

## BAB IV HASIL DAN ANALISA

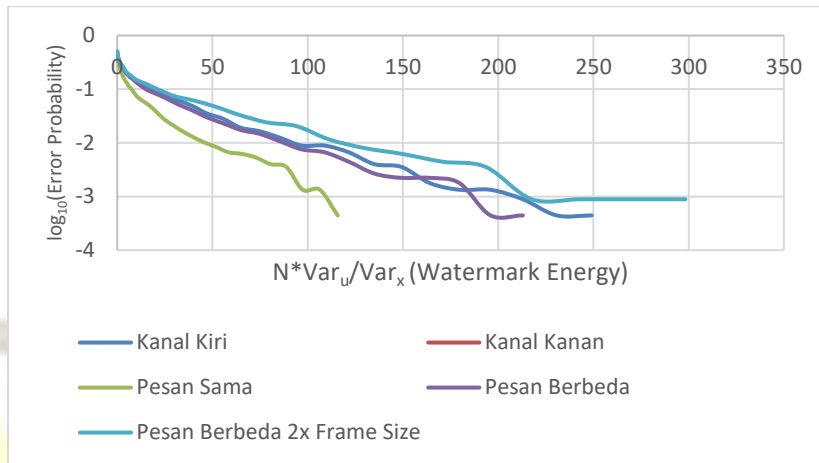
### 4.1. Analisis Kinerja Sistem

Penelitian ini melakukan simulasi penerapan metode steganografi *Spread Spectrum* pada audio *Parametric Stereo* untuk menganalisis kemampuan penyembunyian pesan rahasia. Evaluasi kinerja difokuskan pada tiga aspek yaitu efisiensi dan kemampuan pemulihan pesan, yang dianalisis melalui hubungan antara *watermark energy* dan *Error probability*, tingkat perubahan sinyal (*fidelity*), yang diukur dengan *Signal-to-Noise Ratio* (SNR), dan *level* terdeteksi secara persepsi (*imperceptibility*), yang dievaluasi menggunakan *Objective Difference Grade* (ODG). Analisis dilakukan dengan membandingkan lima skenario penyisipan pada Kanal Tunggal Kiri, Kanal Tunggal Kanan, Kedua Kanal dengan Pesan yang Sama, Kedua Kanal dengan Pesan yang Berbeda, dan Kedua Kanal dengan Pesan yang Berbeda  $2x$  *frame size* pada berbagai variasi ukuran frame (1024 dan 5120 sampel) dan dua tingkat *bitrate* (64 kbps dan 128 kbps). Tujuannya adalah untuk mengidentifikasi konfigurasi optimal yang menjamin pesan dapat dipulihkan sepenuhnya sambil meminimalkan perubahan yang dapat dideteksi dalam lingkungan kompresi *Parametric Stereo* yang kompleks.

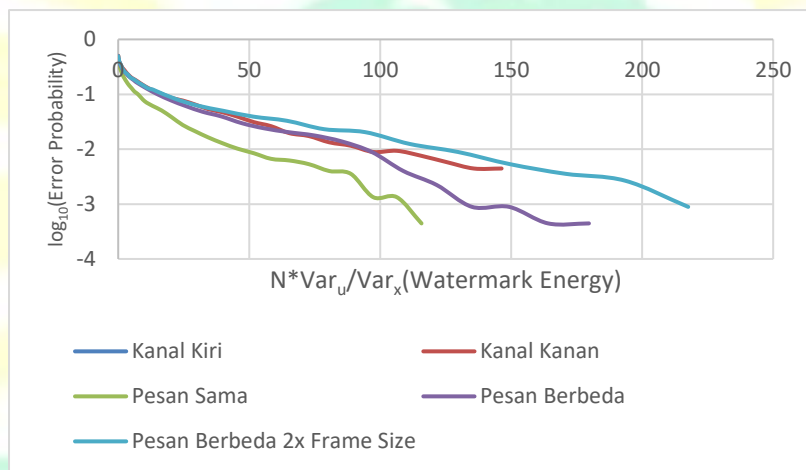
#### 4.1.1. Analisis Grafik Antara *Watermark energy* Terhadap *Error probability* dari Pesan Rahasia Yang Diterima

Pada pengujian grafik *watermark energy* terhadap *error probability* ini memberikan gambaran bagaimana efisiensi dan kemampuan pemulihan pesan dengan menganalisis hubungan antara *watermark energy* dengan *error probability* pada sistem steganografi audio, pada kanal kiri dan kanal kanan, dengan 2 variasi bit rate yaitu 64 kbps dan 128 kbps, dan lima variasi *frame size* yaitu 1024 dan 5120. Setiap grafik menyajikan performa dari lima variasi penyisipan pesan, yakni: pesan pada kanal kiri saja, pesan pada kanal kanan saja, pesan yang berbeda, pesan yang sama dan pesan berbeda  $2x$  *frame size*. Seluruh nilai *Error Probability* dalam penelitian ini disajikan dalam skala logaritma basis 10 ( $\log_{10}$ ).

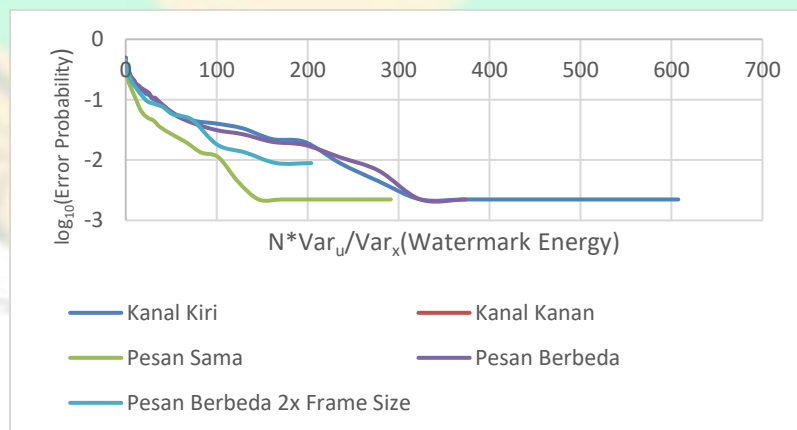
### 4.1.1.1. Bitrate 64 kbps



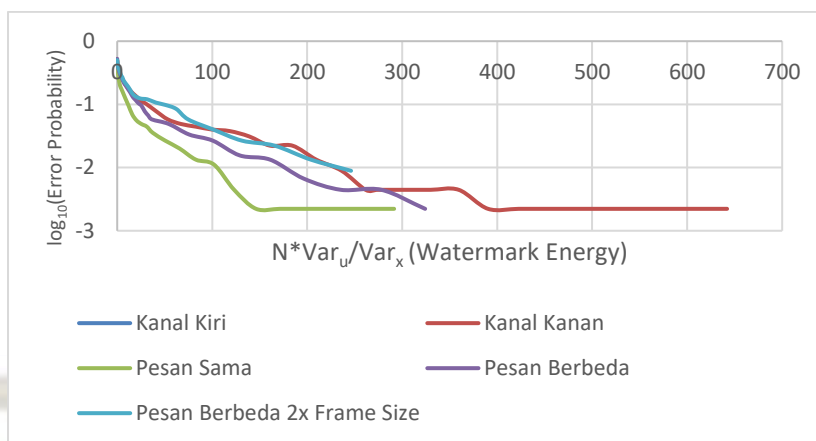
(a)



(b)



(c)



(d)

**Gambar 4. 1** Grafik *Watermark energy* terhadap  $\log_{10}$  *Error probability* dari pesan rahasia yang diterima pada variasi penyisipan di (Kanal kiri, Kanal kanan, Pesan sama, Pesan berbeda, dan Pesan berbeda 2x) pada *Frame size* 1024 dan 5120 pada *Bitrate* 64 kbps ((a dan c) Kanal kiri dan (b dan d) Kanal kanan)

