal audio. Tahapan dari proses ini digambarkan seperti Gbr. 4. Adapun tahap prosesnya adalah sebagai berikut.

1. Masukan yang digunakan adalah berkas audio. Pada makalah ini, berkas audio yang digunakan berekstensi *.wav*. Hal ini karena berkas ini tidak mengalami kompresi, sehingga tidak ada *sampling* sinyal audio yang hilang.
2. Tahapan normalisasi mengubah nilai *sampling* dari berkas audio sesuai dengan karakteristik audio 16 bit, sehingga dilakukan pengolahan nilai *sampling* tersebut dari nilai input *range* -1 dan 1 bertipe data *double* menjadi unit 16 yang memiliki *range* 0 hingga 65.535.
3. Pada tahapan GSS, dibuat sinyal baru dari sinyal audio yang telah ternormalisasi. Sebagai contoh sinyal masukan audio seperti berikut.

$$x_1 = 80, x_2 = 100, x_3 = 80, x_4 = 100, x_5 = 80$$

dengan x_n adalah nilai *magnitude* sinyal audio asli dan n adalah indeks dari sinyal audio. Proses memperoleh sinyal baru GSS dilakukan dengan mengacu pada (1).

$$\bar{x}_1 = \frac{80 + 100}{2}, \quad \bar{x}_1 = 90$$

$$\bar{x}_2 = \frac{100 + 80}{2}, \quad \bar{x}_2 = 90$$

$$\bar{x}_3 = \frac{80 + 100}{2}, \quad \bar{x}_3 = 90$$

$$\bar{x}_4 = \frac{100 + 80}{2}, \quad \bar{x}_4 = 90$$

dengan \bar{x}_n merupakan *sampling* baru data audio yang dihasilkan dengan GSS dan n adalah indeks dari sinyal GSS.

4. Pada tahapan *sampling* GSS dibuat sebuah vektor yang terdiri atas *sampling* sinyal GSS dengan menggunakan penyusunan seperti (2).

$$[\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_n, \dots]. \quad (2)$$

Dengan menerapkan (2), maka hasil yang diperoleh adalah sebagai berikut.

$$[90, 90, 90, 90]$$

Selanjutnya, vektor ini yang akan menjadi ruang penyisipan data *payload*.

5. Pada tahapan memasukkan data *stream payload* dilakukan pembacaan data bit *payload* yang telah diproses pada tahapan *preprocessing*. Diperoleh bit data *stream payload* yaitu '01101001'.
6. Segmen bit data rahasia. Setelah pembacaan bit *payload*, dilakukan pembuatan segmen data rahasia (k). Besarnya nilai k dapat ditentukan sesuai kebutuhan dan harus konsisten dalam menggunakan nilai tersebut. Pada penelitian, nilai (k) dicoba antara 1 hingga 8. Sebagai contoh, berikut penggunaan nilai ($k = 2$ dan $k = 4$).

$$k = 2, \text{seg}_2^1 = 01, \text{seg}_2^2 = 10, \text{seg}_2^3 = 10, \text{seg}_2^4 = 01$$

$$k = 4, \text{seg}_4^1 = 0110, \text{seg}_4^2 = 1001.$$

Notasi seg_k^n merupakan notasi pembentukan segmen data *payload*, dengan k adalah nilai pembentuk segmen tersebut dan n adalah indeks segmen.

7. Perubahan nilai segmen data rahasia biner menjadi desimal. Hal ini bertujuan agar *payload* dapat disisipkan pada nilai *sampling* GSS. Sebagai contoh, dengan menggunakan segmen ($k = 2$) diperoleh

$$\text{seg}_2^1 = 01 \rightarrow \text{seg}_2^1 = 1, \text{seg}_2^2 = 10 \rightarrow \text{seg}_2^2 = 2$$

$$\text{seg}_2^3 = 10 \rightarrow \text{seg}_2^3 = 2, \text{seg}_2^4 = 01 \rightarrow \text{seg}_2^4 = 1$$

8. Penyisipan LSD pada sinyal hasil GSS digunakan sebagai *pivoting point* untuk mengganti *location map*. Tahapan ini melakukan penyisipan pada digit terkakhir dari sinyal *sampling* audio asli dan *sampling* GSS. Penentuan digit terakhir dilakukan menggunakan (3).

$$[x_n, \bar{x}_n] \rightarrow [x_n, \bar{x}_n]. \quad (3)$$

Notasi \bar{x}_n merupakan sinyal sisipan, \bar{x}_n , diperoleh dari penjumlahan antara \bar{x}_n dan seg_k^n (dalam desimal), sehingga diperoleh (4).

$$\bar{x}_n = \bar{x}_n + \text{seg}_k^n. \quad (4)$$

Dari (4) diperoleh

$$\bar{x}_1 = 90 + 1, \bar{x}_2 = 90 + 2,$$

$$\bar{x}_3 = 90 + 2, \bar{x}_4 = 90 + 1$$

Maka dengan mengacu pada (3), tahapan LSD menjadi seperti berikut.

$$[80, 90] \rightarrow [80, 91],$$

$$[100, 90] \rightarrow [100, 92],$$

$$[80, 90] \rightarrow [80, 92],$$

$$[100, 90] \rightarrow [100, 91]$$

9. Penggabungan sinyal audio asli dengan sinyal GSS yang telah disisipi data *payload* membuat sebuah vektor yang terdiri atas *sampling* sinyal audio dan *sampling* sinyal GSS yang telah disisipkan dengan menggunakan penyusunan seperti (5).