Hasil analisis grafik pada gambar 4.1 menunjukkan kebutuhan *watermark energy* (WE) untuk mendapatkan  $\log_{10}$  *Error probability* (EP) terendah, Pada *frame size* 1024 *bitrate* 64 kbps yang terdapat pada gambar 4.1 (a dan b). Grafik pada menunjukkan variasi penyisipan pesan sama membutuhkan *watermark energy* (WE) paling rendah untuk mencapai  $\log_{10}$  *error probability* (EP) terendah. Variasi pesan sama berarti *watermark* pada kanal kiri dan kanan identik (ada redundansi antar-kanal), sehingga efisiensi dilihat dari WE terkecil yang mampu mencapai  $\log_{10}$  EP terendah. Hasilnya, pada pesan sama diperoleh WE 115,76 dengan  $\log_{10}$  EP -3,352, jauh lebih rendah dibanding kanal kiri saja (WE 249,07,  $\log_{10}$  EP -3,352), kanal kanan saja (WE 146,32,  $\log_{10}$  EP -2,352), pesan berbeda pada kanal kanan (WE 179,67,  $\log_{10}$  EP -3,352) dan kanal kiri (WE 212,96,  $\log_{10}$  EP -3,352), serta pesan berbeda 2x *frame size* (WE 298,29,  $\log_{10}$  EP -3,051). Jadi, berdasarkan data ini, pesan sama merupakan variasi paling efisien karena mencapai  $\log_{10}$  EP yang sama atau lebih baik dengan WE paling kecil.

Pada variasi kanal tunggal (*watermark* hanya di satu kanal), kebutuhan WE cenderung lebih besar atau  $\log_{10}$  EP menjadi lebih buruk karena tidak ada redundansi. Pada kanal kiri saja, untuk mencapai  $\log_{10}$  EP yang sangat rendah diperoleh WE 249,07 dengan  $\log_{10}$  EP -3,352, sedangkan pada kanal kanan saja diperoleh WE 146,32 namun  $\log_{10}$  EP hanya -2,352 (lebih buruk). Ini menunjukkan bahwa penyisipan pada satu kanal saja tidak seefisien pesan sama, karena performa sangat bergantung pada kanal yang dipakai dan tidak ada “bantuan” dari kanal lainnya.

Pada variasi pesan berbeda, kedua kanal membawa pesan yang tidak sama sehingga tidak ada redundansi, namun keduanya tetap aktif menyisipkan *watermark*. Hasilnya, untuk mencapai  $\log_{10}$  EP terendah diperoleh pada kanal kanan

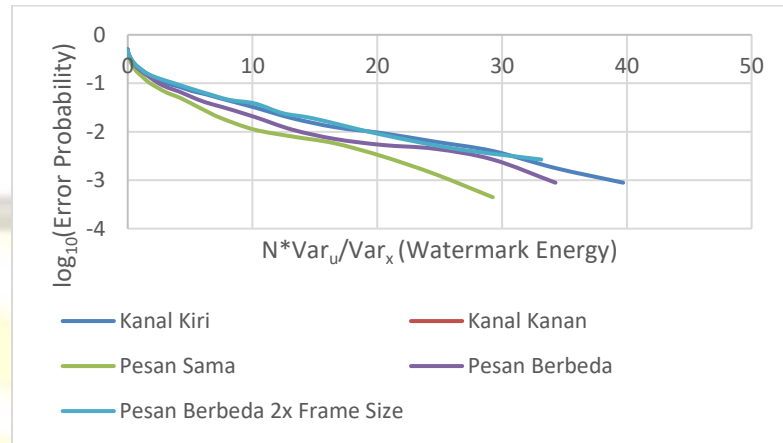
WE 179,67 dengan  $\log_{10} EP -3,352$  dan pada kanal kiri WE 212,96 dengan  $\log_{10} EP -3,352$ . Nilai WE ini lebih besar dibanding pesan sama, tetapi masih lebih baik daripada kanal tunggal kiri saja yang membutuhkan WE lebih tinggi untuk  $\log_{10} EP$  yang sama rendahnya.

Pada variasi pesan berbeda  $2x$  *frame size*, *frame size* diperbesar menjadi 2048 sampel ( $1024 \text{ sampel} \times 2$ ), kebutuhan energi menjadi paling besar dan  $\log_{10} EP$  juga lebih buruk dibanding variasi yang mencapai  $\log_{10} EP -3,352$ . Hasilnya diperoleh WE 298,29 dengan  $\log_{10} EP -3,051$  pada kanal kiri dan kanan. Artinya, pada kondisi ini peningkatan beban penyisipan (dua pesan berbeda) membuat sistem memerlukan WE lebih tinggi dan tetap menghasilkan EP yang tidak serendah variasi pesan sama maupun pesan berbeda biasa.

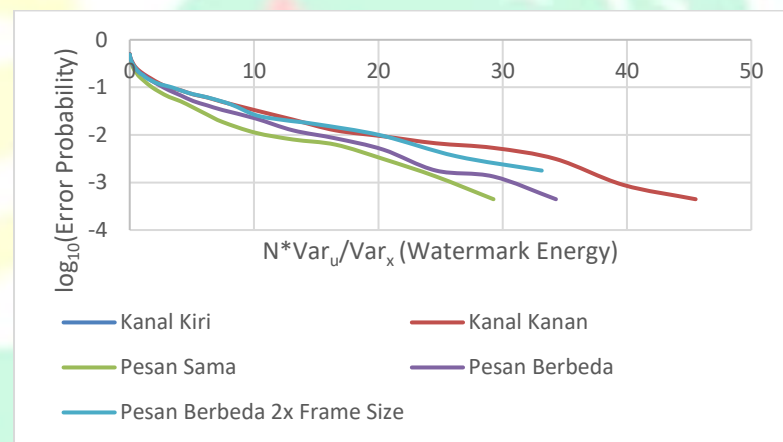
Pada *frame size* 5120 bitrate 64 kbps yang terdapat pada gambar 4.1 (c dan d). Pada grafik tersebut menunjukkan WE yang dibutuhkan meningkat dan  $\log_{10} EP$  cenderung memburuk dibanding *frame size* 1024 pada hampir semua variasi. Hal ini terjadi karena pada *frame size* yang lebih besar, *watermark* dibaca pada rentang sampel yang jauh lebih panjang dalam satu *frame*. Saat melewati proses kompresi *Parametric Stereo* pada bitrate rendah (kuantisasi dan parameterisasi lebih agresif), perubahan-perubahan kecil pada sinyal di sepanjang *frame* yang panjang ini dapat mengganggu kestabilan pola *watermark*, sehingga proses pembacaan *watermark* menjadi kurang tegas dan deteksi menjadi lebih sulit ( $\log_{10} EP$  naik). Untuk mengimbangi hilangnya *watermark* akibat kompresi, WE total perlu dinaikkan sejak awal, namun pada bitrate 64 kbps distorsi kompresi masih cukup besar sehingga meskipun WE sudah dinaikkan,  $\log_{10} EP$  yang dicapai tetap umumnya tidak serendah pada *frame size* 1024. Ini terlihat pada pesan sama yang naik dari WE 115,76 ( $\log_{10} EP -3,352$ ) menjadi 291,50 ( $\log_{10} EP -2,65$ ), kanal kiri saja dari 249,07 ( $\log_{10} EP -3,352$ ) menjadi 607,4 ( $\log_{10} EP -2,653$ ), kanal kanan saja dari 146,32 ( $\log_{10} EP -2,352$ ) menjadi 641,99 ( $\log_{10} EP -2,65$ ), dan pesan berbeda dari 179,67–212,96 ( $\log_{10} EP -3,352$ ) menjadi 324,1–373,73 ( $\log_{10} EP -2,65$ ). Pada pesan berbeda  $2x$  *frame size*, WE memang turun menjadi 203,91–246,14, tetapi  $\log_{10} EP$  tetap memburuk menjadi -2,05, yang menunjukkan bahwa pada *frame* yang besar, efisiensi energi tidak otomatis menghasilkan ketahanan yang lebih baik terhadap kompresi.

#### 4.1.1.2. Bitrate 128 kbps

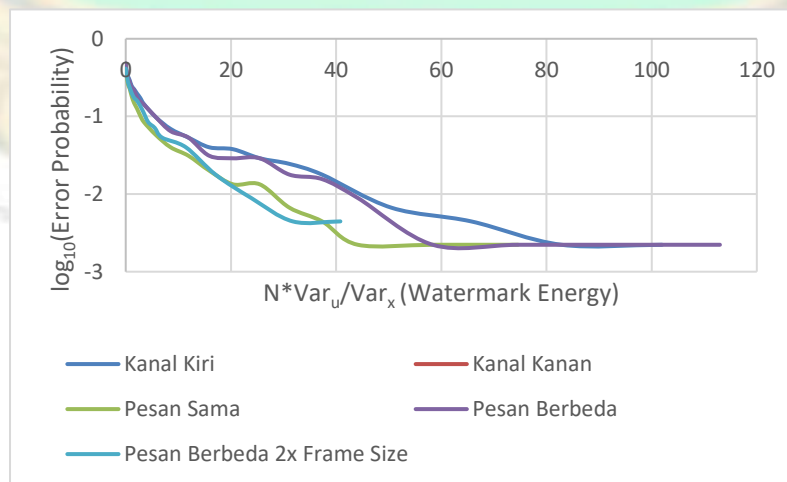
Pada gambar 4.2 di bawah ini menyajikan perbandingan kinerja kelima skenario penyisipan data rahasia pada *frame size* 1024 dan 5120 pada *bitrate* 128 kbps.



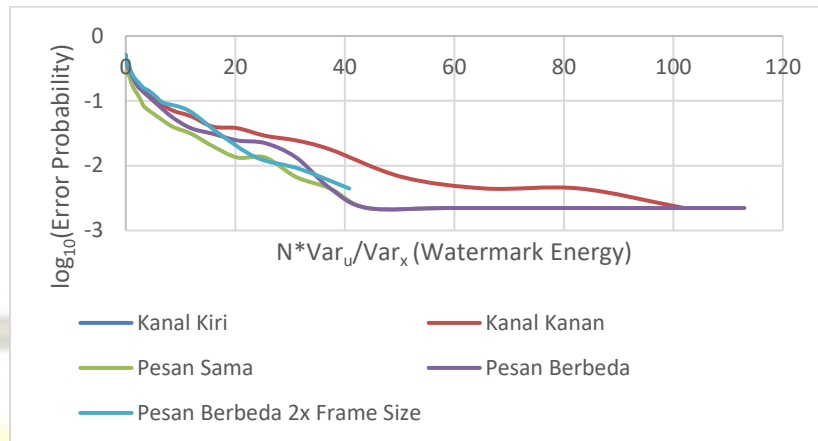
(a)



(b)



(c)



(d)

**Gambar 4. 2** Grafik *Watermark Energy* terhadap  $\log_{10}$  *Error Probability* dari Pesan Rahasia yang Diterima Pada Variasi Penyisipan di (Kanal kiri, Kanal kanan, Pesan sama, Pesan berbeda, dan Pesan berbeda 2x) Pada *Frame size* 1024 dan 5120 Pada *Bitrate* 128 kbps ((a dan c) Kanal Kiri dan (b dan d) Kanal Kanan)

Hasil analisis grafik pada gambar 4.2 menunjukkan kebutuhan *Watermark Energy* (WE) untuk mendapatkan  $\log_{10}$  *error probability* (EP) terendah, Pada *frame size* 1024 sampel *bitrate* 128 kbps yang terdapat pada gambar 4.2. (a dan b) Grafik variasi penyisipan pesan sama kembali menunjukkan kebutuhan *watermark energy* (WE) paling rendah untuk mencapai  $\log_{10}$  *error probability* (EP) terendah. *Watermark* pada kanal kiri dan kanan identik sehingga terdapat redundansi antar-kanal. Hasilnya, pada pesan sama diperoleh WE 29,260 dengan  $\log_{10}$  EP -3,352, jauh lebih rendah dibanding kanal kiri saja (WE 39,70,  $\log_{10}$  EP -3,051), kanal kanan saja (WE 45,51,  $\log_{10}$  EP -3,352), pesan berbeda (WE 34,28,  $\log_{10}$  EP -3,051), serta pesan berbeda 2x *frame size* (WE 33,14,  $\log_{10}$  EP -2,75). Dibandingkan *bitrate* 64 kbps yang membutuhkan WE 115,76 untuk  $\log_{10}$  EP yang sama, kenaikan *bitrate* ke 128 kbps menurunkan kebutuhan WE secara drastis. Hal ini konsisten dengan fakta bahwa pada *bitrate* lebih tinggi, kompresi tidak perlu membuang atau mengkuantisasi data audio sebanyak pada *bitrate* rendah, sehingga *watermark* yang disisipkan cenderung mengalami lebih sedikit distorsi dan dapat bertahan dengan energi yang lebih rendah.

Pada variasi kanal tunggal, kebutuhan WE meningkat dibanding pesan sama karena tidak ada redundansi antar-kanal. Pada kanal kiri saja diperoleh WE 39,70 dengan  $\log_{10}$  EP -3,051, sedangkan pada kanal kanan saja diperoleh WE 45,51 dengan  $\log_{10}$  EP -3,352. Menariknya, pada *bitrate* 128 kbps kanal kanan saja justru membutuhkan WE lebih besar namun mencapai  $\log_{10}$  EP lebih rendah dibanding kanal kiri saja. Perbedaan ini menunjukkan bahwa performa antar-kanal tidak selalu konsisten dan dapat dipengaruhi oleh bagaimana proses *encoding* memperlakukan masing-masing kanal.

Pada variasi pesan berbeda, kedua kanal membawa pesan yang tidak identik sehingga tidak ada redundansi. Diperoleh WE 34,28 dengan  $\log_{10}$  EP -3,051. Nilai WE ini lebih besar dibanding pesan sama (29,260), tetapi lebih rendah dibanding kedua variasi kanal tunggal. Hal ini dapat dijelaskan oleh kondisi di mana kedua kanal tetap aktif membawa *watermark* sehingga beban penyisipan tidak tertumpuk pada satu kanal saja, meskipun masing-masing pesan tetap harus bertahan sendiri dari proses kompresi.

Pada variasi pesan berbeda  $2x$  *frame size*, *frame size* diperbesar menjadi 2048 ( $1024 \times 2$ ) dengan dua pesan berbeda pada masing-masing kanal. Diperoleh WE 33,14 dengan  $\log_{10}$  EP -2,75. Meskipun WE-nya relatif rendah (lebih rendah dari kanal tunggal),  $\log_{10}$  EP-nya (-2,75) justru paling buruk di antara seluruh variasi pada *frame size* ini. Hal ini konsisten dengan tren bahwa *frame size* yang lebih besar mendistribusikan energi watermark ke lebih banyak sampel, sehingga meskipun WE total tidak perlu terlalu besar, *watermark* per sampel menjadi lebih tipis dan  $\log_{10}$  EP yang dicapai tidak sebaik *frame size* yang lebih kecil.