$$[x_1, \bar{x}_1, x_2, \bar{x}_2, x_3, \bar{x}_3, x_4, \bar{x}_4, x_5, \bar{x}_n, x_n, \dots, \dots]. \quad (5)$$

Diperoleh sebuah vektor baru yang tersusun dari *sampling* sinyal gabungan antara *sampling* audio asli dan *sampling* GSS yang telah disisipkan. Jika diterapkan dengan (5), hasil yang diperoleh adalah sebagai berikut.

$$[80, 91, 100, 92, 80, 92, 100, 91, 80]$$

10. Denormalisasi, yaitu proses pengembalian tipe data yang dilakukan terhadap hasil dari tahap 9.

11. Perubahan pada *frame rate* audio menjadi dua kali dari *frame rate* sebelumnya. Hal ini terjadi karena gabungan sinyal ini memiliki jumlah *sampling* data menjadi $2 * (m - 1)$, dengan m adalah jumlah *sampling* sinyal audio.

C. Ekstraksi (Decoding)

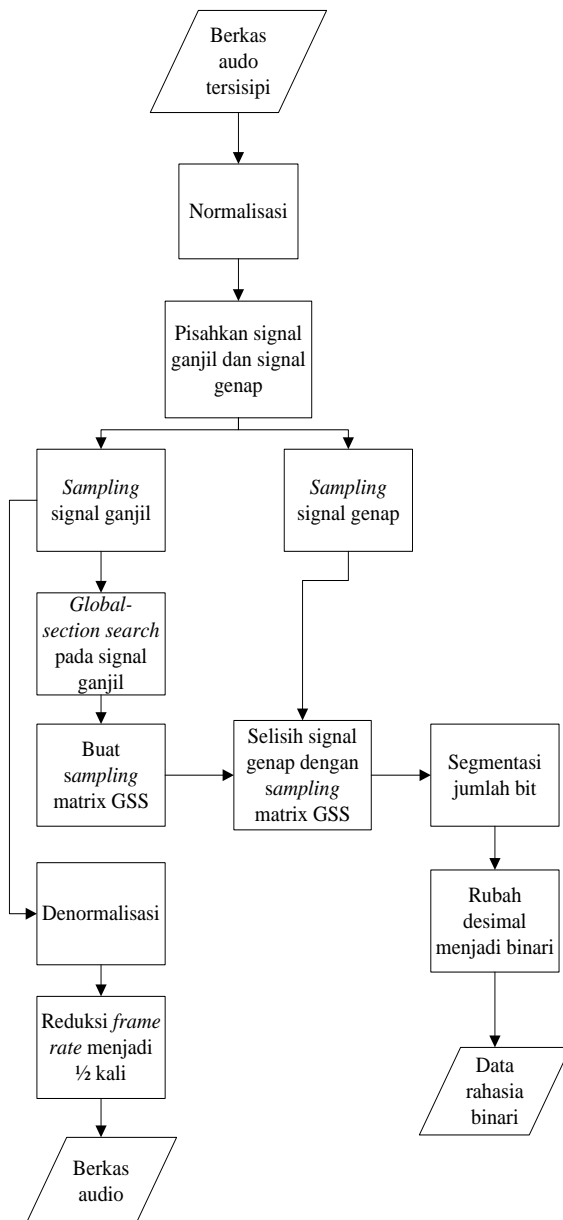
Bagian ini menjelaskan tentang rancangan metode ekstraksi pesan yang diusulkan dan proses *recovery* berkas audio. Masukan dari proses ini adalah *stego-audio*. Keluaran dari proses *decoding* berupa dua jenis berkas, yaitu berkas audio dan berkas *payload* yang berbentuk biner. Berkas biner diolah pada *post processing* yang prosesnya menjelaskan perubahan data hingga kembali menjadi berkas *.txt*. Penjelasan tahap *decoding* ini sesuai dengan Gbr. 5. Tahapan-tahapan ini adalah sebagai berikut.

1. Tahap pertama yaitu masukan *stego-audio* berformat *.wav*.
2. Pembacaan berkas dilanjutkan dengan normalisasi data. Proses normalisasi ini sama dengan proses normalisasi *encoding*.
3. Tahapan ketiga dari bagian *decoding* yaitu melakukan pemisahan sinyal audio. Sinyal ganjil merupakan sinyal audio asli (x_n) sedangkan sinyal genap ialah sinyal yang telah disisipi data *payload* (\bar{x}_n). Berikut ini merupakan contoh pemisahan yang dilakukan.

$$\text{stego - audio} \rightarrow [80, 91, 100, 92, 80, 92, 100, 91, 80]$$

$$\text{Sampling signal ganjil} \rightarrow [80, 100, 80, 100, 80]$$

$$\text{Sampling signal genap} \rightarrow [91, 92, 92, 91]$$



Gbr. 5 Skema ekstraksi data payload dan audio asli.