Pada *frame size* 5120 bitrate 128 kbps yang terdapat pada gambar 4.2. (c dan d) Grafik seluruh variasi menunjukkan kebutuhan WE yang lebih tinggi dan  $\log_{10}$  EP yang cenderung lebih buruk dibanding *frame size* 1024. Pada pesan sama WE naik dari 29,260 ( $\log_{10}$  EP -3,352) menjadi 74,5 ( $\log_{10}$  EP -2,65), pada kanal kiri saja naik dari 39,70 ( $\log_{10}$  EP -3,051) menjadi 101,96 ( $\log_{10}$  EP -2,653), pada kanal kanan saja naik dari 45,51 ( $\log_{10}$  EP -3,352) menjadi 101,96 ( $\log_{10}$  EP -2,65), dan pada pesan berbeda naik dari 34,28 ( $\log_{10}$  EP -3,051) menjadi 112,98 ( $\log_{10}$  EP -2,65). Pada pesan berbeda  $2x$  *frame size* WE justru naik menjadi 40,81 dengan  $\log_{10}$  EP -2,35, yang merupakan  $\log_{10}$  EP paling buruk pada *frame size* ini. Tren ini menunjukkan bahwa pada *frame size* besar, watermark dibaca pada rentang sampel yang jauh lebih panjang dalam satu *frame*. Untuk mengompensasi hal tersebut, WE total perlu dinaikkan, namun karena kompresi tetap mengikis sebagian *watermark*,  $\log_{10}$  EP yang dicapai umumnya tidak serendah pada *frame size* 1024 sampel. Meskipun demikian, dibandingkan bitrate 64 kbps pada *frame size* yang sama, kebutuhan WE pada 128 kbps tetap jauh lebih rendah pada semua variasi, yang menunjukkan bahwa bitrate lebih tinggi secara konsisten membuat sistem lebih efisien karena distorsi kompresi yang lebih kecil.

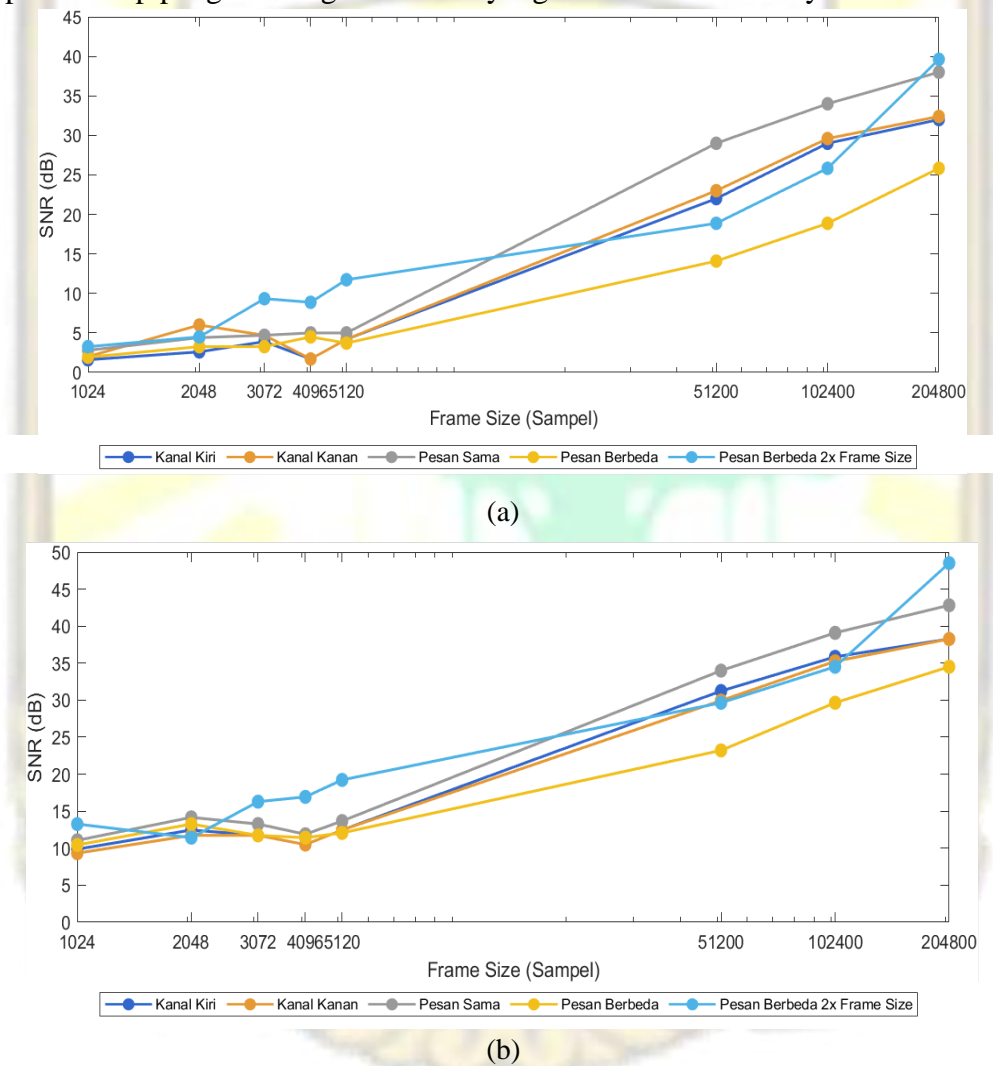
#### **4.1.2. Signal to Noise Ratio**

##### **4.1.2.1. Pengaruh Frame size Pada Pengukuran Signal to Noise Ratio Sebagai Indikator Fidelity**

Pengukuran *Signal-to-Noise Ratio* (SNR) dengan variasi *Frame size* (1024, 2048, 3072, 4096, 5120, 51200, 102400, dan 204800 sampel) dengan *bitrate* (64 kbps serta 128 kbps) dilakukan untuk mengetahui kinerja dari variasi penyisipan dan mengevaluasi tingkat perubahan sinyal (*fidelity*) setelah proses penyisipan pesan rahasia (*Embedding*). Dalam konteks steganografi, nilai SNR yang tinggi

mengindikasikan bahwa distorsi (*noise*) yang ditimbulkan oleh proses *embedding* sangat kecil dibandingkan dengan sinyal asli. Hal ini berarti jejak penyisipan pesan menjadi minimal, sehingga semakin sulit dideteksi melalui analisis sinyal secara objektif. Dengan demikian, pengukuran SNR berfungsi sebagai indikator kuantitatif awal untuk menilai seberapa aman suatu konfigurasi steganografi dari segi perubahan sinyal yang dihasilkan.

Pada grafik pada gambar 4.3 memberikan gambaran bagaimana pengaruh *Frame size* terhadap *Signal Noise to Ratio* pada Empat Skenario Penyisipan (Kanal kiri, Kanal kanan, Pesan sama, Pesan berbeda ) dengan *bitrate* 64 kbps dan 128 kbps terhadap pengaruh tingkat distorsi yang ditambahkan ke sinyal audio.