4. *Sampling* ganjil dipersiapkan untuk tahapan 12, yang tahap selanjutnya proses denormalisasi. Proses ini dikarenakan sinyal berindeks ganjil merupakan audio dari media pembawa.
5. *Sampling* sinyal berindeks genap digunakan untuk mengekstraksi antara sinyal sisipan dan sinyal GSS. Sinyal ini mengandung dua arti, yaitu sinyal dari proses GSS dan data *payload*. Sinyal ini pun menjadi *pivoting* penentu bagian-bagian sinyal yang diisi oleh data *payload*. Jika sinyal berindeks genap sama dengan sinyal GSS, maka sinyal itu tidak mengalami perubahan. Hal ini bermakna tidak ada penyisipan yang dilakukan pada sinyal tersebut.
6. GSS dilakukan pada sinyal asli berindeks ganjil. Proses ini bertujuan untuk memperoleh kembali sinyal GSS yang tanpa sisipan, karena sinyal ini menjadi penentu

dan menjadi parameter ekstraksi untuk data *payload*. Proses ini sama dengan proses ketiga tahap *encoding*.

$$\bar{x}_1 = \frac{80 + 100}{2}, \quad \bar{x}_1 = 90$$

$$\bar{x}_2 = \frac{100 + 80}{2}, \quad \bar{x}_2 = 90$$

$$\bar{x}_3 = \frac{80 + 100}{2}, \quad \bar{x}_3 = 90$$

$$\bar{x}_4 = \frac{100 + 80}{2}, \quad \bar{x}_4 = 90.$$

Persamaan yang digunakan mengacu pada (1).

7. Membuat *sampling* matriks dari sinyal GSS dengan menggunakan (2), sehingga diperoleh hasil sebagai berikut.

$$[90, 90, 90, 90].$$

8. Mencari selisih antar sinyal *stego-audio* berindeks genap hasil dari tahap 3 dengan sinyal GSS yang diperoleh dari tahapan ke-6. Mengacu pada (6), persamaan yang digunakan untuk pemisahan ini adalah seperti berikut.

$$\Delta\bar{x}_n = \bar{x}_n - \bar{x}_n. \quad (6)$$

Notasi $\Delta\bar{x}_n$ merupakan hasil dari pengurangan kedua sinyal. Notasi ini bermakna sama dengan seg_k^n . Dari persamaan tersebut, untuk indeks 1 diperoleh $\Delta\bar{x}_1 = 91 - 90$. Dengan mengacu pada (6), diterapkan untuk sinyal-sinyal berikutnya, sehingga diperoleh berturut-turut $\Delta\bar{x}_1 = 1$, $\Delta\bar{x}_2 = 2$, $\Delta\bar{x}_3 = 2$, $\Delta\bar{x}_4 = 1$.

9. Setelah diperoleh selisih, akan dipersiapkan segmen jumlah bitnya. Saat ini diperoleh data *payload* dalam bentuk decimal, yaitu [1, 2, 2, 1].
10. Hasil dari segmen dalam desimal diubah ke biner sesuai dengan segmen yang telah ditentukan. Karena pada proses *encoding* digunakan $k = 2$, maka proses *decoding payload* dikonversi menjadi 2 bit biner.
11. Data *payload* disusun dari hasil sinyal pertama hingga sinyal terakhir. Diperoleh data rahasia dari tahap sebelumnya sebagai berikut.

$$[1, 2, 2, 1] \rightarrow [01 \ 10 \ 10 \ 01].$$

12. Terhadap sinyal ganjil hasil dari pemisahan dilakukan denormalisasi menjadi bertipe *double*.
13. Reduksi *frame rate* menjadi setengah dari *frame rate* yang terbaca. Hal ini karena data audio menjadi setengah dari audio *stego-audio*.

D. Post-processing Payload

Data rahasia yang diekstraksi masih berbentuk biner, sehingga perlu diproses kembali menjadi data yang berbentuk teks. Adapun tahapannya adalah sebagai berikut.

1. Data rahasia atau *payload* berbentuk berkas biner, berisi sekumpulan bit-bit data rahasia. Pada langkah pertama ini dilakukan *load* data dengan membuka berkas tersebut. Masukan tahapan ini diperoleh dari tahapan *decoding*. Pada kasus ini, data rahasia yang digunakan berupa data biner '01101001'.

2. Tahapan kedua adalah melakukan perhitungan jumlah bit yang tersimpan, dalam hal ini terdapat 8 bit.
3. Tahapan ketiga yaitu mengubah bit tersebut menjadi karakter. Dari hasil konversi bit tersebut, diperoleh karakter huruf 'i'.

IV. UJI COBA DAN ANALISIS

Dalam makalah ini digunakan masukan berkas audio dengan jenis *wave audio format* dan berekstensi *.wav*. Pemilihan berkas audio *.wav* dikarenakan berkas ini tidak mengalami proses kompresi, sehingga diharapkan tidak ada berkas yang hilang, baik sebelum maupun sesudah dilakukan penyisipan berkas.

Data set yang digunakan yaitu *Data set For Instrument Recognition In Musical Audio Sinyals (IRMAS)* [22]. Berkas audio dapat diunduh pada alamat url www.mtg.upf.edu/download/dataset.

Berkas audio yang digunakan sebanyak 15 data, terdiri atas lima instrumen dan masing-masing instrumen terdiri atas tiga genre, seperti ditampilkan pada Tabel I. Durasi berkas audio yang digunakan pada masing-masing jenis musik adalah selama 3 detik. Karakteristik audio seperti ditunjukkan pada Tabel II. Sedangkan berkas yang digunakan untuk data *payload* merupakan berkas teks berekstensi *.txt*. Data ini diperoleh dari generator teks. Laman www.id.lipsum.com merupakan penyedia dari generator teks.

Data *payload* yang digunakan bervariasi, sebanyak 12 jenis. Masing-masing jenis berbeda ukurannya, dari 1 kB, 5 kB, 10 kB, 15 kB, 20 kB, 25 kB, 30 kB, 35 kB, 40 kB, 45 kB, 50 kB, dan 55 kB.

Beberapa skenario ujicoba pada penelitian yaitu skenario pertama dengan menghitung kapasitas data *payload* yang mampu ditampung oleh media pembawa, skenario kedua dengan menentukan besarnya segmen (*k*) *payload* yang ideal untuk dilakukan penyisipan, dan skenario ketiga dengan menggunakan berkas audio yang bervariasi, sesuai dengan genre dan instrumen yang digunakan.

Evaluasi yang digunakan pada makalah ini adalah dengan mengukur kualitas yang dihasilkan dari *stego-audio*.

Pengukuran ini membandingkan antara dua audio, yaitu berkas audio yang telah disisip data rahasia dan berkas audio asli. Pengukuran dilakukan untuk memperoleh nilai kualitas yang dihasilkan, sehingga digunakan *peak sinyal to noise ratio* (PSNR), seperti ditunjukkan pada (7).

$$PSNR(s_o, s_w) = 10 \log_{10} \frac{\sum_n [s_o(n)]^2}{\sum_n [s_w(n) - s_o(n)]^2} \quad (7)$$

Notasi $PSNR(s_o, s_w)$ merupakan *peak sinyal to noise ratio*, dengan s_o adalah sinyal asli dari *cover* berkas audio, dan s_w adalah sinyal dari hasil sisipan berkas audio. $\sum_n [s_o(n)]^2$ merupakan total nilai kuantisasi dari seluruh data sinyal asli sebanyak n data yang dikuadratkan, sedangkan $\sum_n [s_w(n) - s_o(n)]^2$ merupakan jumlah selisih dari masing-masing data berkas audio yang telah disisipkan dan berkas audio asli yang dikuadratkan.

Hasil dan pembahasan dibagi menjadi tiga subbagian, yaitu pengaruh tingkat keberhasilan penyisipan terhadap besarnya

penentuan segmen yang digunakan terhadap besarnya kapasitas penyisipan pada berkas audio, *trend* yang dihasilkan dengan menggunakan data audio 1 dan berbagai variasi besarnya data *payload*, serta perbandingan kualitas dari *stego-audio* yang diusulkan dengan audio yang asli.

TABEL I
INFORMASI DATA SET

Instrumen	Genre	Data	Nama
Cello (cel)	Country-Folk (cou_fol)	[cel][jaz_blu]0011_2	Data 1
	Classical (cla)	[cel][cla]0007_1	Data 2
	Pop-Rock (pop-rock)	[cel][pop_roc]0060_2	Data 3
Acoustic guitar (gac)	Country-Folk (cou_fol)	[gac][jaz_blu]0549_2	Data 4
	Classical (cla)	[gac][cla]0530_1	Data 5
	Pop-Rock (pop-rock)	[gac][pop_roc]0560_3	Data 6
Piano (pia)	Country-Folk (cou_fol)	[pia][jaz_blu]1348_1	Data 7
	Classical (cla)	[pia][cla]1291_1	Data 8
	Pop-Rock (pop-rock)	[pia][pop_roc]1306_1	Data 9
Saxophone (sax)	Country-Folk (cou_fol)	[sax][jaz_blu]1605_2	Data 10
	Classical (cla)	[sax][cla]1598_1	Data 11
	Pop-Rock (pop-rock)	[sax][pop_roc]1588_1	Data 12
Human singing voice (voi)	Country-Folk (cou_fol)	[voi][jaz_blu]2358_1	Data 13
	Classical (cla)	[voi][nod][cou_fol]2442_1	Data 14
	Pop-Rock (pop-rock)	[pop_roc]2547_3	Data 15

TABEL II
KARAKTERISTIK DATA SET

No.	Karakteristik	Keterangan
1	Data Pembawa	Data 1 ~ Data 15
2	Jumlah Kanal	1 Buah
3	Sampling Rate	44100
4	Jumlah Sampling	132299
5	Durasi Waktu	~ 3 Detik
6	Bit Per Sample	16 bit

A. Tingkat Keberhasilan Penyisipan

Pada skenario dilakukan pengujian terhadap berkas audio yang selanjutnya disisipkan data yang bervariasi dari 1 kb hingga 55 kb. Pengujian ini diterapkan pada kedua metode, yaitu metode usulan dan metode LSD. Tabel III menunjukkan keberhasilan penyisipan dan kegagalan dalam proses penyisipan dari metode yang diusulkan, sedangkan Tabel IV menunjukkan penyisipan pada metode LSD. Dari skenario ini, ada 96 kali pengujian dengan parameter segmen dan besar data *payload*.

TABEL III
HASIL PENYISIPAN LSD-GSS

		Data Sisipan											
		1	2	3	4	5	6	7	8	9	10	11	12
Segmen	1	V	V	V	V	X	X	X	X	X	X	X	X
	2	V	V	V	V	V	V	V	X	X	X	X	X
	3	V	V	V	V	V	V	V	V	V	V	X	X
	4	V	V	V	V	V	V	V	V	V	V	V	V
	5	V	V	V	V	V	V	V	V	V	V	V	V
	6	V	V	V	V	V	V	V	V	V	V	V	V
	7	V	V	V	V	V	V	V	V	V	V	V	V
	8	V	V	V	V	V	V	V	V	V	V	V	V

Keterangan : (V) Penyisipan Behasil (X) Penyisipan Gagal

TABEL IV
HASIL PENYISIPAN METODE LSD

		Data Sisipan											
		1	2	3	4	5	6	7	8	9	10	11	12
Segmen	1	V	V	X	X	X	X	X	X	X	X	X	X
	2	V	V	V	X	X	X	X	X	X	X	X	X
	3	V	V	V	V	X	X	X	X	X	X	X	X
	4	V	V	V	V	V	X	X	X	X	X	X	X
	5	V	V	V	V	V	V	X	X	X	X	X	X
	6	V	V	V	V	V	V	V	X	X	X	X	X
	7	V	V	V	V	V	V	V	V	X	X	X	X
	8	V	V	V	V	V	V	V	V	V	X	X	X