**Gambar 4.3** Pengaruh Ukuran *Frame size* terhadap *Signal Noise to Ratio* pada Lima Skenario Penyisipan (Kanal kiri, Kanal kanan, Pesan sama, Pesan berbeda , Pesan berbeda 2x *frame size*) dengan *bitrate* (a) 64 kbps dan (b) 128 kbps

Hasil pada Gambar 4.3 (a) menunjukkan bahwa semakin besar *frame size* maka nilai SNR cenderung meningkat pada semua skenario penyisipan. SNR di sini

dipakai sebagai indikator tingkat perubahan sinyal (*fidelity*) setelah proses *embedding*, di mana SNR tinggi berarti selisih antara audio asli dan audio stego semakin kecil sehingga jejak penyisipan menjadi minimal. Bukti pada grafik memperlihatkan bahwa pada *frame size* kecil (1.024–5.120 sampel) SNR masih rendah 2 dB sampai dengan 13 dB, sedangkan ketika *frame size* diperbesar ke 51.200, 102400, dan 204.800 sampel SNR naik tajam dengan selisih 20 dB hingga masuk dengan nilai tertinggi yaitu 39,62 dB pada *frame size* 204800 sampel. Kenaikan ini terjadi karena pada *frame size* besar, pesan dapat disebarkan ke lebih banyak sampel, sehingga perubahan yang terjadi pada tiap sampel menjadi kecil dan *noise* steganografi ikut mengecil. Kesimpulannya, pada 64 kbps, *frame size* yang lebih besar menghasilkan perubahan sinyal yang lebih kecil dan nilai SNR menjadi lebih tinggi.

Pada bitrate 128 kbps, dengan grafik pada gambar 4.3 (b) juga menunjukkan tren yang sama, yaitu *frame size* yang lebih besar 51200, 102400, dan 204800 sampel menghasilkan SNR yang lebih tinggi. SNR tetap menjadi indikator untuk melihat seberapa besar perubahan sinyal akibat *embedding*, sehingga SNR yang lebih tinggi berarti perubahan sinyal terhadap sinyal referensi semakin kecil. Bukti pada grafik menunjukkan bahwa pada *frame size* yang sama, SNR pada 128 kbps umumnya berada lebih tinggi dibanding 64 kbps dan pada *frame size* terbesar dapat mencapai nilai tertinggi yaitu 48,62 dB pada variasi pesan berbeda  $2x$  *frame size*. Hal ini terjadi karena pada konfigurasi 128 kbps, sinyal audio yang digunakan memiliki kualitas representasi yang lebih baik, sehingga ketika *embedding* dilakukan dengan parameter yang sama, selisih yang terbentuk antara audio stego dan audio asli menjadi lebih kecil secara relatif. Kesimpulannya, selain *frame size*, penggunaan bitrate 128 kbps menghasilkan SNR lebih tinggi dibanding 64 kbps karena perubahan akibat *embedding* menjadi lebih kecil relatif terhadap sinyal asli.

Perbedaan antar skenario menunjukkan bahwa pola penyisipan ikut memengaruhi besar kecilnya perubahan sinyal. Indikatornya tetap sama, yaitu skenario dengan SNR lebih tinggi berarti perubahan sinyal lebih kecil. Bukti pada grafik menunjukkan skenario “Pesan sama” cenderung menghasilkan SNR lebih tinggi dibanding “Pesan berbeda”, sedangkan penyisipan satu kanal (kanal kiri atau kanal kanan) berada di tengah, dan pada *frame size* sangat besar “Pesan berbeda  $2x$  *Frame Size*” dapat mencapai nilai yang sangat tinggi. Hal ini terjadi karena perbedaan cara pembagian beban *embedding* antar kanal dan perbedaan pesan akan memengaruhi total selisih yang terbentuk antara audio asli dan audio stego. Kesimpulannya, *frame size* tetap faktor yang paling dominan menaikkan SNR, sedangkan skenario penyisipan menentukan seberapa besar SNR naik pada bitrate dan *frame size* yang sama.

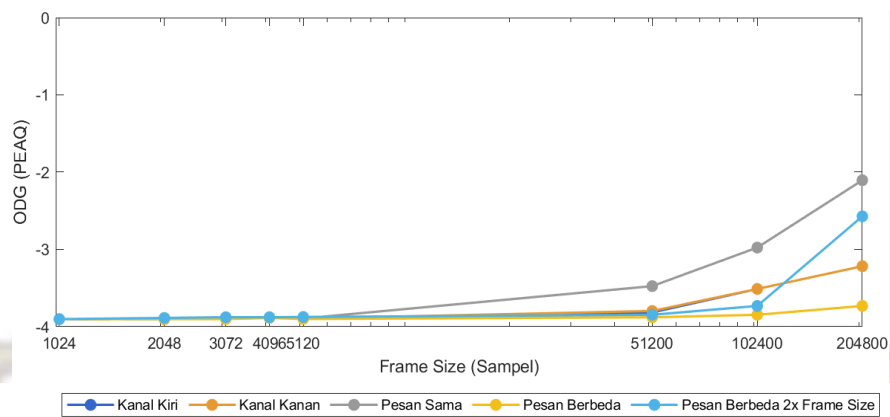
Dari hasil di atas, dapat disimpulkan bahwa terdapat *trade-off* antara kapasitas sistem dan kualitas penyisipan yang diukur melalui SNR. *Frame size* kecil 1024 - 5120 memberikan kapasitas yang lebih besar untuk menyisipkan pesan per detik, namun membutuhkan *embedding energy* yang lebih besar sehingga distorsi meningkat dan SNR menurun. Sebaliknya, *frame size* besar 51200, 102400, dan

204800 sampel yang memberikan kapasitas yang lebih kecil, namun memungkinkan penggunaan *embedding energy* yang lebih kecil sehingga distorsi sangat minimal dan SNR meningkat drastis. Untuk aplikasi yang memprioritaskan ketidakterdeteksian, penggunaan *frame size* besar dengan pola penyisipan yang sama pada kedua kanal (Pesan sama) adalah pilihan paling optimal, meskipun harus menerima konsekuensi penurunan kapasitas penyisipan per detik, Karena semakin besar *frame size*, semakin sedikit jumlah *frame* yang muat dalam 1 detik audio. Selain itu, kenaikan *bitrate* dari 64 kbps ke 128 kbps juga berpengaruh pada nilai SNR. Pada semua ukuran *frame*, SNR pada *bitrate* 128 kbps selalu lebih tinggi dibandingkan 64 kbps. Dengan *bitrate* yang lebih tinggi, perubahan sinyal audio asli dengan sinyal stego lebih baik sehingga proses penyisipan *watermark* tidak menimbulkan perubahan sebesar pada *bitrate* 64 kbps. Akibatnya, *noise* yang muncul akibat *embedding* jadi lebih kecil dan SNR naik.

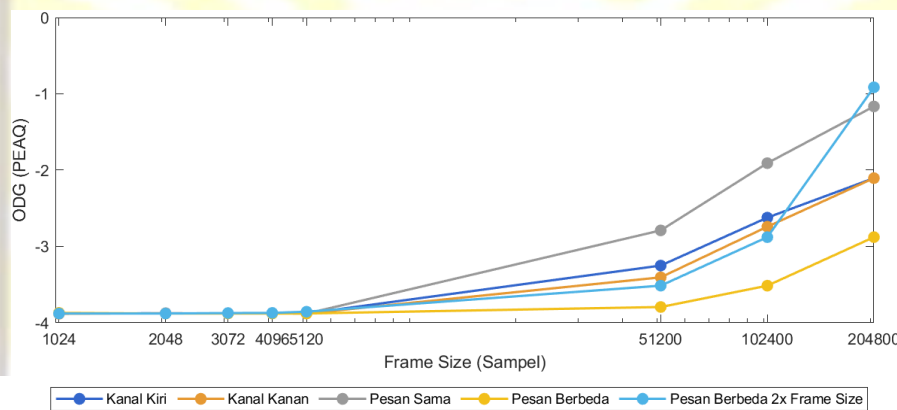
#### **4.1.3. Objective Different Grade**

##### **4.1.3.1. Pengaruh *Frame size* Pada Pengukuran *Objective Different Grade* Sebagai Indikator *Imperceptibility***

Analisis tingkat ketidakterdeteksian secara persepsi (*imperceptibility*) dari audio stego dilakukan menggunakan metrik *Objective Difference Grade* (ODG) yang mengukur perubahan sinyal menggunakan simulasi telinga manusia sebagai persepsi antara audio asli (*cover*) dan audio yang telah disisipi pesan (stego) berdasarkan model psikoakustik manusia. Nilai ODG berkisar dari 0 hingga -4, di mana nilai yang 0 menunjukkan perbedaan yang tidak dapat dideteksi oleh telinga manusia, sementara nilai yang semakin negatif mengindikasikan degradasi kualitas yang semakin mudah terdeteksi secara persepsi. Analisis perbandingan ukuran *frame* (N) ini dilakukan untuk mengevaluasi sejauh mana peningkatan *frame size* (N) dapat meningkatkan *imperceptibility* (yang diukur melalui ODG) pada setiap skenario penyisipan.