Keterangan : (V) Penyisipan Behasil (X) Penyisipan Gagal

Pada metode usulan, seperti ditunjukkan pada Tabel III, dari 96 kali pengujian diperoleh 81 pengujian dapat dilakukan penyisipan. Sebanyak 15 pengujian tidak dapat dilakakn penyisipan. Kegagalan penyisipan ini disebabkan oleh ruang penyisipan yang terbatas. Dari segmen ($k = 1$) penyisipan hanya dapat dilakukan dari *payload* 1 kB – 15 kb dan selebihnya tidak dapat disisipkan. Hal ini disebabkan karena banyak kapasitas maksimum data yang dapat ditampung, sebesar 132.298 bit. Pada segmen ($k = 2$), penyisipan hanya dapat dilakukan dari *payload* 1 kB – 30 kb dan selebihnya tidak dapat disisipkan. Hal ini disebabkan karena banyak kapasitas data yang dapat ditampung, sebesar 264.596 bit. Pada segmen ($k = 3$), penyisipan hanya dapat dilakukan dari *payload* 1 kB – 45 kb dan besar kapasitas data yang dapat ditampung sebesar 396.894 bit.

Pada segmen ($k = 4, k = 5, k = 6, k = 7, k = 8$), metode usulan telah mampu melakukan penyisipan hingga data sisipan ke-12, sebesar 55 kB. Kapasitas maksimum yang dapat ditampung dari segmen $k = 4$ sebesar 528.192 bit, $k = 5$ sebesar 661.490 bit, $k = 6$ sebesar 793.788 bit, dan $k = 7$ sebesar 926.086 bit, serta $k = 8$ sebesar 1.058.284 bit.

Pada hasil pengujian metode perbandingan, LSD, Tabel IV menunjukkan bahwa dari 96 kali pengujian penyisipan, hanya berhasil sebanyak 44 kali dan selebihnya gagal sebanyak 52 kali. Pada parameter segmen $k = 1$ pada metode LSD diperoleh ruang kapasitas maksimum penyisipan data *payload* sebesar 44.099 bit, $k = 2$ kapasitas maksimum 88.198 bit, $k = 3$ kapasitas maksimum 132.297 bit, $k = 4$ d kapasitas maksimum 176.396 bit, $k = 5$ kapasitas maksimum 220.495 bit, $k = 6$ kapasitas maksimum 264.594 bit, $k = 7$ kapasitas maksimum 308.693 bit, dan $k=8$ kapasitas maksimum 352.792 bit.

TABEL V
HASIL PERBANDINGAN PNSR AUDIO 1, 2, DAN 3

Pay-load	Audio 1 (dB)		Audio 2 (dB)		Audio 3 (dB)	
	Usulan	LSD	Usulan	LSD	Usulan	LSD
1	74,45	71,07	85,77	82,25	38,59	57,77
2	67,43	64,05	78,75	75,27	52,17	53,98
3	64,44	58,67	75,75	69,89	49,75	48,76
4	62,67	54,39	73,99	65,61	47,99	45,08
5	59,06	50,67	70,29	61,91	44,09	42,24
6	58,09	46,83	69,32	58,08	43,32	38,82
7	57,26	43,67	68,52	54,84	42,52	35,57
8	54,03	40,65	65,45	51,82	37,92	32,55
9	53,44	38,92	64,86	50,10	38,21	30,83
10	52,92	-	64,33	-	35,60	-
11	50,07	-	61,39	-	35,28	-
12	49,67	-	49,67	-	34,98	-

Besarnya kapasitas penyisipan ini dipengaruhi oleh ruang penyisipan dari sinyal GSS. Semakin banyak *sampling* dari berkas audio, maka semakin banyak pula sinyal GSS yang dihasilkan. Maka, ruang penyisipan juga berbanding lurus dengan banyak data *payload* yang dapat disisipkan. Besarnya segmen juga menjadi faktor penentu dari banyaknya kapasitas data *payload* yang dapat disisipkan. Semakin besar segmen yang digunakan, maka semakin banyak data *payload* yang dapat ditampung.

B. Trend Kualitas Terhadap Segmen (k)

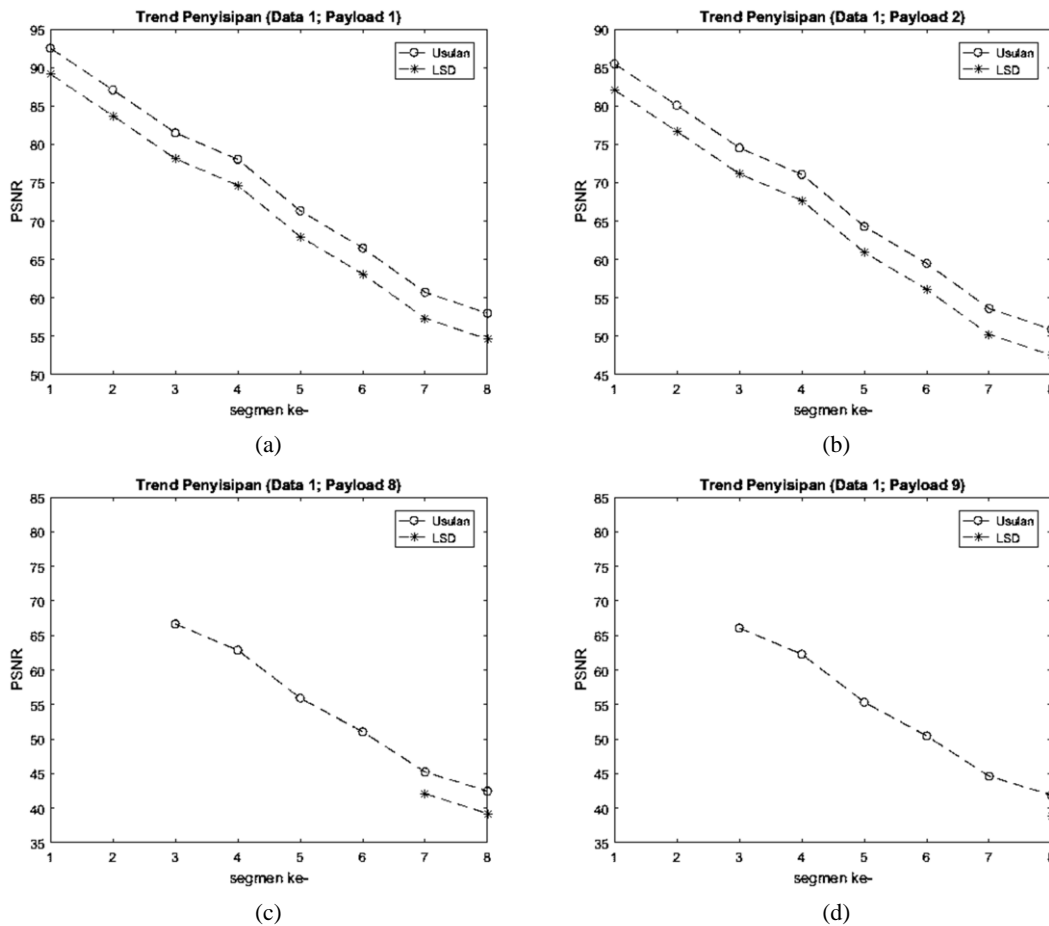
Pengujian ini dilakukan pada audio data 1. Pengujian ini bertujuan untuk melihat *trend* nilai kualitas yang dihasilkan terhadap data *payload* dengan segmen (k).

Gbr. 6 merupakan hasil pola *trend* yang dihasilkan. Terhadap metode usulan dan metode perbandingan diterapkan penyisipan dengan segmen $k = 1$ hingga $k = 8$. Gbr. 6 menunjukkan bahwa kedua metode dapat melakukan penyisipan dan *trend* yang dihasilkan tampak menurun. Hal ini disebabkan oleh selisih antara sinyal sisipan dan sinyal asli semakin besar. Gbr. 6(a) dan Gbr. 6(b) menunjukkan bahwa kedua metode dapat melakukan penyisipan dari segmen $k = 1$ hingga $k = 8$, hanya saja terjadi penurunan kualitas audio. Gbr. 6(a) memiliki hasil kualitas *stego-audio* lebih baik dibandingkan dengan Gbr 6(b). Hal ini karena jumlah bit sisipan data rahasia lebih sedikit.

Pada Gbr. 6(c), penyisipan dilakukan dengan segmen $k = 1$ hingga $k = 8$ dengan besar *payload* 35 kb, hanya saja kedua metode tidak dapat melakukan penyisipan sepenuhnya. Metode usulan paling kecil hanya mampu menyisipkan dengan menggunakan segemen $k = 3$ sedangkan metode perbandingan $k = 7$.

Gbr. 6(d) menggunakan *payload* sebesar 40 kb. Hasil metode usulan yang diperoleh sama dengan Gbr 6(c), berbeda dengan metode perbandingan yang hanya dapat menyisipkan menggunakan segmen $k = 8$. Hal ini disebabkan oleh keterbatasan ruang penyisipan data pada metode LSD.

Dari pengujian ini dapat disimpulkan bahwa semakin besar segmen yang digunakan, maka kualitas dari *stego-audio* semakin menurun, dan berlaku sebaliknya, semakin kecil nilai parameter segmen yang digunakan, maka kualitas dari *stego-audio* semakin baik.



Gbr. 6 Trend kualitas dari penyisipan.

TABEL VI
HASIL PERBANDINGAN PNSR AUDIO 4, 5, DAN 6

Payload	Audio 4 (dB)		Audio 5 (dB)		Audio 6 (dB)	
	Usulan	LSD	Usulan	LSD	Usulan	LSD
1	73,94	86,39	44,34	56,77	34,15	46,33
2	87,52	82,60	57,92	52,97	47,72	42,54
3	85,11	77,38	55,51	47,75	45,31	37,32
4	83,34	73,70	53,74	44,07	43,55	33,64
5	79,45	70,86	49,84	41,23	39,65	30,80
6	78,67	67,43	49,07	37,81	38,88	27,37
7	77,88	64,19	48,28	34,57	38,08	24,13
8	73,27	61,17	43,67	31,55	33,48	21,11
9	73,57	59,45	43,97	29,82	33,77	19,39
10	70,96	-	41,36	-	31,16	-
11	70,64	-	41,04	-	30,84	-
12	70,34	-	40,73	-	30,54	-