(a)



(b)

**Gambar 4.4** Pengaruh Ukuran *Frame size* terhadap audio stego yang diterima dengan menggunakan metrik *Objective Difference Grade* (ODG) pada Empat Skenario Penyisipan (Kanal kiri, Kanal kanan, Pesan sama, Pesan berbeda, dan Pesan berbeda 2x *frame size*) dengan *bitrate* (a) 64 dan (b) 128 kbps

Analisis tingkat ketidakterdeteksian (*imperceptibility*) melalui perubahan sinyal secara simulasi persepsi telinga manusia dilakukan menggunakan metrik *Objective Difference Grade* (ODG) yang mengukur perbedaan psikoakustik antara audio asli (*cover*) dan audio stego tepat setelah proses penyisipan pesan (*embedding*) selesai dilakukan. Berdasarkan standar ITU-R BS.1387, nilai ODG berkisar dari 0 hingga -4, di mana nilai 0 menunjukkan perbedaan yang tidak terdengar (*imperceptible*), sementara nilai -4 menunjukkan penurunan kualitas yang sangat mengganggu (*very annoying*). Pengujian ini mengevaluasi sejauh mana transparansi sinyal *watermark* dapat dipertahankan pada berbagai ukuran *Frame size* (1024 hingga 204800 sampel) sebelum audio tersebut memasuki tahap kompresi.

Gambar 4.4 Pada grafik (a) dengan *bitrate* 64 kbps, hasilnya cukup mengkhawatirkan untuk *Frame size* standar. Di rentang 1024 hingga 5120 sampel, semua skenario penyisipan terjebak di nilai ODG sekitar -3,8 hingga -3,9. Angka ini masuk kategori sangat mengganggu, yang artinya telinga manusia bisa dengan mudah mendeteksi adanya perubahan pada audio. Penyebabnya cukup jelas, ketika

pesan disisipkan dalam *frame* yang sempit, energi *watermark* jadi terlalu padat sehingga perubahan amplitudo di setiap sampel terasa sangat kentara. Kondisi ini tentu tidak ideal untuk keperluan steganografi karena tujuan utamanya adalah menyembunyikan pesan tanpa menimbulkan kecurigaan.

Nilai ODG mulai membaik ketika *Frame size* diperbesar ke skala ekstrem. Pada *Frame size* 51200 sampel, nilai ODG mulai bergeser ke kisaran -3,2 hingga -3,5 yang meskipun masih dalam kategori mengganggu, sudah menunjukkan perbaikan. Peningkatan lebih terasa di *Frame size* 102400 sampel dengan ODG -2,8 hingga -3,0. Di titik maksimal 204800 sampel, skenario Pesan sama berhasil mencapai nilai ODG sekitar -2,2 yang sudah masuk kategori sedikit mengganggu. Meski belum sempurna, pencapaian ini cukup berarti karena menunjukkan bahwa menyebarkan energi pesan ke jumlah sampel yang lebih banyak memang efektif untuk memperbaiki kualitas persepsi audio.

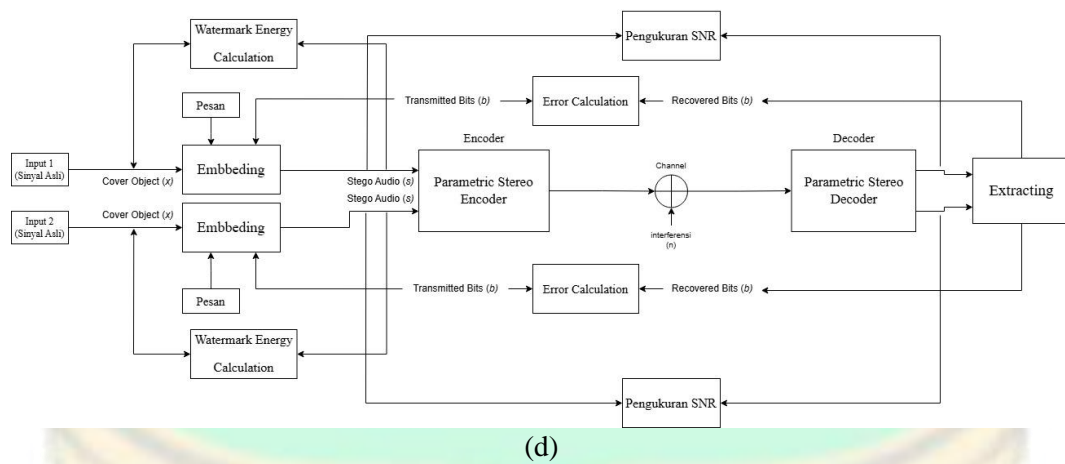
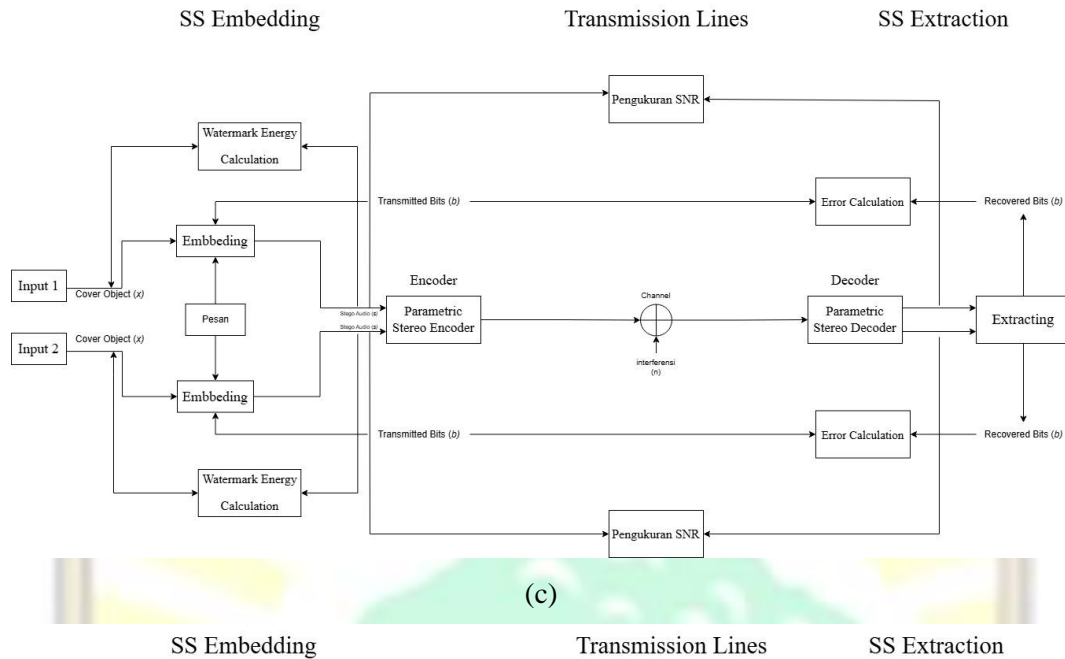
Hasil yang jauh lebih memuaskan terlihat pada grafik (b) dengan *bitrate* 128 kbps. Memang pada *Frame size* kecil 1024 - 5120, nilai ODG masih berada di level rendah sekitar -3,5 hingga -3,8, namun perbedaannya mulai terasa dibanding *bitrate* 64 kbps. Yang menarik adalah terlihat perubahan nilai ODG ketika memasuki *Frame size* besar 51200, 102400, dan 204800 sampel. Pada 51200 sampel, ODG sudah membaik ke nilai -2,5 hingga -3,0. Di *Frame size* 102400 sampel, nilainya naik lagi ke rentang -1,8 hingga -2,3. Puncaknya terjadi di *Frame size* 204800 sampel, di mana skenario Pesan berbeda 2x berhasil mencatatkan nilai ODG sekitar -1,0. Pencapaian ini sangat penting karena nilai -1,0 artinya audio stego sudah masuk kategori terdengar berbeda tapi tidak mengganggu. Dengan kata lain, meskipun ada sedikit perubahan yang bisa dideteksi, pendengar tidak akan merasa terganggu atau curiga bahwa ada sesuatu yang disembunyikan di dalam audio tersebut.

Pada hasil analisa tersebut, pesan yang disisipkan di kedua kanal (*stereo*) identik/sama, menyebabkan penambahan energi *watermark* terjadi secara sinkron di kedua kanal. Ini mengurangi ketidakseimbangan antar kanal dan menghasilkan distorsi yang lebih "merata", sehingga perubahan sinyal lebih sulit dikenali oleh sistem persepsi manusia atau model psikoakustik seperti PEAQ atau ODG.

Pada *bitrate* tinggi dan *frame size* yang lebih besar, adanya dua pesan berbeda namun energi *watermark* tersebar lebih banyak dan dilokasikan ke lebih banyak sampel, mengurangi konsentrasi energi modifikasi pada satu area. Dengan *frame size* sangat besar, meskipun pesan berbeda, modifikasi jadi sangat tipis per sample audio sehingga efek distorsinya juga sangat kecil itulah yang membuat ODG-nya bisa sangat baik hingga bernilai -0,9 pada variasi penyisipan pesan berbeda 2x *frame size* pada *frame size* 204800 (2x *frame size*, 409600 sampel) dengan *bitrate* 128 kbps.

Pada variasi penyisipan satu kanal dapat disimpulkan jika penyisipan hanya di satu kanal, efek *watermark* terlokalisasi pada salah satu kanal sehingga lebih mudah dikenali oleh sistem deteksi maupun manusia yang memperhatikan



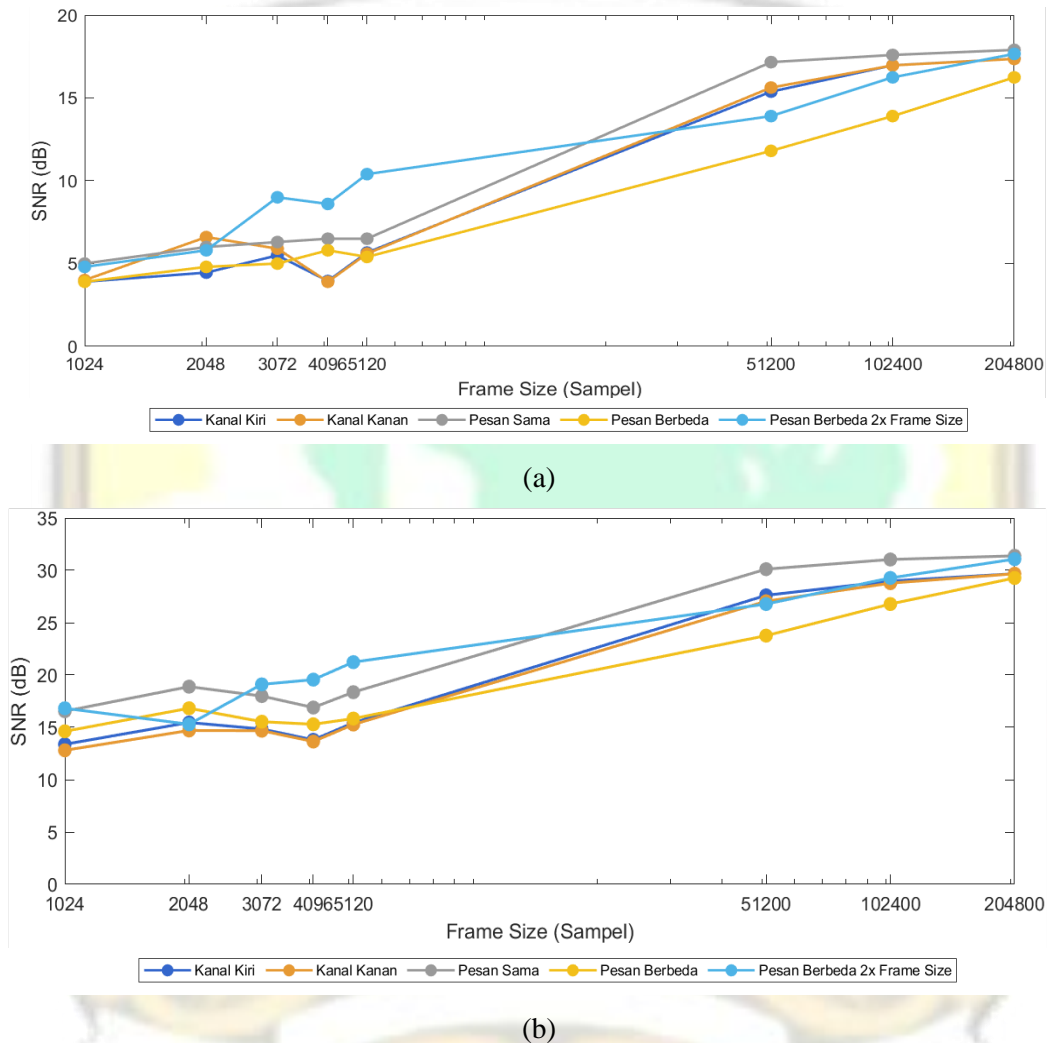


**Gambar 4.5** Blok Diagram Pengukuran SNR

Analisa tambahan ini dilakukan untuk mengetahui seberapa besar perubahan sinyal yang terjadi akibat proses penyisipan pesan rahasia setelah audio melalui tahapan kompresi. Hal tersebut dilakukan karena proses kompresi khususnya *Parametric Stereo*, menambahkan distorsi dan perubahan sinyal baru yang tidak muncul pada tahap *embedding* saja. Ketika audio stego dikompresi, *encoder Parametric Stereo* melakukan *downmix* dan merekonstruksi ulang sinyal *stereo* berdasarkan parameter-parameter tertentu. Proses ini bisa memperburuk ketidakseimbangan energi antar kanal yang disebabkan oleh penyisipan *watermark*, terutama jika *embedding* dilakukan secara tidak seimbang seperti pada variasi kanal tunggal atau pesan berbeda. Pada gambar 4.5. Gambar tersebut menunjukkan blok diagram pengukuran SNR, dengan mengukur SNR antara audio stego sebelum dan

sesudah kompresi, kita bisa melihat seberapa besar tambahan perubahan sinyal yang muncul akibat interaksi antara *watermark* dan proses *kompresi* tersebut. Hal ini penting karena perubahan sinyal total bukan hanya berasal dari *embedding*, tetapi juga dari bagaimana *watermark* tersebut berinteraksi dengan *codec* saat dikompresi. Jadi, analisa tambahan ini memberikan evaluasi lebih tentang dampak keseluruhan dari teknik penyisipan yang digunakan.