TABEL VII
HASIL PERBANDINGAN PNSR AUDIO 7, 8, DAN 9

Payload	Audio 7 (dB)		Audio 8 (dB)		Audio 9 (dB)	
	Usulan	LSD	Usulan	LSD	Usulan	LSD
1	33,94	46,36	75,15	87,60	74,02	86,45
2	47,52	42,56	88,73	83,80	87,60	82,65
3	45,10	37,34	86,31	78,58	85,18	77,44
4	43,34	33,66	84,55	74,91	83,42	73,76
5	39,44	30,82	80,65	72,07	79,52	70,92
6	38,67	27,40	79,88	68,64	78,75	67,49
7	37,87	24,16	79,08	65,40	77,95	64,24
8	33,27	21,14	74,48	62,38	73,35	61,23
9	33,56	19,41	74,77	60,66	73,64	59,51
10	30,95	-	72,16	-	71,03	-
11	30,63	-	71,84	-	70,71	-
12	30,33	-	71,54	-	70,41	-

C. Perbandingan Rerata PSNR yang Dihasilkan

Hasil pengujian pada Tabel V hingga Tabel IX merupakan perbandingan kualitas dari dua metode. Hasil pengujian diperoleh dari nilai rerata segmen $k = 1$ hingga $k = 8$. Tujuan dari pengujian adalah untuk mengetahui instrumen dan genre apa saja yang tepat digunakan untuk penyisipan. Besar nilai kualitas PSNR yang ideal lebih besar atau sama dengan 30 dB.

Pada penggunaan kasus *payload* (1 kB), hasil pengujian menunjukkan kualitas PSNR terbaik diperoleh pada audio 8 (instrumen piano dan genre klasik) dengan PSNR sebesar 87,6 dB. Pada penggunaan data rahasia 2 (5 kB) hingga data rahasia 12 (55kB), kualitas PSNR terbaik diperoleh pada audio 8 (instrumen piano dan genre klasik) dengan PSNR sebesar 88,7 dB. Kualitas PSNR paling buruk diperoleh pada audio 10 (instrumen *saxophone* genre *country-folk*), dengan PSNR sebesar 14,98 dB dengan metode LSD.

TABEL VIII
HASIL PERBANDINGAN PNSR AUDIO 10, 11, DAN 12

Pay-load	Audio 10 (dB)		Audio 11 (dB)		Audio 12 (dB)	
	Usulan	LSD	Usulan	LSD	Usulan	LSD
1	31,00	41,92	52,03	64,49	64,69	77,15
2	44,58	38,12	65,61	60,69	78,27	73,35
3	42,17	32,90	63,20	55,47	75,86	68,13
4	40,40	29,23	61,43	51,79	74,09	64,45
5	36,50	26,39	57,53	48,95	70,19	61,61
6	35,73	22,96	56,76	45,53	69,42	58,19
7	34,94	19,72	55,97	42,29	68,62	54,94
8	30,33	16,70	51,36	39,27	64,02	51,93
9	30,63	14,98	51,66	37,54	64,31	50,20
10	28,02	-	49,05	-	61,71	-
11	27,70	-	48,73	-	61,38	-
12	27,39	-	48,42	-	61,08	-

TABEL IX
HASIL PERBANDINGAN PNSR AUDIO 13, 14, DAN 15

Pay-load	Audio 13 (dB)		Audio 14 (dB)		Audio 15 (dB)	
	Usulan	LSD	Usulan	LSD	Usulan	LSD
1	67,32	79,78	41,42	52,54	72,67	85,16
2	80,90	75,98	55,00	48,74	86,24	81,37
3	78,49	70,76	52,59	43,52	83,83	76,15
4	76,72	67,08	50,82	39,84	82,07	72,47
5	72,82	64,24	46,92	37,00	78,17	69,63
6	72,05	60,82	46,15	33,58	77,40	66,21
7	71,26	57,58	45,36	30,34	76,60	62,96
8	66,65	54,56	40,75	27,32	72,00	59,94
9	66,95	52,83	41,05	25,59	72,29	58,22
10	64,34	-	38,44	-	69,68	-
11	64,02	-	38,12	-	69,36	-
12	63,72	-	37,82	-	69,06	-

Pada audio 1, seperti pada Tabel V, dan audio 2, seperti pada Tabel VI, pada metode usulan terjadi peningkatan kemampuan penyisipan, dibandingkan dengan metode LSD. Audio 3 hingga audio 15, ditunjukkan pada Tabel V hingga Tabel IX, dengan menggunakan *payload* 1 kb, metode LSD lebih baik dibandingkan dengan metode usulan. Hal ini dikarenakan *slot* area yang telah dialokasikan untuk penyisipan tidak terisi seluruhnya, sehingga nilai selisih magnitude *stego-audio* metode LSD lebih mirip dengan nilai media pembawa.

Secara keseluruhan, pada audio 1 hingga audio 15, metode usulan mampu menyisipkan dan mengekstraksi kembali pesan rahasia. Sedangkan pada metode LSD terjadi kegagalan dan ketidakmampuan penyisipan pada *payload* 45 kB, 50 kB, dan 55 kb. Penyebabnya ialah alokasi ruang penyisipan yang terbatas. Hasil ini menunjukkan bahwa metode usulan dapat meningkatkan kapasitas penyisipan dibandingkan metode sebelumnya.