#### 4.1.4.1. Analisis Proses dan SNR pada Variasi Penyisipan Pesan



**Gambar 4. 6** SNR pada Stego Audio setelah tahap *Embedding* dan setelah Kompresi Audio dengan *Parametric Stereo* pada *bitrate* (a) 64 kbps dan (b) 128 kbps

Pada Gambar 4.6 menunjukkan hasil SNR pada stego audio setelah proses *embedding* dan setelah kompresi audio menggunakan *Parametric Stereo* pada *bitrate* 64 kbps dan 128 kbps. Pada variasi penyisipan pesan di kanal tunggal, baik kiri maupun kanan, proses *embedding* hanya dilakukan pada salah satu kanal, sehingga energi *watermark* hanya bertambah di sisi tersebut, sedangkan kanal lainnya

tetap dibiarkan tanpa perubahan. Hal ini menyebabkan energi di kedua kanal menjadi tidak seimbang. Kondisi ketidakseimbangan seperti ini berdampak pada hasil setelah melalui proses *Parametric Stereo*, di mana *noise* dari *watermark* pada kanal yang diberi *embedding* cenderung lebih terlihat. Kanal yang tidak diberi *watermark* tetap lebih stabil, sehingga SNR-nya lebih tinggi.

Sebagai contoh, pada *bitrate* 128 kbps dan *frame size* 102.400 sampel, kanal kiri yang diberi *watermark* memiliki SNR 28,98 dB, sedangkan kanal kanan yang tidak di-*embedding* mencatat SNR 30,08 dB. Hal ini menunjukkan bahwa kanal yang terkena *embedding* memang mengalami distorsi lebih tinggi akibat adanya *watermark*. Perlu dicatat, proses *Parametric Stereo* pada dasarnya merupakan tahap penggabungan dua kanal lalu memisahkan kembali menjadi dua kanal (*downmix* dan *upmix*). Penurunan kualitas yang terjadi pada kasus ini lebih banyak muncul karena dari awal sudah ada ketidakseimbangan *watermark* antar kanal, bukan sepenuhnya dikarenakan proses *Parametric Stereo* itu sendiri.

Hal serupa terjadi pada penyisipan di kanal kanan saja. Energi *watermark* hanya bertambah di kanal kanan, sehingga ketidakseimbangan antara dua kanal tetap terjadi. Proses *downmix Parametric Stereo* menampakkan ketidakseimbangan ini dalam hasil perubahan sinyal, sehingga SNR pada kanal yang diberi *watermark* lebih rendah dibanding kanal yang tidak dimodifikasi. Contohnya, pada *bitrate* 128 kbps dan *frame size* 102.400 sampel, SNR kanal kanan tercatat 28,98 dB, sementara kanal kiri bertahan di 30,08 dB.

Pada variasi pesan sama, *watermark* disisipkan di kedua kanal dengan pola yang identik. Hal ini menjaga keseimbangan energi antara kanal kiri dan kanan. Saat audio melewati proses *Parametric Stereo*, *watermark* yang identik di kedua kanal ikut tercampur rata, sehingga hasil perubahan sinyal audionya menjadi lebih stabil. *Noise* akibat *embedding* juga menjadi lebih kecil, dan SNR pada kedua kanal pun seimbang, misalnya 31,04 dB pada kedua sisi di *bitrate* 128 kbps dan *frame size* 102.400 sampel. Kondisi ini menunjukkan bahwa penyisipan pesan yang seragam pada dua kanal dapat meminimalkan distorsi dan menjaga kualitas sinyal hasil kompresi tetap baik.

Pada variasi pesan berbeda, *watermark* yang disisipkan pada kanal kiri dan kanan tidak sama. Ketidakseimbangan *watermark* seperti ini menyebabkan *noise* tambahan yang muncul setelah proses *Parametric Stereo* menjadi lebih besar. Nilai SNR pada kedua kanal pun turun, misalnya di *bitrate* 128 kbps dan *frame size* 102.400, SNR kanal kiri 26,77 dB dan kanal kanan 26,78 dB. Peningkatan *frame size* dan *bitrate* sangat membantu mengurangi gangguan tersebut, dan perbedaan *watermark* antar kanal tetap memberi efek terhadap penurunan nilai SNR hasil rekonstruksi.

Pada variasi pesan berbeda  $2x$  *frame size*, kapasitas pesan rahasia diturunkan dengan memperbesar *frame size*, misalnya hingga 204.800 sampel. Meski *watermark* pada dua kanal tetap berbeda, jumlah sampel yang banyak membuat perubahan pada setiap titik audio semakin kecil. Dampaknya, SNR pada kedua kanal

jadi lebih tinggi dan hampir menyamai hasil pada variasi pesan sama. Sebagai contoh, pada *bitrate* 128 kbps dan *frame size* 204.800, SNR kanal kiri dan kanan tercatat masing-masing 31,06 dB dan 31,08 dB.

Pada analisa tersebut dapat disimpulkan bahwa *frame size* yang besar dapat memperkecil setiap gangguan atau ketidakseimbangan *watermark* antar kanal pada parameter SNR dikarenakan kapasitas pengiriman yang dibutuhkan sangat sedikit, sehingga membutuhkan *embedding energy* yang sedikit, *embedding energy* yang disisipkan tersebut berupa *noise*, jika terlalu banyak maka akan mengakibatkan penurunan SNR dan sebaliknya. Proses *Parametric Stereo* pada dasarnya hanya menggabungkan dan memisahkan ulang dua kanal, efek dominan pada nilai SNR lebih banyak ditentukan oleh seberapa seimbang *embedding energy* yang disisipkan pada awal proses. Dapat terlihat pada pesan berbeda  $2x$  *frame size*, Meskipun tetap ada ketidakseimbangan antar kanal, efeknya tertutup oleh kapasitas pengiriman yang sedikit dengan jumlah sampel yang sangat banyak. Akibatnya, perubahan sinyal tambahan akibat kompresi juga sangat kecil dan SNR tetap tinggi, bahkan bisa menyamai atau melampaui pesan sama pada *frame size* besar. Pada variasi pesan sama, *watermark* yang disisipkan di kanal kiri dan kanan selalu sama. Karena pola *watermark* di kedua kanal identik, saat audio diproses dan digabungkan (mixing dan kompresi), kedua pesan ini justru terkadang saling memperkuat di sisi lain, kadang juga bisa saling meniadakan bagian-bagian kecil dari perubahan yang muncul. Dengan kata lain, perubahan yang terjadi tidak pernah "menumpuk" sendiri di salah satu kanal. Ini membuat *watermark* yang dihasilkan jauh lebih kecil, jadi nilai SNR yang diukur pun menjadi lebih tinggi. Berbeda dengan variasi penyisipan pesan berbeda, di mana *embedding energy* yang disisipkan isi dan polanya memang tidak sama. Dalam kasus ini, setiap kanal pada pesan berbeda membawa perubahan sendiri-sendiri yang tidak bisa saling membantu atau memungkinkan meniadakan. Akibatnya, perubahan sinyal yang terjadi cenderung lebih besar. Inilah sebabnya SNR untuk pesan sama bisa tetap lebih tinggi daripada pesan berbeda, baik sebelum maupun setelah kompresi. Singkatnya, karena *watermark* pada pesan sama bisa saling menguatkan atau menyeimbangkan, hasil akhirnya lebih stabil dan lebih aman untuk menjaga perubahan sinyal.