V. KESIMPULAN DAN SARAN

Dari hasil percobaan dapat disimpulkan bahwa dari sisi kapasitas dengan berbagai segmen, hasil metode usulan lebih baik dengan metode LSD. Hal ini ditunjukkan dari 12 variasi

data *payload* dan delapan variasi segmen, diperoleh metode usulan 15 kali gagal melakukan penyisipan dan 81 berhasil melakukan penyisipan. Artinya, metode usulan memiliki tingkat keberhasilan sebesar 84,4%, sedangkan metode LSD memiliki 52 kali kegagalan penyisipan dan hanya 44 kali keberhasilan penyisipan. Dengan kata lain, tingkat keberhasilan metode LSD hanya sebesar 45,8%. Dari evaluasi kualitas menggunakan PSNR dapat disimpulkan bahwa semakin besar kapasitas data yang disisipkan, maka kualitas dari *stego-audio* semakin menurun. Secara keseluruhan, metode usulan memiliki kualitas jauh lebih unggul dibandingkan dengan kualitas dengan metode LSD, meskipun pada beberapa kondisi metode LSD lebih unggul, seperti pada audio 3--audio 15. Namun, hal itu terjadi hanya pada kapasitas yang kecil, sebesar 1 kB, sedangkan untuk kapasitas yang lebih besar, metode usulan lebih baik dibandingkan LSD. Kualitas penyisipan terbaik diperoleh pada audio 8, dengan instrumen yang digunakan piano dan genre klasik, sehingga audio ini dapat merekomendasikan sebagai media pembawa untuk penyisipan data.

Untuk penelitian selanjutnya, disarankan penyisipan segmen dapat dikembangkan sehingga kualitas dari *stego-audio* dapat lebih ditingkatkan.

REFERENSI

- [1] "Internet live stats," [Online]. Available: <http://www.internetlivestats.com/internet-users/>. [Accessed 5 April 2017].
- [2] C. P. Pflieger dan S. L. Pflieger, *Security in Computing*, fourth edition, New Jersey: Prentice Hall PTR Upper Saddle River, 2006.
- [3] E. K. Marnani, Z. Karami, dan M. E. Jazi, "A Comparison of Some Audio Watermarking Methods," *International Conference on Electrical Engineering, Computing Science and Automatic Control*, Toluca, Mexico, 2009.
- [4] Bhateja, Ashok K. Chaudhury, dan S. Saxena, "Robust Online Signature Based Cryptosystem," *14th International Conference on Frontiers in Handwriting Recognition*, 2014.
- [5] Liang, B. Wu, dan Z. Linyou, "A Novel Fingerprint-Based Biometric Encryption," *Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, 2014.
- [6] Thangadurai, K. Devi, dan G. Sudha, "An analysis of LSB based image steganography techniques," *International Conference on Computer Communication and Informatics*, 2014.
- [7] H. J. Kim, Y. H. Choi, J. W. Seok, dan J. W. Hong, "Audio watermarking techniques," *Intelligent Watermarking Techniques*, 2004.
- [8] H. J. Kim, dan Y. H. Choi, "A novel echo-hiding scheme with backward and forward kernels," *IEEE Transactions on Circuits and Systems for Video and Technology*, vol. 13, no. 8, pp. 885-889, 2003.
- [9] W. Bender, D. Gruhl, N. Marimoto, dan A. Lu, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, pp. 313-336, 1996.
- [10] M. Swanson, B. Zhu, A. Tewfik, dan L. Boney, "Robust audio watermarking using perceptual masking," *Signal Processing*, vol. 66, pp. 337-355, 1998.
- [11] G. W. Wornell, dan B. Chen, "Quantization index modulation: a class of provably good methods for digital watermarking," *IEEE Transaction on Information Theory*, vol. 47, pp. 1423-1443, 2001.
- [12] S. K. Lee dan Y. S. Ho, "Digital audio watermarking in the cepstrum domain," *IEEE Transactions on Consumer Electronics*, vol. 46, pp. 744-750, 2000.

- [13] I. K. Yeo dan H. J. Kim, "Modified patchwork algorithm: A novel audio watermarking scheme," *IEEE Transactions on Speech and audio Processing*, vol. 11, pp. 381-386, 2003.
- [14] M. L. M. Kiah, A. A. Zaidan, B. B. Zaidan, A. M. Ahmed, dan S. M. Al-bakri, "A review of audio based steganography and digital watermarking," *International Journal of the Physical Sciences*, vol. 6, pp. 3837-3850, 2011.
- [15] D. Gruhl, dan W. Bender, "Echo hiding," *Proceeding of Information Hiding Workshop*, 1996.
- [16] L. Gang, A. N. Akansu, dan M. Ramkumar, "MP3 resistant oblivious steganography," *Proceedings of IEEE International Conference onAcoustics, Speech, and Signal Processing*, Salt Lake City, 2001.
- [17] F. Djebbar, B. Ayady, dan H. H. K. A. Meraimx, "A view on latest audio steganography techniques," *International Conference on Innovations in Information Technology*, 2011.
- [18] B. Paillard, P. Mabillean, S. Morissette, dan J. Soumagne, "PERCEVAL : Perceptual Evaluation of the Quality of AudioSignals," *journal of Audio Engeneering Society*, vol. 40, pp. 21-31, 1992.
- [19] J. Liu, K. Zhou, dan H. Tian, "Least-significant-digit Steganography in Low Bitrate Speech," *IEEE ICC 2012 - Communication and Information Systems Security Symposium*, 2012.
- [20] R. Ansari, H. Malik, dan A. Khokhar, "Data-hiding in audio using frequency-selective phase alteration," *IEEE International Conference onAcoustics, Speech, and Signal Processing*, Montreal, Quebec, Canada, 2004.
- [21] D. Yadav, M. Agrawal, dan A. Arora, "Performance Evaluation of LSB and LSD in Steganography," *IEEE 2014 5th International Conference-Confluence The Next Generation Information Technology Summit*, Nioda, India, 2014.
- [22] J. J. Bosch, J. Janer, F. Fuhrman, dan P. Herrera , "A Comparison of Sound Segregation Techniques for Predominant Instrument Recognition in Musical Audio Signals," *Int Proc. ISMIR.*, 2012.